



RES - 2024 - 722 - CD-ECO # UNNE

Sesión 12-12-2024.

VISTO:

El Expediente 2024-19177; y

CONSIDERANDO:

Que en el mencionado Expediente obra nota del Secretario de Extensión y Ejercicio Profesional, a través de la cual eleva propuesta para llevar adelante la **Diplomatura en Herramientas Digitales para la Gestión y Análisis de Negocios**;

Esta **Diplomatura** es una oferta de capacitación de nivel universitario dependiente de la Secretaría de Extensión y Ejercicio Profesional de la Facultad de Ciencias Económicas de la UNNE;

Que los objetivos de la Diplomatura son: desarrollar competencias técnicas progresivas en el uso de herramientas digitales clave, que permitan a los participantes gestionar y analizar información de manera eficiente; fomentar la capacidad de implementar soluciones automatizadas y personalizadas; promover el análisis estratégico de datos; facilitar la aplicación práctica de las herramientas adquiridas en escenarios reales e impulsar el desarrollo de habilidades transversales en innovación tecnológica;

Que la **Diplomatura en Herramientas Digitales para la Gestión y Análisis de Negocios**, se desarrollará en modalidad virtual sincrónica y asincrónica, a través de la plataforma Moodle de UNNE Virtual, y esta destinada a profesionales, estudiantes avanzados y emprendedores vinculados a las ciencias económicas, la administración y áreas afines que deseen potenciar sus competencias en el uso de herramientas digitales para la gestión y análisis de información, orientada a gestores empresariales, contadores, consultores y responsables de la toma de decisiones en organizaciones que buscan mejorar la productividad, optimizar procesos y generar valor a través del manejo estratégico de datos;

Lo aconsejado por la Comisión de Enseñanza e Investigación;

Lo resuelto por el Consejo Directivo en Sesión Ordinaria de fecha 12-12-2024;



Por ello:

**EL CONSEJO DIRECTIVO
DE LA FACULTAD DE CIENCIAS ECONOMICAS
RESUELVE:**

ARTICULO 1º: Aprobar el dictado de la **Diplomatura en Herramientas Digitales para la Gestión y Análisis de Negocios**, con las características que figuran en el Anexo de la presente Resolución.

ARTICULO 2º: Establecer que la **Diplomatura en Herramientas Digitales para la Gestión y Análisis de Negocios**, se desarrollará en modalidad virtual sincrónica y asincrónica a través de la plataforma Moodle de UNNE Virtual, con una **carga horaria total de doscientos cuarenta (240) horas** y una **duración total de ocho (8) bimestres**.

ARTICULO 3º: Elevar copia de la presente Resolución a la Secretaría General Académica de la Universidad Nacional del Nordeste para su conocimiento, registro y difusión.

ARTICULO 4º: Registrar la presente Resolución, efectuar las comunicaciones correspondientes y oportunamente proceder a su archivo.

VERONICA M. L. GLIBOTA LANDRIEL
SECRETARIA ACADÉMICA

MOIRA YANINA CARRIO
DECANA

ANEXO

DIPLOMATURA EN HERRAMIENTAS DIGITALES PARA LA GESTIÓN Y ANÁLISIS DE NEGOCIOS

FACULTAD DE CIENCIAS ECONÓMICAS – UNNE

DENOMINACIÓN:

Diplomatura en Herramientas Digitales para la Gestión y Análisis de Negocios.

UNIDAD RESPONSABLE

Esta Diplomatura es una oferta de capacitación de nivel universitario dependiente de la Secretaría de Extensión y Ejercicio Profesional de la Facultad de Ciencias Económicas de la UNNE.

FORMACIÓN

La Diplomatura en Herramientas Digitales para la Gestión y Análisis de Negocios de la Facultad de Ciencias Económicas de la UNNE, les permitirá a los estudiantes adquirir conocimientos, herramientas y aptitudes desde la construcción de saberes teóricos y saberes prácticos en el campo de la tecnología digital aplicada a negocios.

OBJETIVOS INSTITUCIONALES Y FUNDAMENTACIÓN DE LA PROPUESTA

En un contexto global caracterizado por la transformación digital y la creciente complejidad de los datos, el manejo eficiente de herramientas tecnológicas se ha convertido en un requisito indispensable para los profesionales de las ciencias económicas. Las empresas y organizaciones demandan perfiles capaces de gestionar información de manera efectiva, realizar análisis estratégicos y optimizar procesos mediante el uso de herramientas digitales avanzadas. En este escenario, la Diplomatura en Herramientas Digitales para la Gestión y Análisis ofrece una formación integral que combina conocimiento técnico, habilidades prácticas y enfoques innovadores para abordar los desafíos de la era digital.

Esta diplomatura se estructura en módulos cuidadosamente diseñados para desarrollar competencias progresivas, comenzando con una sólida base en Excel, desde lo básico hasta funciones avanzadas, y extendiéndose hacia el manejo de herramientas especializadas como Power Query, Power Pivot y Power BI. Estos programas no solo fomentan la eficiencia en la organización y análisis de grandes volúmenes de datos, sino que también potencian la capacidad de crear soluciones dinámicas y automatizadas mediante lenguajes como Visual Basic y aplicaciones modernas como Appsheet. De esta manera, el programa se adapta tanto a quienes desean iniciarse en el ámbito digital como a aquellos que buscan especializarse y perfeccionar sus habilidades.

La diplomatura no solo se enfoca en el uso técnico de herramientas, sino también en su aplicación práctica en contextos reales, como la contabilidad, la gestión empresarial y el análisis de datos para la toma de decisiones estratégicas. Asimismo, los módulos optativos y complementarios permiten personalizar la formación según los intereses y necesidades específicas de los participantes, promoviendo una experiencia educativa relevante y alineada con las exigencias del mercado laboral. Esto resulta especialmente valioso en un entorno donde la versatilidad y la capacidad de adaptación son clave para destacar como profesional.

Con un enfoque innovador y orientado a resultados, esta diplomatura busca empoderar a los participantes para enfrentar los retos del futuro profesional, ya sea desde la gestión de emprendimientos, la optimización de procesos en organizaciones o el análisis avanzado de datos en entornos altamente competitivos.

De este modo, la Facultad de Ciencias Económicas de la UNNE reafirma su compromiso con la excelencia académica y su contribución al desarrollo profesional y empresarial en la región, formando especialistas preparados para liderar el cambio en la economía digital.

Objetivos de la Diplomatura

- Desarrollar competencias técnicas progresivas en el uso de herramientas digitales clave, que permitan a los participantes gestionar y analizar información de manera eficiente.
- Fomentar la capacidad de implementar soluciones automatizadas y personalizadas.
- Promover el análisis estratégico de datos.
- Facilitar la aplicación práctica de las herramientas adquiridas en escenarios reales.
- Impulsar el desarrollo de habilidades transversales en innovación tecnológica.

Estructura del Programa

El programa de la diplomatura está diseñado para ser completado en un máximo de 240 horas, distribuidas en ocho módulos. Cada módulo está estructurado para ofrecer una combinación de teoría y práctica, asegurando una formación robusta y aplicable.

CARGA HORARIA TOTAL

La carga horaria total de este curso es de doscientos cuarenta (240) horas.

DURACIÓN

La carga horaria mencionada estará distribuida en ocho (8) bimestres.

MODALIDAD

La Diplomatura en Herramientas Digitales para la Gestión y Análisis de Negocios se desarrollará en modalidad virtual sincrónica y asincrónica.

DESTINATARIOS

La Diplomatura en Herramientas Digitales para la Gestión y Análisis de Negocios está dirigida a profesionales, estudiantes avanzados y emprendedores vinculados a las ciencias económicas, la administración y áreas afines que deseen potenciar sus competencias en el uso de herramientas digitales para la gestión y análisis de información. También está orientada a gestores empresariales, contadores, consultores y responsables de la toma de decisiones en organizaciones que buscan mejorar la productividad, optimizar procesos y generar valor a través del manejo estratégico de datos. Asimismo, es una excelente opción para quienes deseen actualizarse o incursionar en tecnologías digitales aplicadas a la gestión y el emprendimiento.

CONTENIDOS DEL PLAN DE ESTUDIO

La Diplomatura se compone de ocho módulos que se describen y detallan a continuación.

Módulos	Carga horaria
1. Excel Básico	30 hs
2. Excel Avanzado	30 hs
3. Power Query	30 hs
4. Power Pivot	30 hs
5. Visual Basic Básico	30 hs
6. Visual Basic Avanzado	30 hs
7. AppSheet Básico	30 hs
8. AppSheet Avanzado	30 hs
Total	240 hs

Descripción de contenidos por módulo:

1. Excel Básico
 - **Introducción a Excel:**
 - Introducción teórica: aplicaciones prácticas en diferentes ámbitos.

- Navegación por la interfaz de usuario: descripción de la pantalla principal, componentes principales como barra de herramientas, cinta de opciones, barra de fórmulas y hoja de cálculo.
- Cinta de opciones: explicación de las pestañas principales como Inicio, Insertar, Diseño de página, Fórmulas, Datos, Revisar y Vista, además del uso del botón "Archivo" para crear, abrir y guardar archivos.
- Estructura de libros, hojas y celdas: diferencias entre un libro y una hoja, identificación de filas y columnas, y cómo agregar, eliminar y mover hojas dentro de un libro.
- Contenido y formato básico: introducir texto, números y fechas en las celdas, aplicar formatos básicos.
- Validaciones básicas: verificación del tipo de datos en celdas y uso de la herramienta de validación de datos para configurar rangos permitidos y listas desplegables básicas.
- Portapapeles y acciones básicas: uso del portapapeles para copiar, cortar y pegar contenido, diferencias entre copiar formato y copiar contenido, y uso del relleno automático para completar series como números, días o meses.

- **Introducción a fórmulas y funciones en Excel:**

- Conceptos básicos sobre qué son las fórmulas y funciones y cómo se utilizan en hojas de cálculo para realizar cálculos automáticos y dinámicos.
- Operadores matemáticos y lógicos: uso de operadores como suma (+), resta (-), multiplicación (*), división (/), y comparativos como mayor que (>), menor que (<), igual (=), entre otros, para realizar operaciones y evaluaciones lógicas en celdas.
- Referencias: explicación de las referencias relativas, absolutas y mixtas, cómo afectan al copiar fórmulas y cuándo utilizarlas para trabajar de forma más eficiente en hojas de cálculo complejas.
- Funciones básicas: introducción a funciones esenciales como SUMA, PROMEDIO, MÁXIMO, MÍNIMO y CONTAR, con ejemplos prácticos de su aplicación en la gestión y análisis de datos.
- Errores comunes: identificación de errores como #DIV/0! (división entre cero), #N/A (valor no disponible) y otros problemas comunes en fórmulas, además de estrategias para resolverlos o evitarlos

- **Funciones en Excel I:**

- introducción al concepto de funciones en Excel como herramientas predefinidas para realizar cálculos complejos y tareas específicas de forma eficiente.
- Funciones matemáticas: uso de funciones como SUMA, PRODUCTO, REDONDEAR, ENTERO y POTENCIA, con ejemplos prácticos aplicados a cálculos financieros y contables.
- Funciones de texto: manipulación de cadenas de texto con funciones como CONCATENAR, IZQUIERDA, DERECHA, LARGO y MAYUSC, para organizar y presentar información textual en las hojas de cálculo.

- Funciones de fecha y hora: utilización de funciones como HOY, AHORA, FECHA, DIA, MES y AÑO, para gestionar y calcular valores relacionados con fechas y horarios, como vencimientos o programaciones.

- **Funciones en Excel II:**

- Introducción a las funciones lógicas como herramientas esenciales para realizar evaluaciones y tomar decisiones basadas en condiciones específicas dentro de una hoja de cálculo.
- Funciones lógicas: explicación y uso de la función SI, con ejemplos prácticos que incluyen evaluaciones condicionales simples como clasificaciones y cálculos basados en criterios específicos.
- Anidación de funciones lógicas: creación de fórmulas avanzadas mediante la combinación de varias funciones SI en una sola fórmula, con ejemplos aplicados a casos comunes en economía y contabilidad.
- Función SI.ERROR: descripción y aplicación para gestionar errores comunes en fórmulas.

- **Funciones en Excel III:**

- introducción a las funciones lógicas compuestas, que permiten evaluar múltiples condiciones dentro de una sola fórmula.
- Funciones Y y O: explicación detallada de estas funciones, cómo se utilizan para evaluar múltiples condiciones al mismo tiempo y cómo combinar múltiples criterios lógicos dentro de una celda.
- Ejemplos prácticos para determinar si se cumplen varios requisitos, como en situaciones de validación o análisis de datos complejos. Diferencias con anidaciones: comparación entre el uso de funciones lógicas simples (como SI) y lógicas compuestas (como Y y O), y cómo elegir la mejor opción según el escenario.
- Estados posibles: descripción de los resultados posibles en funciones lógicas compuestas (VERDADERO, FALSO), y cómo utilizar estos resultados para tomar decisiones en análisis y reportes financieros o contables

- **Herramientas adicionales de Excel:**

- introducción a herramientas útiles que mejoran la eficiencia y la organización dentro de las hojas de cálculo.
- Quitar duplicados: explicación de cómo usar la función “Quitar duplicados” para eliminar valores repetidos en un rango de datos o en una tabla, lo que es útil para limpieza y organización de datos.
- Validaciones de datos: configuración de reglas de validación para asegurar que los datos introducidos en las celdas sean correctos según criterios específicos (como listas desplegables, rangos numéricos o fechas).
- Inmovilización de paneles: descripción de la función de inmovilización de paneles para mantener visibles filas o columnas clave mientras se desplaza por grandes hojas de cálculo.

- Buscar y reemplazar: uso de la herramienta "Buscar" para encontrar información específica en la hoja de cálculo, y "Reemplazar" para modificar rápidamente valores o formatos en múltiples celdas de forma eficiente.
- **Formatos en Excel.**
 - introducción al formato condicional como una herramienta para resaltar datos en función de ciertas condiciones o reglas.
 - Creación de reglas de formato condicional simples, como resaltar los valores mayores o menores que un número específico, o las celdas que contengan un texto particular.
 - Diferencias entre Google Sheets y Excel: comparación de las funcionalidades y herramientas de formato condicional en Excel y Google Sheets, destacando las similitudes y limitaciones de ambas plataformas. Discusión sobre la compatibilidad de funciones y la facilidad de uso entre ambas herramientas, especialmente en relación con el formato condicional y las funciones de automatización.
 - Crear formato condicional con funciones: cómo combinar funciones como SI, SUMA, o CONTAR con formato condicional para aplicar reglas dinámicas y personalizadas, como cambiar el color de una celda si el valor es el más alto o bajo de una lista, o si cumple con una condición compleja
- **Funciones en Excel IV:**
 - Funciones de referencia y búsqueda: introducción a las funciones que permiten buscar y referenciar datos dentro de una hoja de cálculo, esenciales para gestionar grandes volúmenes de información.
 - BUSCARV: explicación y aplicación de la función BUSCARV (Búsqueda Vertical) para buscar un valor en la primera columna de un rango y devolver un valor en la misma fila, pero en una columna diferente.
 - Diferencias entre búsqueda exacta y aproximada, y cómo seleccionar el tipo de coincidencia en función del análisis que se desea realizar.
 - BUSCARX: presentación de la función BUSCARX como una alternativa más avanzada a BUSCARV, que permite buscar un valor en cualquier parte de la hoja de cálculo y devolver un valor relacionado en una columna o fila específica. Comparación de BUSCARV y BUSCARX en términos de flexibilidad y facilidad de uso.
 - Función COINCIDIR: explicación de la función COINCIDIR, que se utiliza para buscar un valor en un rango y devolver la posición relativa de ese valor. Ejemplos de cómo combinar COINCIDIR con otras funciones como INDICE para realizar búsquedas más complejas y personalizadas.
- **Tablas y Filtros en Excel.**
 - Tablas y Filtros en Excel: introducción a las herramientas de tablas y filtros en Excel, que permiten organizar y analizar datos de manera más eficiente y dinámica.
 - Beneficios de tablas: explicación de las ventajas de convertir un rango de datos en una tabla (como la actualización automática de rangos al agregar nuevos datos, la posibilidad

de aplicar filtros fácilmente, y el uso de estilos predefinidos). Cómo utilizar las funciones de totalización automática, como las sumas, promedios, etc., directamente en las tablas.

- Orden con multi criterio: explicación de cómo ordenar los datos de una tabla o rango de celdas utilizando varios criterios, como ordenar por más de una columna simultáneamente (por ejemplo, ordenar por fecha y luego por valor). Aplicación práctica de ordenación ascendente y descendente, así como el uso de listas personalizadas para ordenar según criterios específicos.
- Filtros de números, textos y fechas: uso de filtros avanzados en Excel para filtrar datos según valores numéricos, cadenas de texto o fechas. Ejemplos de filtros con criterios específicos, como filtrar valores superiores a un número, fechas dentro de un rango o textos que contengan una palabra clave.
- Segmentadores: introducción a los segmentadores como herramienta de filtrado visual en tablas y gráficos. Cómo crear y utilizar segmentadores para permitir a los usuarios filtrar datos fácilmente de manera interactiva, aplicando filtros de múltiples campos con un solo clic.

- **Funciones en Excel V:**

- Funciones lógico-matemáticas: introducción a las funciones que combinan lógica y matemáticas para realizar cálculos en función de ciertas condiciones o criterios.
- Funciones lógico-matemáticas (SUMAR.SI, CONTAR.SI, etc.): explicación de funciones como SUMAR.SI y CONTAR.SI, que permiten realizar sumas o contar elementos en un rango según un criterio específico. Ejemplos de uso en contextos financieros y contables.
- Combinación con validaciones de datos: cómo integrar funciones lógico-matemáticas con validaciones de datos para realizar análisis más dinámicos, como asegurarse de que los valores ingresados en una celda cumplen con criterios antes de aplicar una fórmula.
- Condiciones simples y complejas: explicación de la diferencia entre condiciones simples y condiciones complejas y cómo aplicar ambas en las funciones lógico-matemáticas.

- **Gráficos en Excel.**

- introducción a la creación y personalización de gráficos en Excel para visualizar datos de manera más comprensible y efectiva.
- Gráficos generales (Barras, columnas y líneas): explicación de cómo crear y personalizar gráficos básicos como gráficos de barras, columnas y líneas. Discusión sobre cuándo utilizar cada tipo de gráfico según el tipo de datos y el análisis que se desea mostrar (por ejemplo, usar gráficos de barras para comparar categorías y gráficos de líneas para mostrar tendencias a lo largo del tiempo).
- Minigráficos: explicación de los minigráficos como una forma compacta y visual de representar tendencias dentro de una celda, utilizando gráficos de líneas, barras o win/loss.
- Gráficos combinados: descripción de cómo crear gráficos combinados, que permiten mostrar diferentes tipos de gráficos en un solo gráfico (por ejemplo, barras y líneas en el

mismo gráfico). Ejemplos de cuándo y cómo usar gráficos combinados para mejorar la presentación de datos con diferentes unidades de medida o categorías.

- Elementos del gráfico: identificación y personalización de los elementos clave de un gráfico, como el título, las etiquetas de los ejes, las leyendas, las etiquetas de datos y las líneas de cuadrícula. Cómo modificar cada elemento para hacer el gráfico más claro y visualmente atractivo.

• **Ejercicio: Cuadro de Amortización.**

- Introducción al cuadro de amortización: explicación de qué es un cuadro de amortización y su propósito, cómo se utiliza en contabilidad y finanzas para distribuir los pagos de un préstamo a lo largo del tiempo. Conceptos clave: definición de términos como capital, intereses, cuota fija, saldo de la deuda, tasa de interés, período y frecuencia de pago.
- Creación del cuadro de amortización: uso de funciones como PAGO, INTERES.CUOTA, para calcular las cuotas fijas, los pagos de interés y los pagos de capital de cada período.
- Cálculos y formato: cómo organizar los datos en una tabla que incluya el número de pago, el monto de la cuota, el pago de interés, el pago de capital y el saldo restante. Se explicará cómo usar fórmulas para calcular el pago de cada componente y cómo aplicar el formato adecuado para mejorar la presentación y claridad de los datos.
- Análisis del cuadro de amortización: cómo interpretar el cuadro de amortización y analizar el impacto de diferentes tasas de interés y plazos sobre el saldo pendiente y los pagos. Ejemplos de cómo ajustar los parámetros para realizar simulaciones y evaluar distintos escenarios financieros.

• **Ejercicio: Liquidación de Sueldos.**

- Introducción a la liquidación de sueldos: explicación de qué es una liquidación de sueldos, su importancia y las principales variables que influyen en ella, como el salario bruto, deducciones (impuestos, seguridad social, etc.), y salario neto.
- Creación de la plantilla de liquidación: uso de tablas, donde ingresarán datos como el nombre del empleado, horas trabajadas, tarifa por hora, bonos, deducciones y el total de las deducciones.
- Cálculos en la plantilla: explicación sobre cómo usar fórmulas para calcular el salario bruto (salario base + horas extras + bonificaciones), las deducciones (porcentaje de impuestos o seguridad social), y finalmente el salario neto. Se explicará cómo utilizar funciones como SUMA, PRODUCTO, PORCENTAJE, y fórmulas condicionales como SI para aplicar deducciones y bonificaciones.
- Formato de la liquidación: cómo organizar y formatear la hoja de cálculo para que sea clara y profesional. Se enseñará a aplicar formato a las celdas, agregar bordes y usar colores para resaltar totales y subtotales.

- Simulación y análisis: los estudiantes realizarán una simulación con varios empleados, variando las horas trabajadas, el tipo de deducciones y beneficios, para observar cómo se ajustan los cálculos y comprender la flexibilidad de la plantilla para diferentes escenarios.
 - **Repaso:**
 - La clase de repaso se enfocará en revisar los conceptos clave aprendidos durante las clases anteriores de Excel Básico.
 - Se realizará una recapitulación de las herramientas y funciones más importantes, resolviendo dudas y reforzando los conocimientos mediante ejercicios prácticos.
 - Los estudiantes tendrán la oportunidad de aplicar lo aprendido en casos reales y recibir retroalimentación personalizada para afianzar su comprensión y destreza en el uso de Excel.
 - **Evaluación.**
 - La evaluación consistirá en un ejercicio práctico donde los estudiantes deberán aplicar los conceptos y herramientas aprendidas durante el curso de Excel Básico.
 - Se evaluará su capacidad para crear y gestionar hojas de cálculo, utilizar fórmulas y funciones adecuadas, aplicar formato y generar gráficos.
2. Excel Avanzado
- **Introducción a Bases de datos.**
 - Introducción a Bases de Datos: bases de datos en Excel, su importancia y cómo utilizar Excel para gestionar grandes volúmenes de información.
 - Claves de datos: explicación sobre las claves primarias y foráneas en bases de datos. Se discutirá cómo utilizar identificadores únicos (como ID de cliente o código de producto) para relacionar registros y asegurar la integridad de los datos.
 - Campos y Registros: descripción de los campos (columnas) y registros (filas) dentro de una base de datos en Excel. Se hará énfasis en cómo cada campo representa un atributo específico (por ejemplo, nombre, fecha de nacimiento, precio) y cómo los registros son las entradas completas de datos en las filas.
 - Combinación de Tablas: introducción al concepto de relaciones entre tablas en una base de datos. Se enseñará a los estudiantes cómo combinar tablas de diferentes fuentes mediante la utilización de funciones de búsqueda como BUSCARV, BUSCARX, ÍNDICE y COINCIDIR.
 - Funciones Básicas: en este contexto, se repasarán las funciones básicas que son fundamentales al trabajar con bases de datos en Excel, como SUMAR.SI, CONTAR.SI, PROMEDIO.SI, entre otras, para realizar cálculos basados en condiciones.
 - Filtros: se explorará cómo usar los filtros avanzados para extraer información específica de una base de datos, permitiendo a los usuarios realizar consultas por fecha, valores numéricos, texto, etc. Los estudiantes aprenderán a aplicar filtros automáticos y personalizados, con ejemplos prácticos de filtrado por múltiples criterios.
 - **Tablas en Excel.**

- Tablas en Excel: crear y trabajar con tablas en Excel. Convertir rangos de datos en tablas formateadas, lo que permitirá gestionar mejor los datos, con funcionalidades automáticas como la adición de filtros y la expansión automática de los rangos al agregar nuevos datos.
- Tablas Complejas: explicación sobre cómo manejar tablas complejas que incluyen múltiples columnas con datos variados (números, texto, fechas). Se enseñará a personalizar las tablas agregando formatos condicionales, utilizando validaciones de datos y configurando cálculos personalizados para columnas. Los estudiantes aprenderán a utilizar las herramientas de tabla avanzada para crear relaciones y realizar análisis más sofisticados.
- Subtotales: los estudiantes aprenderán a aplicar la herramienta de Subtotales en Excel para calcular totales por grupos de datos, lo que es muy útil en el análisis de grandes volúmenes de información. Se explicará cómo agrupar datos por categorías y cómo aplicar diferentes funciones de resumen (como suma, promedio, máximo, mínimo) a cada grupo. Además, se enseñará a agregar y quitar subtotales de manera dinámica.
- Filtros Avanzados: en esta sección, se explorarán los filtros avanzados en Excel, una herramienta esencial para trabajar con bases de datos más grandes. Los estudiantes aprenderán a crear filtros personalizados utilizando condiciones complejas que involucren múltiples criterios, operadores lógicos y de comparación (como "Y" o "O"). Se explicará cómo usar filtros avanzados para extraer registros que cumplan con condiciones específicas y cómo copiar los resultados filtrados a otro rango o hoja de cálculo
- **Variables Calculadas.**
- Campos o variables calculadas: crear campos o columnas calculadas en tablas y bases de datos. Se explicará cómo agregar nuevas columnas que dependan de otros datos existentes en la tabla mediante fórmulas y funciones.
- Uso de Lógicas: se profundizará en el uso de funciones lógicas para crear fórmulas que realicen cálculos condicionales. Se cubrirán funciones como SI, Y, O, y cómo usarlas para evaluar condiciones múltiples dentro de una base de datos o tabla. Los estudiantes aprenderán a aplicar estas funciones para controlar los resultados en función de las condiciones.
- BuscarV: se revisará a fondo el uso de la función BUSCARV para su utilización en modelos de datos.
- Funciones SI.CONJUNTO: Aplicación de esta función para el cálculo de campos de tabla
- Auditoría de funciones: uso de "Mostrar fórmulas" y las herramientas de "Rastrear precedentes" y "Rastrear dependientes" para identificar errores en las fórmulas y asegurarse de que los cálculos sean correctos.
- **Google Sheets.**
- Tablas en Google Sheets: se enseñará a los estudiantes a crear y gestionar tablas en Google Sheets, con un enfoque en las funcionalidades específicas de esta herramienta. Además, se explorarán las características de formato, como el uso de colores, bordes, y estilos para mejorar la presentación de las tablas.

- Funciones Matriciales: en esta parte de la clase, los estudiantes aprenderán a utilizar funciones matriciales en Google Sheets. Se explicará cómo las funciones como FILTER, SORT, UNIQUE.
- **Funciones de texto para limpieza.**
- Funciones de texto para limpieza en Excel: limpiar y manipular datos en formato de texto.
- Funciones útiles: SUSTITUIR, LARGO, HALLAR, ESPACIOS, MAYUSC, MINUSC, CONCATENAR
- Texto en Columnas: se explicará cómo usar la herramienta de Texto en Columnas en Excel para dividir datos en columnas separadas.
- Archivos CSV y Txt: se discutirá el manejo de archivos CSV y TXT en Excel, que son formatos comunes para exportar e importar datos.
- **Introducción a Tablas Dinámicas.**
- Orígenes de Datos: seleccionar y preparar los datos que se usarán como base para una tabla dinámica
- Campos de Tabla Dinámica: panel de campos de la tabla dinámica, que incluye áreas como Filas, Columnas, Valores, y Filtros. Arrastrar y soltar campos en estas áreas para estructurar el informe dinámico
- Formatos: formato a una tabla dinámica para mejorar su presentación. Diseño de las tablas dinámicas utilizando los estilos predefinidos de Excel y personalizar formatos manualmente para resaltar datos importantes. Actualización automática de formatos al cambiar los datos de origen.
- Propiedades: ajustar las propiedades de las tablas dinámicas, incluyendo la configuración de opciones como Actualizar al abrir el archivo, Retener elementos eliminados, y la optimización del rendimiento con grandes volúmenes de datos.
- Agrupamiento de Variables: agrupar datos dentro de las tablas dinámicas, como agrupar fechas en meses, trimestres o años, y agrupar valores numéricos en rangos definidos. Crear categorías personalizadas para facilitar el análisis y organizar los datos de manera más clara.
- Mostrar Valores Como: Realizar cálculos adicionales en las tablas dinámicas. Cambiar el resumen de datos (por ejemplo, de suma a porcentaje del total), calcular diferencias año tras año, y mostrar datos como índices o contribuciones relativas.
- **Tablas y Gráficos Dinámicos.**
- Tablas y Gráficos Dinámicos: Descripción de los gráficos dinámicos, su vínculo directo con las tablas dinámicas, y los tipos de gráficos más adecuados para representar datos resumidos.
- Presentación de los segmentadores como filtros visuales interactivos, destacando su utilidad para explorar categorías o intervalos específicos, y la posibilidad de vincularlos a varias tablas o gráficos dinámicos al mismo tiempo.

- Introducción a los gráficos combinados, con ejemplos prácticos que incluyan la superposición de barras y líneas para comparar valores y tendencias, y una descripción de cómo añadir líneas de tendencia para destacar patrones y proyecciones.
- Revisión del formato condicional aplicado a tablas y gráficos dinámicos, con ejemplos de reglas para resaltar valores relevantes, patrones o errores, y su capacidad de actualizarse dinámicamente según los cambios en los datos
- **Dashboards.**
- Tablero interactivo: tablas y gráficos dinámicos, para consolidar y visualizar información clave. Explicación del concepto de dashboard como una representación visual de indicadores clave y datos esenciales para la toma de decisiones. Desarrollo de un diseño efectivo que integre gráficos dinámicos, tablas resumen y segmentadores para facilitar la interacción con los datos. Descripción de la conexión entre tablas y gráficos dinámicos mediante segmentadores y filtros para sincronizar las visualizaciones. Selección y configuración de indicadores clave de desempeño (KPIs) que se adapten a los objetivos del análisis, como ventas, costos o márgenes.
- Formato y diseño del dashboard para garantizar claridad y estética, utilizando colores, estilos y distribución visual efectiva. Ejemplo práctico de un dashboard que combine un análisis de ventas por región, un gráfico de tendencias de ingresos y un filtro interactivo para seleccionar períodos específicos
- **Tablas Dinámicas en Google Sheets.**
- Crear y configurar tablas dinámicas en Google Sheets, utilizando datos simples o complejos para análisis interactivos. Descripción del proceso para aplicar filtros, organizar campos y personalizar el diseño de tablas dinámicas en este entorno.
- Presentación de la integración con Google Forms, detallando cómo los datos recopilados automáticamente a través de formularios pueden alimentarse directamente en Google Sheets para su análisis.
- Ejemplo práctico de la combinación de datos recolectados con Forms y su posterior procesamiento en tablas dinámicas, con filtros y segmentaciones para generar reportes rápidos y efectivos. Exploración de las ventajas de trabajar en Google Sheets para compartir dashboards en tiempo real con equipos o clientes.
- **Migraciones de Datos.**
- Introducción a los procesos de transferencia de información entre diferentes plataformas utilizando Excel como herramienta principal.
- Incompatibilidades comunes entre Excel y Google Sheets, como formatos no admitidos, funciones exclusivas de cada plataforma y limitaciones en el tamaño de los archivos. Descripción de métodos para solucionar problemas de formato, incluyendo la conversión de columnas de texto a números, fechas y otros tipos de datos compatibles. Presentación de las herramientas de limpieza y validación de datos para asegurar la integridad de la información durante el proceso de migración.

- Uso de atajos de teclado en Excel para acelerar tareas comunes, como selección, edición y navegación dentro de las hojas de cálculo, optimizando el tiempo en procesos repetitivos.
- Ejemplo práctico de migración, resaltando los pasos para exportar datos desde Excel, convertirlos a formatos compatibles y adaptarlos para ser utilizados en Google Sheets
- **Ejercicio: Sistema de Inventory.**
 - creación de un sistema de inventario sencillo utilizando las herramientas de Excel, enfocándose en el seguimiento de productos, cantidades y valores. Definición de las tablas de datos, donde se incluirán columnas como nombre del producto, código, cantidad disponible, precio unitario y valor total.
 - Fórmulas y Funciones de agregación, rastreo y Formateo.
 - Uso de tablas dinámicas para resumir la información del inventario por categorías o marcas, permitiendo un análisis rápido de los productos más vendidos, el valor total en inventario, y las existencias mínimas.
 - Implementación de formato condicional para resaltar productos con bajo stock o aquellos que superan un valor de inventario determinado. Integración de gráficos dinámicos para representar la distribución del inventario por categorías y su evolución en el tiempo.
 - Aplicación de segmentadores para permitir la filtración de datos, como productos específicos o rangos de fechas.
- **Ejercicio: Evaluación del Personal en Google Sheets.**
 - Desarrollo de un sistema de evaluación del personal utilizando Google Sheets, donde se incluyen columnas para el nombre del empleado, departamento, criterios de evaluación (como desempeño, puntualidad, trabajo en equipo), y puntuaciones correspondientes. Se aplicarán fórmulas básicas para calcular el promedio de las evaluaciones por empleado y por departamento.
 - Uso de tablas dinámicas en Google Sheets para resumir y analizar los resultados de las evaluaciones por criterios, destacando las áreas de fortaleza y las que requieren mejora.
 - Implementación de formato condicional para resaltar empleados con calificaciones sobresalientes o que necesiten atención en áreas específicas.
 - Integración de gráficos dinámicos para visualizar el desempeño general de los empleados o el rendimiento de cada departamento a lo largo del tiempo. Configuración de segmentadores para filtrar los datos por fechas o departamentos, permitiendo un análisis detallado y personalizado.
- **Ejercicio: Proyección de Ventas.**
 - Creación de un modelo de proyección de ventas utilizando Excel, donde se incluyen datos históricos de ventas, objetivos mensuales y variables como precios, unidades vendidas y tendencias estacionales.
 - Aplicación de funciones como PROMEDIO, TENDENCIA y CRECIMIENTO para estimar las ventas futuras basadas en los datos históricos.

- Creación de tablas dinámicas para analizar las ventas por región, producto o mes, permitiendo identificar patrones de comportamiento y áreas de mejora.

- Uso de gráficos de líneas y columnas para representar visualmente las proyecciones y comparar las ventas proyectadas con los datos reales.

- Implementación de formato condicional para resaltar las variaciones significativas entre las ventas proyectadas y las ventas reales, así como los objetivos alcanzados.

- Aplicación de segmentadores para filtrar la proyección por diferentes criterios, como producto, año o región. Finalización con la interpretación de los resultados y ajustes en las proyecciones en función de los análisis obtenidos

- **Repaso:**

- La clase de repaso se enfocará en revisar los conceptos clave aprendidos durante las clases anteriores de Excel Avanzado.

- Se realizará una recapitulación de las herramientas y funciones más importantes, resolviendo dudas y reforzando los conocimientos mediante ejercicios prácticos.

- Los estudiantes tendrán la oportunidad de aplicar lo aprendido en casos reales y recibir retroalimentación personalizada para afianzar su comprensión y destreza en el uso de Excel.

- **Evaluación.**

- La evaluación consistirá en un ejercicio práctico donde los estudiantes deberán aplicar los conceptos y herramientas aprendidas durante el curso de Excel Avanzado.

- Se evaluará su capacidad para crear y gestionar hojas de cálculo, utilizar fórmulas y funciones adecuadas, aplicar formato y generar poderosos Dashboards

3. Power Query

- **Qué es Power Query:**

- Herramienta de transformación de datos que permite importar, limpiar, transformar y combinar datos de diversas fuentes, para luego cargarlos en Excel

- Conexiones a diversas fuentes de datos, como bases de datos, archivos de texto, hojas de cálculo, web, APIs, entre otros.

- Realización de tareas comunes como filtrar, ordenar, agrupar, dividir columnas, eliminar duplicados y realizar transformaciones complejas.

- Automatización de procesos, ya que una vez definidas las transformaciones

- La ventaja de Power relativa a su capacidad para manejar grandes volúmenes de datos de manera eficiente.

- **Conexiones Básicas:**

- Conexión con fuentes dentro de Excel. Importación de tablas completas o rangos específicos desde estas fuentes. Selección y carga de los datos en Power Query para su posterior transformación. Aplicación de filtros durante la importación para reducir la cantidad de datos importados.

- Renombrado de columnas y ajuste de tipos de datos según sea necesario.

- Visualización en tiempo real de la vista previa de los datos para realizar ajustes inmediatos.
- Edición y modificación de consultas sobre las tablas importadas para realizar transformaciones adicionales si es necesario
- Conexión y carga de archivos CSV y TXT en Power Query. Importación de datos desde archivos de texto delimitados por comas, tabuladores u otros caracteres.
- Ajuste de delimitadores según el formato del archivo. Identificación y manejo de encabezados en los archivos de texto. Conversión de datos en texto a tipos adecuados (números, fechas).
- Eliminación de filas vacías o innecesarias durante la importación. Manejo de caracteres especiales y codificación de texto (UTF-8, ANSI). Aplicación de transformaciones básicas como separación de columnas, cambio de tipos de datos y eliminación de duplicados.
- Uso de las opciones de vista previa para validar el contenido antes de cargar los datos en el modelo. Organización de los datos en Power Query para asegurar que la estructura sea adecuada para el análisis posterior.
- **Conexión con carpetas**
- Conexión con Carpetas en el dispositivo desde Power Query. Errores comunes. Ejemplo con datos diarios
- Consolidación de datos. Actualización. Coincidencia de columnas y formatos. Nombre de Hojas. Archivo ejemplo
- **Conexión con Google Sheets:**
- Conexión a hojas de cálculo de Google Sheets desde Power Query. Autenticación y permisos necesarios para acceder a los datos de Google Sheets.
- Importación de hojas de cálculo específicas desde una cuenta de Google. Selección de rangos o tablas dentro de la hoja de cálculo para su importación. Manejo de datos en Google Sheets con formato adecuado para su carga en Power Query. Aplicación de filtros o transformaciones durante la carga de los datos.
- Conexión dinámica para actualizar los datos cuando se realicen cambios en el archivo de Google Sheets. Configuración de consultas para manejar múltiples hojas dentro de un mismo documento de Google Sheets. Uso de Power Query para procesar y combinar datos de Google Sheets con otras fuentes de datos.
- **Limpieza de Datos 1**
- Eliminación de filas y columnas vacías: eliminación de espacios en blanco innecesarios que pueden afectar los análisis.
- Filtrar datos duplicados: identificación y eliminación de valores duplicados en las tablas.
- Corrección de errores comunes: manejo de celdas con errores (#N/A, #DIV/0!, etc.) y su reemplazo o eliminación.
- Conversión de tipos de datos: conversión de texto a números, fechas o valores booleanos según sea necesario.

- Reemplazo de valores: sustitución de valores incorrectos o inconsistentes por los correctos utilizando la función "Reemplazar".
- Recorte de espacios adicionales: eliminación de espacios al inicio o final de los textos con la función ESPACIOS.
- Unificación de formatos: homogeneización de formatos de datos, como fechas, para asegurar consistencia.
- **Limpieza de Datos 2**
- Separación de columnas: uso de la función "Texto en columnas" para dividir información contenida en una sola celda (como nombres, direcciones, etc.).
- Combinar columnas: fusión de varias columnas de datos en una sola usando funciones como CONCATENAR o &.
- Eliminación de caracteres no deseados: eliminación de caracteres especiales o no alfanuméricos que interfieren con el análisis.
- Normalización de texto: convertir todo el texto a mayúsculas, minúsculas o formato de título para mantener consistencia en la presentación de datos.
- Validación de datos: uso de reglas de validación para asegurar que los datos introducidos en las celdas cumplan con un formato o rango esperado.
- Manejo de celdas vacías: sustitución de celdas vacías por valores predeterminados o la eliminación de filas que contengan celdas vacías críticas.
- **Limpieza de Datos 3**
- Detección de valores atípicos: identificación y manejo de valores atípicos que pueden distorsionar el análisis.
- Manejo de datos faltantes: estrategias para gestionar celdas con valores faltantes, como la imputación o eliminación de datos.
- Consolidación de datos: uso de funciones para consolidar información dispersa en varias filas o columnas.
- Transformación de datos: cambio de la estructura de los datos para facilitar el análisis, como transponer filas y columnas.
- Limpieza avanzada con Power Query: uso de Power Query para realizar transformaciones más complejas, como combinar fuentes de datos, eliminar registros duplicados y aplicar pasos de limpieza de forma automática.
- Revisión de consistencia y coherencia: verificación de que los datos sean coherentes y consistentes en todo el conjunto de datos (fechas, formatos numéricos, etc.).
- **Anexar y Combinar Consultas:**
- Anexar Consultas: proceso de unir filas de diferentes tablas o consultas que tienen la misma estructura.
- Combinación de consultas (Merge): unión de tablas mediante una clave común, similar a una operación de JOIN en bases de datos.

- Anexar en Power Query: utilizar la opción de "Anexar consultas" para agregar los datos de una tabla debajo de otra, creando un solo conjunto de datos.
- Combinación de varias tablas: unión de múltiples tablas provenientes de diferentes fuentes o de diferentes hojas dentro de un mismo archivo.
- Tipos de combinación (Inner Join, Left Join, etc.): explicación de los diferentes tipos de uniones disponibles, como combinación interna, externa izquierda, externa derecha y completa.
- Eliminación de duplicados después de combinar: cómo limpiar datos duplicados después de realizar la combinación de tablas.
- Manejo de columnas repetidas: resolución de conflictos cuando hay columnas con el mismo nombre en las tablas que se están combinando.
- Transformación de datos después de la combinación: realización de cambios y ajustes después de haber combinado las consultas, como renombrar columnas o ajustar tipos de datos.
- **Agregar Columna Condicional:**
- Uso de la función "Columna Condicional": creación de una nueva columna con valores asignados según condiciones definidas.
- Estructura básica de una columna condicional: definición de condiciones usando el formato "Si... Entonces" (If... Then) para establecer qué valores se deben asignar en función de los datos en otras columnas.
- Múltiples condiciones: combinación de múltiples condiciones utilizando operadores lógicos (Y, O) para crear reglas más complejas.
- Asignación de valores personalizados: definir qué valores se deben asignar cuando se cumplan o no las condiciones, como texto específico, valores numéricos o resultados de otras columnas.
- Manejo de valores nulos o vacíos: tratamiento de celdas vacías o con valores nulos en la columna condicional, utilizando valores predeterminados o condiciones adicionales.
- **Columnas Personalizadas I**
- Creación de Columnas Personalizadas: cómo agregar nuevas columnas utilizando fórmulas personalizadas en Power Query.
- Sintaxis Básica de Lenguaje M: introducción a la sintaxis del lenguaje M, incluyendo operaciones básicas como asignación de valores, operadores aritméticos y lógicos.
- Uso de la función "Add Column": creación de columnas personalizadas mediante la interfaz de Power Query.
- Fórmulas simples: aplicación de fórmulas simples en columnas personalizadas, como operaciones matemáticas entre columnas existentes.
- Estructura de las fórmulas M: comprensión de la estructura de las fórmulas M, incluyendo el uso de funciones predefinidas.

- Ejemplo práctico de una columna personalizada básica: realizar una operación entre dos columnas (por ejemplo, suma o resta).

- **Columnas Personalizadas II**

- Funciones predefinidas de M en columnas personalizadas: uso de funciones integradas como Text.Contains, Date.AddDays, Number.Round, entre otras.
- Manejo de cadenas de texto: manipulación de textos con funciones como Text.Length, Text.Upper, Text.ToLower, Text.Substring, etc.
- Manejo de fechas: uso de funciones de fecha como Date.Year, Date.Month, Date.Day, para extraer partes de una fecha o realizar cálculos.
- Operadores lógicos y condicionales: implementación de condiciones en el lenguaje M, como if...then y operadores lógicos and, or, para crear reglas más complejas.
- Uso de la función "if...then...else": implementación de condiciones múltiples con estructuras de control.

- **Columnas Personalizadas III**

- Optimización de fórmulas en M: mejores prácticas para escribir código eficiente y evitar redundancias.
- Depuración de consultas en M: cómo identificar y corregir errores en el lenguaje M utilizando herramientas como el paso de "Vista previa" y "Avanzado".
- Manipulación de errores en M: técnicas para manejar y depurar errores dentro de las fórmulas personalizadas.

- **Ejercicio: Migración de datos**

- Concepto de Migración de Datos. Potencial de Power query. Pasos programados. Archivos Base.
- Extracción y limpieza de datos. Exportar datos. Archivos Resultantes.

- **Repaso:**

- La clase de repaso se enfocará en revisar los conceptos clave aprendidos durante las clases anteriores.
- Se realizará una recapitulación de las herramientas más importantes, resolviendo dudas y reforzando los conocimientos mediante ejercicios prácticos.
- Los estudiantes tendrán la oportunidad de aplicar lo aprendido en casos reales y recibir retroalimentación personalizada para afianzar su comprensión y destreza en el uso de Excel.

- **Evaluación.**

- La evaluación consistirá en un ejercicio práctico donde los estudiantes deberán aplicar los conceptos y herramientas aprendidas durante el curso de Power Query
- Se evaluará su capacidad para gestionar datos, extraerlos, limpiarlos y presentarlos

- 4. Power Pivot

- **Repaso de Tablas Dinámicas**

- Concepto de Tablas Dinámicas: recordatorio de qué son las tablas dinámicas y su propósito como herramienta para resumir y analizar grandes cantidades de datos de manera interactiva.
- Estructura de una Tabla Dinámica: revisión de los elementos principales: campos de fila, columnas, valores y filtros.
- Creación de Tablas Dinámicas: pasos para crear una tabla dinámica desde un rango de datos o una tabla en Excel.
- Agrupamiento de Datos: agrupación de valores en rangos o categorías personalizadas dentro de la tabla dinámica.
- Mostrar Valores como: uso de opciones avanzadas para mostrar los datos como porcentaje del total, diferencia con respecto a otros valores, índices, etc.
- Segmentadores y Barras de Tiempo: integración de segmentadores para filtros visuales y uso de barras de tiempo para análisis de series temporales.
- Formato de Tablas Dinámicas: personalización de la apariencia de la tabla dinámica, incluyendo estilos, diseño de informe y opciones de formato condicional.
- Uso de Datos Externos: conexión de tablas dinámicas a fuentes externas simples, como archivos CSV o bases de datos ligeras.
- Limitaciones de las Tablas Dinámicas: introducción a las restricciones de las tablas dinámicas tradicionales, como el manejo limitado de datos relacionados o la falta de medidas personalizadas.
- Preparación para Power Pivot: exploración de los desafíos al trabajar con múltiples tablas en Excel y cómo Power Pivot aborda estas limitaciones mediante modelos de datos y relaciones
- **Aplicación de Power Query**
- Repaso de Power Query: breve revisión de sus capacidades principales para transformar y limpiar datos antes de usarlos en análisis avanzados.
- Integración de Power Query y Power Pivot: explicación del flujo de trabajo entre ambas herramientas, donde Power Query actúa como etapa de preparación de datos y Power Pivot como herramienta de análisis.
- Importación de datos desde Power Query a Power Pivot: cómo cargar datos procesados en Power Query directamente en el modelo de datos de Power Pivot.
- Limpieza y transformación previa con Power Query: ventajas de preparar los datos en Power Query para reducir errores y simplificar la creación del modelo en Power Pivot.
- Consolidación de datos en Power Query para Power Pivot: ejemplos prácticos de combinar, anexar y normalizar datos antes de integrarlos en Power Pivot.
- Uso de datos relacionales: cómo Power Query permite estructurar datos en formato de tabla relacional para facilitar la creación de relaciones en Power Pivot.
- Manejo de grandes volúmenes de datos: optimización de los datos en Power Query para trabajar eficientemente en Power Pivot con conjuntos de datos extensos.

- Automatización de procesos: creación de flujos de trabajo automatizados en Power Query para actualizar y sincronizar datos con Power Pivot en tiempo real.
- Preparación para el análisis avanzado en Power Pivot: cómo Power Query simplifica la creación de columnas calculadas, medidas y jerarquías en Power Pivot al entregar datos consistentes.

- **Modelado de Datos**

- Concepto de Modelo de Datos: introducción al modelo de datos como estructura relacional que organiza tablas y facilita análisis complejos.
- Relaciones entre tablas: definición de relaciones entre tablas mediante claves primarias y foráneas, y su importancia en el modelo de datos.
- Tipos de relaciones: análisis de relaciones uno a uno, uno a muchos y muchos a muchos, con ejemplos prácticos.
- Diagramas de relaciones: uso del diagrama de modelo de datos en Power Pivot para visualizar y gestionar las conexiones entre tablas.
- Cardinalidad de relaciones: explicación del flujo de datos según la dirección de la relación y cómo afecta los cálculos en el modelo.
- Relaciones activas e inactivas: diferencias entre relaciones activas e inactivas y cómo activar relaciones específicas para cálculos particulares.
- Tablas de Dimensión y Hechos: introducción a la estructura de datos en modelos analíticos, diferenciando tablas de hechos (transacciones) y de dimensiones (atributos).
- Problemas comunes en el modelado de datos: identificación y resolución de errores comunes como datos duplicados, relaciones incorrectas o falta de claves únicas.
- Optimización del modelo de datos: mejores prácticas para reducir la complejidad y mejorar el rendimiento del modelo, como la eliminación de columnas innecesarias o el uso de tablas resumidas.

- **Tablas Pivot**

- Definición y utilidad de las Tablas Pivot: explicación de las tablas pivot como herramienta avanzada para resumir, analizar y explorar datos en el modelo de datos de Power Pivot.
- Diferencias entre Tablas Dinámicas y Tablas Pivot: comparación de las capacidades adicionales de las Tablas Pivot, como el manejo de datos relacionados y medidas personalizadas.
- Creación de Tablas Pivot desde el Modelo de Datos: pasos para insertar una tabla pivot conectada al modelo de datos en Power Pivot.
- Agrupación y jerarquías: cómo agrupar datos y utilizar jerarquías creadas en el modelo de datos para un análisis más estructurado.
- Segmentadores y Filtros: integración de segmentadores y filtros para crear análisis dinámicos e interactivos.

- **Dashboards en Power Pivot**

- Concepto de Dashboard: definición de un dashboard como una representación visual interactiva que consolida datos clave para facilitar la toma de decisiones.
- Elementos de un Dashboard: introducción a los componentes esenciales como tablas pivot, gráficos dinámicos, segmentadores, barras de tiempo y texto informativo.
- Diseño de Dashboards: principios básicos de diseño visual, como organización, simplicidad, consistencia y enfoque en la interpretación rápida de los datos.
- Creación de Gráficos Dinámicos desde Power Pivot: pasos para insertar gráficos interactivos basados en el modelo de datos, como gráficos de barras, columnas, líneas y combinados.
- Uso de Segmentadores y Barras de Tiempo: configuración de segmentadores y barras de tiempo para filtrar y analizar información en tiempo real.
- Personalización de Formatos y Estilos: aplicación de formatos personalizados a tablas y gráficos para mejorar la presentación visual.
- Vinculación entre elementos del Dashboard: cómo conectar tablas pivot, gráficos y segmentadores para garantizar una interacción fluida entre los componentes.

• **Introducción a DAX**

- Definición de DAX: introducción al lenguaje de expresiones de análisis de datos (Data Analysis Expressions) como herramienta para realizar cálculos avanzados en Power Pivot, Power BI y otras plataformas de análisis de Microsoft.
- Elementos básicos de DAX: estructura de fórmulas en DAX, incluyendo funciones, operadores y sintaxis general.
- Tipos de datos en DAX: revisión de los tipos de datos compatibles, como texto, números, fechas y valores booleanos.
- Operadores en DAX: explicación de los operadores básicos (matemáticos, lógicos y relacionales) y su aplicación en cálculos.
- Funciones Básicas de DAX: introducción a funciones comunes como SUM, AVERAGE, MIN, MAX y COUNT, utilizadas para cálculos simples.
- Uso del Contexto en DAX: introducción al concepto de contexto de fila y contexto de filtro, fundamentales para entender cómo se aplican los cálculos en DAX.

• **Medidas en DAX**

- Concepto de Medidas: definición de medidas como cálculos dinámicos realizados a nivel de agregación en el modelo de datos.
- Diferencias entre Columnas Calculadas y Medidas: cuándo utilizar medidas para análisis dinámicos y cómo estas se adaptan al contexto del filtro.
- Creación de Medidas: pasos básicos para crear medidas en Power Pivot utilizando el editor de DAX.
- Funciones de Agregación en Medidas: uso de funciones como SUM, AVERAGE, MIN, MAX y COUNT para realizar cálculos fundamentales.

- Formato de Resultados: configuración de formatos numéricos, porcentajes y decimales en medidas para una presentación adecuada de los datos.
- Errores comunes en Medidas: cómo identificar y resolver problemas típicos, como resultados inesperados debido a mal manejo del contexto.
- Optimización de Medidas: estrategias para mejorar el rendimiento y la claridad de las medidas en modelos grandes o complejos.
- **Funciones Condicionales en DAX**
- Definición de Funciones Condicionales: explicación de cómo las funciones condicionales permiten ejecutar cálculos basados en condiciones específicas en el modelo de datos.
- IF: uso de la función IF para evaluar condiciones y devolver resultados diferentes según si la condición es verdadera o falsa.
- SWITCH: introducción a SWITCH como una alternativa más eficiente para manejar múltiples condiciones o casos.
- IFERROR: manejo de errores en cálculos utilizando IFERROR para devolver un valor alternativo cuando ocurre un error.
- Combinación de Condiciones: cómo anidar funciones IF y SWITCH para manejar escenarios complejos con múltiples criterios.
- Operadores Relacionales y Lógicos: uso de operadores como $>$, $<$, $=$, $&&$ (AND) y $\|$ (OR) para construir condiciones más avanzadas.
- **Función CALCULATE en DAX**
- Definición de CALCULATE: explicación de CALCULATE como una función clave en DAX que permite modificar el contexto de filtro para realizar cálculos personalizados.
- Sintaxis Básica de CALCULATE: estructura general de la función y cómo combinarla con expresiones y filtros.
- Uso de CALCULATE con Filtros Simples: cómo aplicar condiciones simples para limitar los datos utilizados en una medida o columna calculada.
- Múltiples Filtros en CALCULATE: combinación de varios filtros utilizando comas dentro de la función y su impacto en el contexto.
- **Funciones de Fechas en DAX**
- Importancia de las Funciones de Fechas en DAX: explicación de cómo estas funciones facilitan el análisis temporal y la generación de informes periódicos.
- CALENDAR y CALENDARAUTO: creación de tablas de calendario dinámicas para habilitar cálculos basados en fechas.
- TODAY y NOW: funciones para obtener la fecha actual y la fecha y hora actuales, respectivamente, y su uso en cálculos.
- YEAR, MONTH, DAY: extracción de componentes específicos de una fecha, como año, mes o día.
- DATEDIFF: cálculo de la diferencia entre dos fechas en unidades como días, meses o años.

- EDATE y EOMONTH: obtención de fechas específicas con base en desplazamientos mensuales, como inicios o finales de mes.
- STARTOFMONTH, ENDOFMONTH, STARTOFTYEAR, ENDOFYEAR: funciones para determinar el inicio y fin de períodos específicos.
- Funciones de Inteligencia de Tiempo: introducción a funciones avanzadas como TOTALYTD, SAMEPERIODLASTYEAR y DATEADD para realizar análisis comparativos entre períodos.
- Cálculo de Períodos Personalizados: uso de funciones como DATESBETWEEN para definir rangos de fechas personalizados para cálculos específicos.
- Fechas y Contexto de Filtro: cómo las funciones de fecha interactúan con los filtros aplicados en Tablas Pivot y Dashboards.
- Errores Comunes con Fechas: soluciones para problemas relacionados con formatos de fecha, zonas horarias o datos incompletos.
- **KPI en Power Pivot**
 - Definición de KPI (Indicador Clave de Desempeño): explicación de qué es un KPI y su importancia para medir el rendimiento y los objetivos en un análisis de datos.
 - Componentes de un KPI: definición de los tres componentes principales de un KPI: objetivo (valor esperado), valor real (valor calculado) y umbrales (límites de desempeño).
 - Creación de KPIs en Power Pivot: cómo configurar KPIs dentro de Power Pivot utilizando medidas y establecer relaciones entre los valores reales y los objetivos.
 - Establecimiento de Umbrales: cómo definir los umbrales de rendimiento, como el límite inferior, el objetivo y el límite superior, que determinan el color o la señalización del KPI.
 - Uso de Medidas para KPIs: creación de medidas personalizadas en DAX para calcular el rendimiento real de un indicador y compararlo con los objetivos.
 - Formato de KPI: aplicación de formato condicional para cambiar la apariencia visual del KPI en función de su desempeño (por ejemplo, verde para cumplimiento, rojo para no cumplimiento).
- **Power Maps**
 - Introducción a Power Maps: definición de Power Maps como una herramienta visual en Excel para crear mapas interactivos a partir de datos geográficos y de ubicación.
 - Tipos de Datos Geoespaciales: explicación de los datos necesarios para usar Power Maps, como coordenadas geográficas (latitud y longitud), direcciones, códigos postales, países, etc.
 - Acceso a Power Maps: cómo acceder y activar la funcionalidad de Power Maps en Excel y sus versiones disponibles.
 - Creación de un Mapa Básico: pasos para crear un mapa interactivo básico usando datos de localización, como crear una representación geográfica de ventas por región o país.

- Visualización de Datos en el Mapa: cómo representar diferentes tipos de datos en el mapa, utilizando colores, tamaños y representaciones de datos como burbujas, columnas y heat maps (mapas de calor).
- Exportación de Power Maps: cómo exportar un mapa a PowerPoint o como imagen estática para incluir en informes.

- **Ejercicio Monitoreo de ventas**

- Extracción de datos: Uso de Power Query para extraer datos de diferentes fuentes. Limpieza de datos. Creación de conexión.
- Modelado de datos: Creación de relaciones. Tablas Pivot que sean necesarias para el modelo. Uso de DAX para crear Medidas Adecuadas.
- Exposición de Información. Uso de Tablas Dinámicas, Gráficos Dinámicos, Segmentadores de datos y Formatos Condicionales

- **Repaso:**

- La clase de repaso se enfocará en revisar los conceptos clave aprendidos durante las clases anteriores.
- Se realizará una recapitulación de las herramientas más importantes, resolviendo dudas y reforzando los conocimientos mediante ejercicios prácticos.
- Los estudiantes tendrán la oportunidad de aplicar lo aprendido en casos reales y recibir retroalimentación personalizada para afianzar su comprensión y destreza en el uso de Power Query y Power Pivot.

- **Evaluación.**

- La evaluación consistirá en un ejercicio práctico donde los estudiantes deberán aplicar los conceptos y herramientas aprendidas durante el curso de Power Pivot.
- Se evaluará su capacidad para gestionar modelos de datos y crear poderosos cuadros de mando.

5. Visual Basic Básico

- **Macros en Excel**

- Introducción a las Macros: definición de las macros como una herramienta para automatizar tareas repetitivas en Excel mediante la grabación de acciones.
- Grabación de Macros: explicación sobre cómo usar el grabador de macros para registrar una serie de pasos y crear una macro básica sin necesidad de escribir código.
- Estructura de una Macro: descripción de cómo Excel organiza las macros, incluyendo los módulos y cómo acceder al editor de Visual Basic (VBE).
- Sintaxis Básica de Macros: introducción a los elementos básicos del código generado por el grabador de macros, como la estructura de comandos y la asignación de variables.
- Ejecuciones de Macros: Ejecución común. Ejecución por Cinta de Opciones. Ejecución con Atajos. Cómo asignar una macro a un botón en la hoja de trabajo para ejecutarla con un solo clic, mejorando la accesibilidad.

- Errores Comunes en Macros: identificación de errores comunes al crear o ejecutar macros, como errores de sintaxis o referencias incorrectas a celdas o rangos.

- **Referencias Absolutas y Relativas**

- Relatividad de Macros. Referencias absolutas y relativas. Diferencias. Cambios en Grabaciones. Escenarios posibles para la aplicación de Referencias Relativas. Errores comunes que suceden al utilizar Relatividad.
- Ejemplos varios de Macros. Navegación por Hojas. Aplicaciones de Formatos. Ventajas en Referencias. Aplicación de Formatos con macros. Seguridad de Macros. Aspectos a considerar en el uso de Macros.
- Mejores Prácticas en Macros: recomendaciones sobre cómo estructurar y escribir macros eficientemente para mejorar su rendimiento y mantenimiento.

- **Introducción de Datos con Macros**

- Concepto de Introducción de Datos en Excel con Macros: Explicación de cómo las macros pueden automatizar la entrada de datos en celdas específicas de una hoja de Excel. Beneficios de usar macros para la introducción de datos repetitivos y cómo simplifican tareas complejas de entrada manual.
- Consideraciones para entradas de datos: Validaciones, Funciones base. Desplazamiento por Hojas. Visibilidad de hojas. Otras Restricciones. Flujo de información en la ejecución. Detección de errores y posibles soluciones.

- **Entorno de desarrollo de VBA.**

- Macros Híbridas: Concepto de Macros Híbridas. Razones para utilizarlas. Restricciones de las macros. Corrección de Macros para evitar grabarlas nuevamente.
- Introducción a la Ventana de Visual Basic (VBE). Definición: la Ventana de Visual Basic para Aplicaciones (VBE) es el entorno donde se escribe y edita el código VBA (Visual Basic for Applications) en Excel. Importancia de la VBE como herramienta para crear automatizaciones, personalizaciones y soluciones avanzadas en Excel.
- Acceso a la VBE. Métodos para abrir la VBE: Tecla de acceso rápido: Presionar Alt + F11. Desde la cinta de opciones: Ir a la pestaña Desarrollador y hacer clic en Visual Basic.
- Componentes de la Ventana de Visual Basic. Explorador de Proyectos: Muestra todos los libros, hojas y formularios abiertos en el proyecto VBA. Permite navegar entre objetos como ThisWorkbook, Sheet1, Module1, etc. Ventana de Código: Área principal donde se escriben y editan los procedimientos VBA. Cada objeto del proyecto tiene su propia ventana de código. Ventana Inmediata: Permite ejecutar comandos y ver resultados inmediatos durante la depuración del código. Ventana de Propiedades: Muestra y permite modificar las propiedades de los objetos seleccionados (como Workbooks, Sheets, UserForms, etc.). Barra de Herramientas: Contiene botones para ejecutar código, depurar, y acceder a otras funciones como compilar o guardar el proyecto.
- Navegación en la VBE. Uso del Explorador de Proyectos para seleccionar módulos y objetos dentro del proyecto. Cómo abrir y cerrar ventanas de código. Acceso a formularios

de usuario (UserForms) y módulos de código. Cómo cambiar entre diferentes objetos y módulos con facilidad.

- Depuración y Errores en la VBE. Uso de la Ventana Inmediata para ejecutar código y evaluar expresiones mientras se depura. Uso de los puntos de interrupción para detener la ejecución y examinar el flujo del programa. Cómo leer y comprender los mensajes de error que se muestran en la Ventana de Depuración.

- **Eventos en Libros y Hojas**

- Introducción a los Eventos en Visual Basic. Definición de eventos: acciones o sucesos que desencadenan procedimientos. Importancia de los eventos en la automatización de tareas en Excel. Diferencia entre eventos de nivel de libro y de nivel de hoja.
- Objetos Clave: Workbook y Worksheet. Descripción del objeto Workbook (Libro). Descripción del objeto Worksheet (Hoja). Propiedades y métodos principales de ambos objetos: Workbook.Name: nombre del libro. Workbook.Sheets: colección de hojas del libro. Worksheet.Cells y Range: trabajar con celdas específicas.
- Eventos del Objeto Workbook. Open: acciones al abrir el libro, como cargar configuraciones personalizadas o inicializar datos. BeforeClose: tareas a realizar antes de cerrar el libro, como guardar cambios automáticamente. BeforeSave: validaciones o ajustes antes de guardar el libro. SheetChange: detectar cambios en cualquier hoja del libro. SheetActivate: ejecutar acciones al activar una hoja específica.
- Eventos del Objeto Worksheet. Change: detectar cambios en celdas específicas de una hoja. SelectionChange: ejecutar código cuando cambia la selección en la hoja. Activate y Deactivate: acciones al activar o desactivar una hoja. BeforeDoubleClick: personalizar el comportamiento al hacer doble clic en una celda. BeforeRightClick: redefinir opciones al hacer clic derecho en una celda.
- Propiedades de los Objetos en Relación con los Eventos. Uso de la propiedad Target en eventos como Change y SelectionChange para identificar las celdas afectadas. Uso de propiedades como Row y Column para trabajar con rangos específicos en eventos. Métodos comunes relacionados con los eventos, como Undo para revertir cambios en un Change.

- **Elementos de Visual Basic**

- Introducción a los Elementos de Control en Visual Basic. Definición de controles en el entorno de Visual Basic para Excel. Propósito de los elementos de control en la creación de interfaces de usuario interactivas. Descripción del entorno UserForm y su importancia como contenedor de controles.
- TextBox (Cuadro de Texto). Uso principal: ingreso y visualización de texto o datos por parte del usuario. Propiedades clave: Text: contenido del cuadro de texto. Enabled y Locked: control de la edición. MaxLength: límite de caracteres permitidos. Eventos asociados: Change: detectar cuando el contenido cambia. KeyPress: manejar entradas específicas, como solo permitir números.

- CommandButton (Botón de Comando). Uso principal: ejecutar acciones o procedimientos al ser presionado. Propiedades clave: Caption: texto visible en el botón. Enabled: activar o desactivar el botón. Eventos asociados: Click: definir la acción que ocurre al hacer clic en el botón.
- ListBox (Cuadro de Lista). Uso principal: mostrar una lista de opciones que el usuario puede seleccionar. Propiedades clave: RowSource: origen de datos para llenar el cuadro de lista. MultiSelect: permitir selección simple o múltiple. List: acceder o manipular los elementos de la lista. Eventos asociados: Click: determinar qué elemento ha sido seleccionado.
- ComboBox (Cuadro Combinado). Uso principal: combinación de un cuadro de lista y un cuadro de texto para selección o ingreso manual de datos. Propiedades clave: Style: configurar como lista desplegable o editable. BoundColumn: columna relacionada con el valor del cuadro. Diferencias clave entre ComboBox y ListBox.
- Label (Etiqueta). Uso principal: mostrar texto estático o dinámico como información para el usuario. Propiedades clave: Caption: texto a mostrar. Font: formato del texto.
- CheckBox (Casilla de Verificación). Uso principal: permitir al usuario seleccionar o deseleccionar una opción. Propiedades clave: Value: estado actual (True, False). Ejemplo de uso en formularios con múltiples opciones.
- OptionButton (Botón de Opción) Uso principal: seleccionar una opción dentro de un grupo mutuamente excluyente. Propiedades clave: GroupName: agrupar botones de opción para asegurar exclusividad.

• **Condicionales en VBA**

- Introducción a las Lógicas en Visual Basic. Explicación de la importancia de las estructuras condicionales en programación. Definición de decisiones lógicas y cómo controlan el flujo de ejecución en Visual Basic. Presentación de las palabras clave: If, Else, ElseIf, And, Or, y Select Case como herramientas clave.
- Estructura Básica de IF y ELSE. Sintaxis básica: Evaluación de una condición dentro de un bloque If. Uso de Else para manejar resultados alternativos cuando la condición es falsa. Ejemplo de aplicación práctica: verificar si un valor cumple un criterio específico.
- ELSEIF para Múltiples Condiciones. Uso de ElseIf para evaluar más de una condición en un solo bloque de decisión. Comparación con el uso de múltiples bloques If separados y cuándo optar por cada uno.

• **Condicionales complejos en VBA**

- Operadores Lógicos: AND y OR. AND: combinación de condiciones donde todas deben ser verdaderas para ejecutar un bloque de código. OR: combinación de condiciones donde basta que una sea verdadera para ejecutar el código. Uso práctico en decisiones complejas, como validar múltiples criterios.
- SELECT CASE: Alternativa al IF para Decisiones Múltiples. Introducción a Select Case como una forma más clara y eficiente de manejar múltiples opciones. Sintaxis básica:

Evaluación de una expresión única contra varios casos predefinidos. Uso de Case Else para manejar situaciones no especificadas. Comparación entre Select Case y ElseIf: Ventajas en legibilidad y rendimiento para grandes cantidades de opciones. Situaciones en las que cada estructura es más adecuada.

- **Box de Mensajes**

- Introducción a las Cajas de Diálogo en VBA. Concepto general de las cajas de diálogo. Diferencias entre MsgBox e InputBox. Aplicaciones comunes en automatización con Excel.
- Uso de MsgBox para Mostrar Mensajes. Sintaxis básica de MsgBox. Personalización del mensaje: Títulos de ventana. Tipos de botones como Sí/No o Aceptar/Cancelar. Iconos informativos como advertencia o confirmación.
- Captura de Respuestas con MsgBox. Uso de la función para registrar la acción del usuario. Variables para almacenar y procesar la respuesta obtenida. Diferentes opciones de flujo según la elección del usuario.
- Introducción a InputBox para Solicitar Datos al Usuario. Sintaxis básica de InputBox. Parámetros disponibles: Texto del mensaje. Título de la ventana. Valor predeterminado opcional.
- Validación de Datos con InputBox. Verificación de campos vacíos. Validación de formatos específicos, como valores numéricos. Uso de funciones de validación y manejo de errores básicos

- **Variables y constantes.**

- Introducción a las Variables y Constantes. Definición de variables como contenedores de datos temporales. Definición de constantes como valores fijos que no cambian durante la ejecución. Importancia de las variables y constantes en la programación estructurada.
- Declaración de Variables. Necesidad de declarar variables antes de usarlas. Tipos de declaración: explícita e implícita. Buenas prácticas en la declaración de variables.
- Tipos de Datos en Visual Basic. Clasificación de tipos de datos: Numéricos (Integer, Double, etc.). Texto (String). Booleanos (True/False). Otros tipos (Date, Object, etc.). Selección del tipo de dato adecuado para optimizar memoria y procesamiento.
- Alcance de las Variables. Variables locales: definición y uso dentro de procedimientos específicos. Variables globales: disponibilidad a lo largo de todo el proyecto. Diferencias entre variables públicas y privadas. Importancia del alcance para evitar conflictos o errores.
- Constantes en Visual Basic. Creación de constantes para valores que no cambian. Diferencias entre constantes y variables. Usos típicos, como almacenar configuraciones o valores repetidos.
- Inicialización y Asignación de Valores. Reglas para la inicialización de variables. Momentos de asignación: en la declaración o durante la ejecución. Buenas prácticas para nombrar variables y constantes.

- Gestión de Errores Relacionados con Variables. Problemas comunes: variables no inicializadas o uso de nombres ambiguos. Estrategias para evitar errores: Uso de la opción para exigir declaración explícita. Comentarios descriptivos en la asignación de valores. Herencias entre constantes, variables públicas y privadas.

• **Procedimientos en VBA**

- Introducción a los Procedimientos. Definición de un procedimiento como un bloque de código que realiza una tarea específica. Diferencias entre procedimientos y funciones en Visual Basic. Importancia de los procedimientos para modularizar el código y mejorar su legibilidad.
- Tipos de Procedimientos en Visual Basic. Procedimientos Sub: Realizan acciones pero no devuelven un valor. Procedimientos Function: Realizan acciones y devuelven un valor.
- Declaración de Procedimientos. Sintaxis básica para declarar procedimientos Sub y Function. Reglas para nombrar procedimientos: Nombres descriptivos y significativos. Evitar palabras reservadas. Uso de comentarios para documentar el propósito del procedimiento.
- Parámetros y Argumentos. Concepto de parámetros como valores que un procedimiento necesita para ejecutarse. Tipos de parámetros: ByVal: envía una copia del valor. ByRef: permite modificar el valor original. Uso de argumentos opcionales y valores predeterminados.
- Llamadas a Procedimientos. Cómo invocar un procedimiento desde otro procedimiento o desde un evento. Manejo de argumentos en las llamadas. Secuencia de ejecución cuando se utilizan múltiples procedimientos.
- Beneficios de los Procedimientos. Reutilización de código en diferentes partes del programa. Mejora en la organización y claridad del código. Reducción de errores al encapsular lógica en bloques específicos

• **Funciones en VBA**

- Introducción a las Funciones. Definición de una función como un bloque de código que realiza una tarea y devuelve un valor. Diferencias entre funciones y procedimientos Sub. Importancia de las funciones para estructurar el código y manejar cálculos complejos.
- Declaración de Funciones. Sintaxis básica para declarar una función: palabras clave, nombre, parámetros y tipo de retorno. Reglas para nombrar funciones: Nombres descriptivos que reflejen la tarea de la función. Uso de notación consistente en proyectos grandes. Uso de comentarios para documentar el propósito de la función.
- Tipos de Retorno en Funciones. Asignación del tipo de dato que devuelve una función: Numéricos (Integer, Double, etc.). Textuales (String). Booleanos (True/False). Otros tipos (Date, Object). Uso de variables internas para calcular y retornar el valor final.
- Parámetros y Argumentos. Uso de parámetros para pasar datos a una función. Tipos de parámetros: ByVal: evita cambios en el valor original. ByRef: permite modificar el valor original. Parámetros opcionales y valores predeterminados.

- Llamadas a Funciones. Cómo invocar funciones desde procedimientos y otras funciones. Uso del valor devuelto por la función en cálculos y decisiones lógicas. Integración de funciones con eventos en Excel, como botones o celdas.

- Beneficios de las Funciones. Modularización del código para dividir problemas complejos en tareas más simples. Mejora en la legibilidad y mantenimiento del programa. Reutilización de lógica en diferentes partes del proyecto.

- **Funciones de Excel en Visual Basic**

- Introducción al Objeto Application. Definición del objeto Application como intermedio para acceder a funcionalidades de Excel desde Visual Basic. Importancia de Application para extender las capacidades de las macros mediante funciones nativas de Excel.

- Acceso a Funciones de Excel. Uso de Application.WorksheetFunction para invocar funciones de hoja de cálculo en Visual Basic. Diferencia entre métodos de Visual Basic y funciones de Excel invocadas con Application.

- Estructura Básica para Llamar a una Función. Sintaxis: Application.WorksheetFunction.FunciónExcel(parámetros). Explicación de cómo pasar argumentos a las funciones según el tipo esperado.

- Ejemplos de Funciones Comunes Usadas con Application. Función SUM: uso para sumar rangos de valores dinámicos. Función VLOOKUP: búsqueda de valores en tablas de datos desde Visual Basic. Función COUNTIF: conteo basado en criterios específicos. Función IFERROR: manejo de errores en cálculos desde VBA.

- Manejo de Errores al Usar Application. Importancia de validar los datos antes de invocar funciones de Excel. Uso de control de errores en Visual Basic (On Error) para evitar interrupciones al llamar funciones.

- **Repaso:**

- La clase de repaso se enfocará en revisar los conceptos clave aprendidos durante las clases anteriores.

- Se realizará una recapitulación de las herramientas más importantes, resolviendo dudas y reforzando los conocimientos mediante ejercicios prácticos.

- Los estudiantes tendrán la oportunidad de aplicar lo aprendido en casos reales y recibir retroalimentación personalizada para afianzar su comprensión y destreza en el uso de Macros y Visual Basic

- **Evaluación.**

- La evaluación consistirá en un ejercicio práctico donde los estudiantes deberán aplicar los conceptos y herramientas aprendidas durante el curso de Macros y Visual Basic.

- Se evaluará su capacidad automatizar procesos utilizando Visual Basic

6. Visual Basic Avanzado

- **Conceptos básicos de Programación**

- Introducción a la Programación. Definición de Programación: La programación es el proceso de escribir instrucciones para que una computadora realice tareas específicas. Lenguajes de Programación: Explicación general de los lenguajes de programación y sus aplicaciones (enfoque en Visual Basic). Compilación y Ejecución: Cómo los programas son compilados y ejecutados por el sistema operativo.
 - Estructura Básica de un Programa. Bloques de Código: Explicación de cómo se estructuran los bloques de código, incluidos procedimientos, funciones y estructuras de control. Comentarios: La importancia de los comentarios en el código para hacer el programa más entendible.
 - Variables y Tipos de Datos. Qué son las Variables: Definición de variables y su rol en la programación como contenedores de datos. Tipos de Datos: Explicación de los tipos de datos básicos (Enteros, Flotantes, Cadenas de texto, Booleanos). Declaración de Variables: Uso de la palabra clave Dim para declarar variables y asignarles valores. Conversión de Tipos de Datos: Cómo convertir un tipo de dato a otro usando funciones como CInt, CDbl, CStr.
 - Operadores Básicos. Operadores Aritméticos: Explicación de los operadores básicos de matemáticas (+, -, *, /, ^). Operadores Lógicos: Introducción a los operadores lógicos básicos como And, Or, Not, y su uso en condiciones. Operadores de Comparación: Uso de operadores como =, <>, >, <, >=, <= para realizar comparaciones entre valores.
 - Control de Flujo: Condicionales y Bucles. Estructuras Condicionales: Introducción al uso de If, Else If, Else para tomar decisiones en el flujo del programa. Bucles Básicos: Introducción a los bucles For, While y Do While para repetir bloques de código. Uso de la Palabra Clave Exit: Cómo salir de un bucle o condicional anticipadamente.
 - Manejo de Errores. Errores Comunes en Programación: Errores de sintaxis, lógica y tiempo de ejecución. Manejo de Errores en Visual Basic: Introducción al uso de Try...Catch para capturar y manejar errores en tiempo de ejecución. Depuración Básica: Herramientas de depuración en el entorno de desarrollo (breakpoints, observación de variables).
 - Buenas Prácticas en Programación. Código Limpio y Legible: La importancia de escribir código claro y bien comentado. Modularización: Uso de funciones y procedimientos para hacer que el código sea más reutilizable y fácil de mantener. Evitar Repetición de Código: Uso de funciones y bucles para evitar escribir el mismo código varias veces.
- **Introducción a los Bucles en Visual Basic**
 - Concepto de Bucle. Explicación general sobre qué es un bucle: una estructura de control que permite ejecutar repetidamente un bloque de código mientras se cumpla una condición. Importancia de los bucles en la automatización de tareas repetitivas.
 - Tipos de Bucles en Visual Basic. Bucle For: Ejecuta un bloque de código un número específico de veces. Bucle While: Ejecuta un bloque de código mientras una condición se

mantenga verdadera. Bucle Do While: Similar al bucle While, pero con la diferencia de que la condición se evalúa después de la ejecución del código.

- Sintaxis Básica de los Bucles. For: Sintaxis general: For i = inicio To fin Step incremento. Explicación de los parámetros: inicio, fin, y Step (opcional). While: Sintaxis general: While condición. Explicación de la condición y cómo termina el bucle. Do While: Sintaxis general: Do While condición. Diferencia clave con el While tradicional.
- Ejemplos Prácticos. Bucle For: Iterar a través de un rango de celdas en Excel y aplicar un cambio. Bucle While: Sumar los números hasta que la suma sea mayor que un valor específico. Bucle Do While: Repetir una acción hasta que se cumpla una condición de parada.

• Bucles For

- Explicación Profunda del Bucle For. Cómo funciona el bucle For y cuándo utilizarlo. Explicación de la estructura de repetición definida: For variable = valor_inicial To valor_final Step incremento. El parámetro Step: ¿cómo afecta el incremento o decremento del contador?
- Modificación de los Parámetros del Bucle For. Cómo hacer que el bucle recorra un rango específico de celdas o valores. Diferencia entre usar Step 1 y Step -1 para incrementar o decrementar. Ejemplo con rangos de celdas: cómo recorrer un conjunto de datos con el bucle For.
- Uso de Bucle For con Rango de Celdas en Excel. Ejemplo práctico: recorrer un conjunto de celdas en una columna para verificar o modificar sus valores. Crear un contador que recorra las filas de una hoja de Excel y realice una operación (por ejemplo, sumar o concatenar valores). Optimización del bucle For para evitar errores y mejorar el rendimiento.
- Salir de un Bucle For Prematuramente. Uso de la instrucción Exit For para salir del bucle antes de que se cumpla la condición de finalización. Cuándo y por qué es útil esta instrucción.

• Bucles While y Do While

- Explicación Profunda de los Bucles While y Do While. While: Ejecuta el bloque de código mientras la condición sea verdadera. Sintaxis: While condición. Do While: Similar al While, pero con la diferencia de que evalúa la condición al final del bucle. Sintaxis: Do While condición.
- Diferencias y Usos Prácticos de While y Do While. Comparación de While y Do While: ¿cuál usar según el caso? Ejemplos de cuándo es más conveniente usar un bucle While (cuando no se sabe cuántas veces se va a ejecutar). Do While: Mejor opción cuando es necesario ejecutar al menos una vez el bloque de código antes de verificar la condición.
- Aplicación Práctica de Bucles While en Excel. Ejemplo de un bucle While para recorrer celdas hasta encontrar una celda vacía o hasta cumplir con una condición (por ejemplo, una suma acumulada). Uso de bucles Do While en formularios o en validaciones de entrada de datos (verificación continua mientras no se cumpla una condición de validación).

- Control de Bucle: Instrucciones Exit While y Exit Do. Uso de la instrucción Exit While o Exit Do para salir del bucle antes de que se cumpla la condición final. Ejemplo práctico: salirse de un bucle cuando se alcanza un valor específico.

- **Introducción a los UserForms**

- Qué es un UserForm: Definición de un UserForm como una interfaz gráfica de usuario dentro de Visual Basic para aplicaciones (VBA). Crear un UserForm: Cómo crear un UserForm desde cero en el Editor de VBA (menú Insertar > UserForm).
- Componentes Básicos de un UserForm: Explicación de los componentes principales: formularios, botones, cuadros de texto, etiquetas, etc.
- Propiedades del UserForm: Descripción de las propiedades básicas del UserForm (tamaño, título, colores, bordes, etc.).
- Diseño de un UserForm: Cómo organizar los controles en un UserForm mediante la interfaz de diseño (arrastrar y soltar controles).
- Mostrar un UserForm: Métodos para mostrar el UserForm, como UserForm.Show y cómo ocultarlo con UserForm.Hide.

- **Controles Básicos en un UserForm.**

- Botones (CommandButton): Cómo agregar y programar botones dentro de un UserForm. Explicación de eventos como Click para ejecutar acciones.
- Cuadros de texto (TextBox): Uso de cuadros de texto para la entrada de datos. Cómo recoger el valor de un TextBox con código y asignar valores programáticamente.
- Etiquetas (Label): Cómo agregar etiquetas para mostrar texto estático y cómo manipular sus propiedades.
- Cuadros de Combinación (ComboBox): Implementación de listas desplegables y cómo manipular los elementos del ComboBox.
- ListBox: Uso de ListBox para crear listas de selección múltiple y cómo acceder a los elementos seleccionados.
- Controles de selección (CheckBox, OptionButton): Implementación de casillas de verificación y botones de opción para opciones de sí/no.
- Eventos Básicos: Explicación de cómo asignar eventos a los controles, como el Click, Change y Enter para interactuar con los usuarios.

- **Programación Avanzada de UserForms**

- Uso de Variables con UserForms: Cómo utilizar variables globales y locales para almacenar datos ingresados en el UserForm.
- Estructura de Eventos en UserForms: Explicación de los diferentes eventos que pueden ser utilizados dentro de un UserForm, como Initialize, Activate, QueryClose, Terminate, etc.
- Control de Flujo con UserForms: Utilización de condicionales y bucles dentro de un UserForm para controlar el comportamiento de los controles según las interacciones del usuario.

- Validación de Datos en UserForms: Técnicas para validar los datos ingresados en los TextBox, ComboBox y otros controles antes de procesarlos.

- Cierre de UserForm: Cómo programar el cierre automático de un UserForm después de una acción, o cómo utilizar un botón de cierre.

- Acciones al Cerrar el UserForm: Definir qué ocurre al cerrar el formulario, incluyendo la validación de datos y la acción de guardar o cancelar cambios.

- **Integración de UserForms con Excel y Uso de Funcionalidades Avanzadas**

- Transferencia de Datos entre Excel y UserForms: Cómo tomar los datos de un UserForm y escribirlos en celdas de una hoja de Excel, o viceversa.

- Uso de UserForms para Crear Reportes: Creación de formularios para que los usuarios seleccionen datos y luego generen reportes automáticos en Excel.

- Multiples UserForms: Manejo de múltiples UserForms abiertos al mismo tiempo. Cómo pasar datos entre UserForms y mantener una navegación eficiente.

- Personalización de la Interfaz de Usuario (UI): Creación de interfaces más sofisticadas, como formularios de login o paneles de control con varias páginas.

- Control de Errores en UserForms: Uso de On Error para manejar posibles errores durante la ejecución de un UserForm y cómo prevenir bloqueos en la aplicación.

- Estilos y Temas de Diseño: Aplicación de estilos visuales (colores, tamaños, fuentes) para hacer el UserForm más atractivo y fácil de usar.

- Optimización de UserForms: Técnicas para optimizar el rendimiento de UserForms grandes o complejos, incluyendo la minimización de recursos y la mejora de la experiencia del usuario

- **Migraciones de Datos con Visual Basic.**

- Manejo de grandes volúmenes de datos: Limpieza de datos para transformar filas y columnas. Corrección de formatos de textos, números y fechas. Fórmulas y aritmética para crear campos calculados.

- Exportación de datos: Generación de reportes en pdf. Comandos necesarios. Generación de archivos en formato txt para posterior uso. Creación de nuevos archivos de Excel. Exportación colocando Ruta.

- Búsqueda de comandos: Explicación de cómo buscar ayuda en internet, Chatgpt y otros. Problemas a resolver.

- **Caso Práctico: Sistema de Inventario 1**

- Conceptos Básicos del Sistema de Inventario: Definición y objetivos de un sistema de inventario, tipos de inventarios (por productos, por categorías, por ubicación, etc.).

- Elementos Clave del Sistema: Revisión de los elementos necesarios para un sistema de inventario básico: productos, cantidades, precios, códigos, fechas, etc.

- Diseño de la Base de Datos en Excel: Estructura básica de la hoja de Excel donde se almacenarán los datos de inventario: columnas para código de producto, nombre, cantidad en stock, precio unitario, etc.

- Uso de Formularios (UserForms): Creación de un formulario para la entrada de datos en el sistema de inventario, incluyendo controles como TextBox para nombre y cantidad de productos, ComboBox para seleccionar categorías, y CommandButton para agregar productos.
- Funciones de Entrada de Datos: Implementación de funciones para agregar nuevos productos al inventario desde el formulario. El código capturará los valores introducidos en los TextBox y actualizará la base de datos en Excel.
- Validaciones de Datos: Validación de entradas, como verificar que los campos no estén vacíos, que las cantidades sean números positivos, y que los códigos de productos sean únicos.
- Uso de ListBox para Mostrar Inventario: Implementación de un ListBox para mostrar el inventario actual, permitiendo a los usuarios ver los productos registrados de manera visual.

• **Caso Práctico: Sistema de Inventario 2**

- Actualización de Inventario: Desarrollo de una función que permita modificar la cantidad de un producto ya existente. El sistema buscará el producto en la base de datos y actualizará su cantidad según el valor introducido por el usuario.
- Eliminación de Productos: Creación de una función para eliminar productos del inventario. Se incluirá una validación para confirmar la eliminación y evitar borrar accidentalmente productos importantes.
- Búsqueda de Productos en el Inventario: Implementación de una función que permita buscar productos por nombre o código. El usuario podrá ingresar un criterio de búsqueda y el sistema mostrará los productos que coincidan con ese criterio.
- Informes de Inventario: Generación de un informe con el estado actual del inventario, que puede incluir el total de productos en stock, el valor total del inventario, o productos con bajo stock.
- Cálculo de Valor: Total de Inventario: Utilización de funciones en Visual Basic para calcular el valor total del inventario en función de la cantidad y el precio de cada producto.
- Exportación de Datos a CSV o Excel: Implementación de una función para exportar el inventario actualizado a un archivo CSV o Excel, facilitando la gestión y el almacenamiento de los datos fuera de la aplicación.
- Interacción con el Usuario: Uso de MsgBox y InputBox para mejorar la experiencia del usuario con mensajes de confirmación, advertencia y alertas cuando se realicen cambios importantes en el inventario (agregar, eliminar, actualizar).
- Optimización del Sistema: Revisión de las buenas prácticas para mantener el sistema eficiente a medida que el inventario crece, incluyendo el uso adecuado de referencias a celdas, la limpieza de datos innecesarios, y la organización de la base de datos.

• **Caso Práctico: Supermercado 1**

- introducción al Sistema de Ventas y Compras: Definición y propósito del sistema para un supermercado. La gestión de inventarios y las transacciones de compras y ventas.

- Estructura de la Base de Datos: Diseño de la base de datos en Excel para gestionar productos. Las columnas principales: código de producto, nombre, categoría, precio de compra, precio de venta, cantidad en stock, etc.

- Formulario de Gestión de Productos: Creación de un formulario de Visual Basic (User-Form) para ingresar, actualizar y eliminar productos del inventario. Uso de controles como TextBox para ingresar datos, ComboBox para categorías de productos y CommandButton para agregar, editar o eliminar productos.

- Búsqueda de Productos: Implementación de una barra de búsqueda que permita a los usuarios localizar rápidamente productos por nombre, código o categoría.

- **Caso Práctico: Supermercado 2**

- Formulario de Ventas: Creación de un formulario para registrar las ventas. El usuario podrá seleccionar productos del inventario, ingresar la cantidad y automáticamente el sistema calculará el total de la venta (precio de venta * cantidad).

- Registro de Ventas: Al completar una venta, el sistema actualizará las cantidades de los productos en el inventario y almacenará un registro de la transacción (producto, cantidad, precio, total, fecha y hora).

- Formulario de Compras: Creación de un formulario para registrar las compras de productos a proveedores. El sistema permitirá agregar productos comprados, especificar la cantidad y el precio de compra, y actualizará el inventario con los nuevos productos.

- Registro de Compras: Al realizar una compra, el sistema actualizará el inventario con la nueva cantidad de productos y almacenará un registro de la transacción (producto, cantidad, precio, total, fecha y hora).

- Cálculo de Totales: Implementación de funciones que calculan el total de la venta y el total de la compra. Se incluyen funcionalidades para calcular descuentos, impuestos y totales con fórmulas personalizadas.

- Reporte de Ventas y Compras: Creación de un reporte que muestre las ventas y compras realizadas durante un periodo específico. Este reporte incluirá detalles como el número de productos vendidos/comprados, total de ventas/compras, ganancias y pérdidas.

- Generación de Informes: Desarrollo de informes que muestren la situación económica del Supermercado.

- **Repaso:**

- La clase de repaso se enfocará en revisar los conceptos clave aprendidos durante las clases anteriores.

- Se realizará una recapitulación de las herramientas más importantes, resolviendo dudas y reforzando los conocimientos mediante ejercicios prácticos.

- Los estudiantes tendrán la oportunidad de aplicar lo aprendido en casos reales y recibir retroalimentación personalizada para afianzar su comprensión y destreza en el uso de Visual Basic

- **Evaluación.**

- La evaluación consistirá en un ejercicio práctico donde los estudiantes deberán aplicar los conceptos y herramientas aprendidas durante el curso de Visual Basic.
- Se evaluará su capacidad automatizar procesos y generar poderosos sistemas utilizando Visual Basic

7. AppSheet Básico

- **Introducción a Appsheets**

- ¿Qué es AppSheet?: Explicación de la plataforma como una herramienta no-code para crear aplicaciones personalizadas.
- Registro y creación de cuenta: Paso a paso para crear una cuenta en AppSheet.
- Interfaz de usuario de AppSheet: Explicación de la interfaz, herramientas principales, y opciones de configuración.
- Creación de la primera aplicación: Conexión con Google Sheets y creación de una app básica.
- Tipos de aplicaciones: Apps para gestión de datos, inventarios, formularios, proyectos, etc.

- **Primer Proyecto: Crear una App Simple**

- Estructura de Google Sheets: Cómo organizar tus datos de manera eficiente en una hoja de cálculo (columnas y filas).
- Conexión de AppSheet con Google Sheets: Vinculación de datos desde una hoja de cálculo a AppSheet.
- Diseño básico de la interfaz: Creación de una vista de tabla, detalles y formularios básicos.
- Primeros pasos con el diseño de la app: Personalización básica del aspecto de la app.
- Prueba en dispositivos móviles: Uso de la app en un dispositivo móvil.

- **Estructura y Diseño de la App**

- Personalización de vistas: Modificación de vistas predeterminadas, cómo crear vistas de tabla, formulario, mapa, etc.
- Configuración de la interfaz: Ajuste de la apariencia de la app: colores, íconos, y estilos.
- Ajustes de diseño para dispositivos móviles: Mejora de la interfaz para visualización en diferentes tamaños de pantalla.
- Vista de detalles y edición: Cómo permitir la edición de registros y cómo mostrar detalles al hacer clic en un registro.
- Vistas personalizadas: Crear vistas filtradas o con criterios de orden específicos.

- **Fuentes de Datos en AppSheet**

- Tipos de fuentes de datos: Google Sheets, Excel, SQL, Airtable y otras fuentes posibles.
- Cómo importar datos desde diferentes fuentes: Conexión a diversas fuentes de datos como hojas de cálculo o bases de datos externas.

- Estructura de datos: Cómo organizar los datos correctamente en una hoja para facilitar la importación en AppSheet.

- Tipos de columna y configuración: Definición de tipo de datos (texto, número, fecha, imagen, etc.) y propiedades asociadas.

- Actualizar y sincronizar datos: Mantenimiento de la actualización en tiempo real entre la app y la fuente de datos.

- **Formatos y Tipos de Columnas**

- Tipos de columnas: Texto, número, fecha, archivo, imagen, URL, entre otras.

- Configuración avanzada de columnas: Cómo establecer restricciones de formato, como número de caracteres o valores válidos.

- Columnas de referencia: Cómo establecer relaciones entre diferentes tablas.

- Formato condicional: Establecer reglas visuales para cambiar el formato de los datos según ciertos criterios.

- Uso de columnas virtuales: Creación de columnas calculadas que no están presentes en la fuente de datos, pero se generan a través de expresiones

- **Creación y Configuración de Vistas**

- Vistas básicas: Crear vistas de tabla, formulario, de mapa, etc.

- Vistas personalizadas: Cómo filtrar, ordenar y agrupar datos para crear vistas más detalladas.

- Cambios dinámicos en vistas: Configuración de reglas para mostrar u ocultar vistas según ciertos criterios.

- Vista de galería: Crear una vista de imágenes o tarjetas de información.

- Vistas de mapa: Uso de coordenadas geográficas para crear mapas interactivos dentro de la app.

- **Filtros y Búsquedas en las Apps**

- Filtros avanzados: Cómo implementar filtros dinámicos dentro de las vistas.

- Búsqueda por columna: Configuración de la barra de búsqueda para buscar dentro de una vista de datos.

- Uso de expresiones en filtros: Cómo utilizar expresiones para filtrar dinámicamente.

- Uso de filtros predeterminados: Crear filtros que automáticamente filtren los registros por valores predefinidos.

- Optimización de búsqueda: Mejora de tiempos de carga y búsqueda eficiente.

- **Validaciones de Datos**

- Reglas de validación: Cómo asegurar que los datos ingresados en formularios sean correctos (por ejemplo, solo valores numéricos).

- Mensajes de error personalizados: Mostrar mensajes de advertencia si los datos no cumplen con los requisitos.
- Restricciones de valores: Validaciones para que un dato no se repita o que esté dentro de un rango determinado.
- Validación de fecha: Asegurar que las fechas ingresadas sean lógicas, por ejemplo, que no se ingresen fechas futuras.
- Uso de expresiones para validaciones: Implementar expresiones complejas para validaciones de datos en tiempo real.

- **Relaciones entre Datos y Referencias**

- Relaciones uno a uno y uno a muchos: Cómo establecer relaciones entre tablas usando columnas de referencia.
- Uso de referencias: Cómo obtener datos de tablas relacionadas usando referencias.
- Vistas de detalle y sub-registros: Mostrar registros relacionados en una vista secundaria (detalle).
- Agrupación de registros: Agrupar registros de una tabla relacionada dentro de otra.
- Optimización de relaciones: Mejores prácticas para mantener las relaciones limpias y rápidas.

- **Acciones Personalizadas**

- Qué son las acciones?: Cómo crear acciones personalizadas como botones para acciones rápidas (agregar, eliminar, editar).
- Acciones dentro de la app: Crear botones de acción en las vistas para que los usuarios interactúen con los datos.
- Navegar entre vistas: Crear botones de navegación para cambiar entre diferentes vistas dentro de la app.
- Acciones en formularios: Implementar acciones personalizadas al enviar o editar formularios.
- Acciones de automatización: Acciones para enviar correos electrónicos, generar informes o crear nuevos registros automáticamente.

- **Funciones y Expresiones en AppSheet**

- . ¿Qué son las expresiones?: Explicación del uso de expresiones en AppSheet para personalizar el comportamiento de la app.
- Funciones de texto y matemáticas: Uso de funciones como CONCATENATE, SUM, TEXT, LEFT, MID, etc.
- Funciones condicionales (IF, SWITCH): Cómo usar funciones para aplicar lógicas condicionales.
- Funciones de búsqueda: Buscar datos dentro de la app usando funciones como LOOKUP, SELECT, FILTER, etc.
- Cálculos dinámicos: Implementación de cálculos en tiempo real para mostrar valores actualizados.

• **Ejercicio: Club Deportivo 1**

- Identificación de las entidades clave: Definir las entidades esenciales para el sistema como "Estudiantes", "Entrenadores", "Clases", "Horarios", "Pagos" y "Asistencia". Estructura de tablas:
- Crear las tablas necesarias en Google Sheets para almacenar los datos de cada entidad. Ejemplo de tablas: Estudiantes: Nombre, Edad, Género, Teléfono, Correo electrónico, Fecha de inscripción, etc. Entrenadores: Nombre, Especialidad, Horarios disponibles, etc. Clases: Nombre de la clase, Descripción, Duración, Entrenador asignado, etc. Horarios: Fecha y hora de cada clase programada. Pagos: Fecha de pago, Monto, Método de pago, Estado del pago. Asistencia: Fecha, Estudiante presente o ausente, Clase a la que asistió.
- Relación entre tablas: Establecer relaciones entre tablas para vincular a los estudiantes con las clases que toman, a los entrenadores con las clases que dictan y los pagos con los estudiantes.
- Creación de la aplicación: Crear una nueva app en AppSheet desde una hoja de cálculo.
- Definición de vistas: Vista de estudiantes: Mostrar la lista de estudiantes con opción para añadir, editar o eliminar registros. Vista de clases: Mostrar las clases disponibles, incluyendo los horarios y entrenadores asignados. Vista de pagos: Visualizar los pagos realizados y pendientes por parte de los estudiantes.
- Configuración de relaciones: Usar referencias entre tablas, como hacer que cada clase esté vinculada a los estudiantes que la toman y que los pagos estén relacionados con los estudiantes correspondientes.
- Formularios para registrar nuevos estudiantes: Crear formularios donde los administradores puedan ingresar los datos de los nuevos estudiantes.
- Formulario de inscripción a clases: Permitir que los estudiantes se inscriban a las clases disponibles seleccionando de una lista desplegable.

• **Ejercicio: Club Deportivo 2**

- Implementación de una vista de asistencia. Registro de asistencia: Crear una vista donde los entrenadores o administradores puedan marcar la asistencia de los estudiantes a las clases.
- Fluxos de trabajo de seguimiento: Generar informes automáticos que muestren la asistencia de los estudiantes a las clases a lo largo del mes o trimestre.
- Alertas para faltas recurrentes: Configurar alertas para que los administradores reciban notificaciones si un estudiante se ausenta frecuentemente.
- Gestión de pagos y facturación. Sistema de pagos: Permitir que los estudiantes registren sus pagos, asignando fechas y montos a cada transacción. Control de pagos pendientes: Crear una vista o dashboard que muestre los pagos pendientes y las fechas de vencimiento.
- Personalización de la interfaz de usuario. Diseño de la app: Personalizar la apariencia de la app, incluyendo colores, logo de la academia deportiva, y estilo general de la interfaz.

Optimización de la navegación: Crear un menú de navegación intuitivo para acceder rápidamente a secciones como "Estudiantes", "Clases", "Pagos", "Asistencia", etc.

- Creación de reportes e informes. Reportes de rendimiento de los estudiantes: Configurar un reporte que muestre el desempeño y asistencia de los estudiantes a lo largo del mes. Reportes financieros: Crear reportes que resuman los pagos realizados por los estudiantes y las ganancias de la academia deportiva. Generación de reportes mensuales o anuales: Automatizar la creación de reportes periódicos para llevar un control de los ingresos, asistencia, y participación en las clases

- **Repasso:**

- La clase de repaso se enfocará en revisar los conceptos clave aprendidos durante las clases anteriores.
- Se realizará una recapitulación de las herramientas más importantes, resolviendo dudas y reforzando los conocimientos mediante ejercicios prácticos.
- Los estudiantes tendrán la oportunidad de aplicar lo aprendido en casos reales y recibir retroalimentación personalizada para afianzar su comprensión y destreza en el uso de Appsheets

- **Evaluación.**

- La evaluación consistirá en un ejercicio práctico donde los estudiantes deberán aplicar los conceptos y herramientas aprendidas durante el curso de Appsheets
- Se evaluará su capacidad de crear y optimizar sus aplicaciones

8. Appsheets Avanzado

- **Funciones de Expresión (Expression Functions)**

- Definición de expresiones: Entender qué son las expresiones en AppSheet y cómo se utilizan para calcular valores dinámicos.
- Tipos de funciones avanzadas: IF(): Uso de funciones condicionales avanzadas para mostrar resultados según una condición. SELECT(): Obtener listas de datos de una columna específica en una tabla, aplicando filtros. LOOKUP(): Buscar un valor en una tabla y devolver un valor de otra columna de la misma fila. ANY(): Función que obtiene el primer valor encontrado en una columna. MAX() y MIN(): Encontrar el valor máximo o mínimo de una columna o conjunto de datos. COUNT() y COUNTIF(): Contar el número de filas en una tabla o contar las filas que cumplen con una condición específica.
- Uso de expresiones en diferentes contextos: Aplicación de estas funciones en diferentes lugares como en fórmulas de columnas, en acciones y en vistas de tipo de formulario.

- **Funciones de Agregación**

- Sumar valores: Uso de funciones como SUM() para agregar los valores de una columna de forma automática.
- Promedio (AVERAGE): Calcular el promedio de un conjunto de valores.

- Agrupar datos: Usar funciones como GROUPBY() para agrupar datos en base a un campo específico y luego aplicar una función agregada (por ejemplo, contar el número de registros por cada grupo).

- Contar elementos únicos (UNIQUE()): Contar solo los elementos únicos en una columna para obtener resultados sin duplicados.

- Funciones de fecha en agregación: Uso de MIN() y MAX() con fechas para calcular la fecha más antigua o más reciente en un conjunto de datos.

- **Expresiones de Fecha y Hora**

- Funciones de manipulación de fechas: DATE(), TIME(), DATETIME(): Convertir texto a fechas, horas o fechas y horas. TODAY() y NOW(): Obtener la fecha o la fecha y hora actual. YEAR(), MONTH(), DAY(): Extraer partes específicas de una fecha, como el año, mes o día. DATEADD() y DATEDIF(): Añadir o restar días a una fecha y calcular la diferencia entre dos fechas. WEEKDAY() y WEEKNUM(): Obtener el día de la semana o el número de la semana de una fecha.

- Comparación de fechas: Uso de expresiones condicionales con fechas, como comparar si una fecha es anterior, posterior o igual a otra.

- **Funciones para Manipulación de Texto**

- Funciones de texto básicas: UPPER() y LOWER(): Convertir el texto a mayúsculas o minúsculas. CONCATENATE() y CONCAT(): Unir dos o más textos en uno solo. LEN(): Calcular la longitud de un texto (número de caracteres). LEFT(), RIGHT(), MID(): Extraer caracteres desde el principio, final o una posición específica dentro de una cadena de texto. Funciones avanzadas de texto:

- SEARCH() y FIND(): Buscar un texto específico dentro de una cadena y devolver la posición en la que se encuentra. SUBSTITUTE(): Reemplazar una parte del texto por otro valor. TEXT(): Formatear un texto o un número en un formato específico (por ejemplo, para fechas o números con decimales).

- **Funciones Condicionales Avanzadas**

- Función SWITCH(): Evaluar múltiples condiciones y devolver un resultado según el valor de una expresión. Similar a múltiples sentencias IF() anidadas.

- Función IFS(): Evaluar varias condiciones en orden y devolver el primer resultado verdadero. Es más eficiente que anidar múltiples IF().

- FUNCIONES LÓGICAS AVANZADAS: Uso de funciones como AND(), OR() para combinar condiciones en expresiones lógicas complejas.

- FUNCIONES DE ERROR: Uso de ISBLANK(), ISERROR(), IFERROR() para manejar valores nulos o errores en las expresiones.

- **Funciones de Lista**

- IN(): Comprobar si un valor está presente dentro de una lista.

- SELECT() y FILTER(): Filtrar elementos dentro de una lista basándose en una condición.

- INDEX(): Acceder a un valor en una posición específica de una lista.
- **Seguridad y Control de Accesos**
 - Autenticación de usuarios: Implementación de diferentes métodos de autenticación, como Google, Microsoft o autenticación personalizada.
 - Control de acceso basado en roles: Crear roles de usuario y asignar permisos específicos para cada grupo (administradores, usuarios, etc.).
 - Restricción de acceso a vistas: Establecer restricciones de acceso a vistas basadas en el rol del usuario o condiciones específicas.
 - Seguridad de datos: Proteger datos sensibles y asegurarse de que solo los usuarios autorizados tengan acceso a la información relevante.
 - Verificación de dispositivos: Limitar el acceso según el tipo de dispositivo (smartphone, tablet, computadora).
 - Protección contra cambios no deseados: Cómo evitar que los usuarios realicen cambios no deseados en los datos.
- **Reportes y Generación de Archivos**
 - Generación de reportes dentro de la app: Crear y personalizar reportes automáticos basados en los datos de la app.
 - Exportación a PDF y Excel: Cómo configurar la exportación de datos y reportes en formatos como PDF o Excel.
 - Diseño de reportes: Establecer el diseño de los reportes, elegir qué campos incluir y cómo formatearlos.
- **Funcionalidades Avanzadas de Formularios**
 - Formularios condicionales: Configurar formularios que cambian dinámicamente según los datos ingresados o el rol del usuario.
 - Dependencia entre campos: Establecer relaciones entre campos (por ejemplo, un campo depende del valor seleccionado en otro).
 - Campos de lista desplegable: Crear listas dependientes que cambian según los valores de otras listas o campos.
 - Autocompletar y sugerencias: Implementar autocompletado para campos de texto y sugerencias basadas en la entrada del usuario.
 - Manejo de errores en formularios: Validar entradas en formularios y mostrar mensajes de error o advertencia según sea necesario.
- **Tablas y Gráficos Avanzados**
 - Tablas avanzadas: Configuración de tablas que muestren datos en múltiples niveles, con agrupación y resumen de datos.
 - Gráficos interactivos: Crear gráficos que interactúan con el usuario, como gráficos de barras, líneas, de torta, etc.
 - Gráficos de tendencia y predicción: Implementación de gráficos que muestran tendencias a partir de los datos históricos.

- Segmentadores de gráficos: Uso de filtros dinámicos para que el usuario seleccione qué datos desea ver.

- **Rendimiento de la App**

- Optimización de bases de datos: Mejoras en la organización de datos para asegurar un buen rendimiento de la app.
- Reducción de tiempos de carga: Técnicas para disminuir los tiempos de carga de las vistas y formularios.
- Uso de datos en caché: Cómo almacenar ciertos datos localmente para que la app cargue más rápido.
- Carga diferida de datos: Cargar datos solo cuando sean necesarios, evitando cargar toda la base de datos en un solo momento.
- Monitoreo de rendimiento: Herramientas para monitorear el rendimiento de la app y detectar posibles problemas.

- **Ejercicio Transporte de Pasajeros 1**

- Crear las siguientes tablas en Google Sheets. Creación de tablas necesarias para la App. Claves primarias y foráneas. Estructuración de datos.
- Creación de la app en AppSheet. Conectar las tablas. Crea una nueva app en AppSheet y conectar las tablas de Google Sheets como las fuentes de datos. Verificación de conexiones. Configuración de Tablas.
- Creación de Vistas Necesarias. Creación una vista de formulario para agregar datos. Creación de vistas de detalle para ver información. Vistas de Reservas de viajes

- **Ejercicio Transporte de Pasajeros 2**

- Desarrollo de informes: Creación de vistas complejas, Dashboards y Calendarios. Obtención de gráficos de monitoreo de información
- Configuración de acciones. Para estandarizar secuencia de tareas.
- Diseño de la interfaz de usuario. Interfaz amigable, fácil de usar para administradores. Utilización de botones, listas desplegables y formularios claros para facilitar el proceso de registro de pasajeros, reservas, y viajes.
- Pruebas y ajustes. Pruebas de la app: pruebas con datos reales para asegurar que la app funciona correctamente. Ajustes en base a los resultados de las pruebas.
- Implementación: Gestión viajes, conductores, vehículos y pasajeros desde una interfaz centralizada.

- **Repaso:**

- La clase de repaso se enfocará en revisar los conceptos clave aprendidos durante las clases anteriores.
- Se realizará una recapitulación de las herramientas más importantes, resolviendo dudas y reforzando los conocimientos mediante ejercicios prácticos.

- Los estudiantes tendrán la oportunidad de aplicar lo aprendido en casos reales y recibir retroalimentación personalizada para afianzar su comprensión y destreza en el uso de Appsheets

- **Evaluación.**

- La evaluación consistirá en un ejercicio práctico donde los estudiantes deberán aplicar los conceptos y herramientas aprendidas durante el curso de Appsheets
- Se evaluará su capacidad de crear y optimizar sus aplicaciones

ESTRATEGIAS DE ENSEÑANZA

Las clases serán virtuales, lo que implica conexión simultánea entre los estudiantes y el equipo docente en los días horarios planteados, a través de la plataforma de videoconferencias.

En las clases, el docente a cargo del módulo presentará el tema, destacando su ubicación e importancia en el marco del curso, los objetivos que se pretenden lograr, los materiales a trabajar y las actividades a desarrollar, además de promover espacios de debate para el abordaje de la temática.

Por otra parte, los docentes también podrán solicitar actividades extra-clases, apoyándose en videos, lecturas, casos y actividades que permitirán afianzar los conocimientos adquiridos y aplicarlas a diferentes situaciones del contexto. Estas se desarrollarán a través de la plataforma Moodle de UNNE Virtual.

A estos efectos se habilitará un aula virtual y cada módulo contará con material de apoyo, bibliografía y diversas actividades tendientes a abordar aspectos teóricos y prácticos relacionados con el contenido. Se realizarán actividades tanto individuales como grupales. Se habilitarán foros tanto de debate como de consultas, los cuales se utilizarán como espacio de encuentro entre los cursantes y entre estos y los docentes.

Para las clases se comunicará, en el aula virtual habilitada, el enlace a la plataforma de videoconferencia a utilizar.

Cada módulo estará a cargo de uno o más docentes con trayectoria en la temática a abordar, estando prevista la participación de expositores invitados que disertarán sobre su experiencia específica en el área. Se plantea un escenario interactivo y de comunicación permanente con los asistentes.

ESTRATEGIAS DE EVALUACIÓN

Para la acreditación del trayecto formativo se tiene en cuenta:

- a. La lectura y visualización de todos los contenidos.
- b. Grado de participación en todos los foros de debate y actividades propuestas.

c. La entrega y aprobación de todas las actividades y ejercicios que se consignen como obligatorios.

d. El alumno deberá finalizar la cursada dentro de la duración del trayecto sin excepción. Terminado el mismo, no podrá entregar ninguna actividad obligatoria adeudada.

Al culminar el trayecto formativo, de haber cumplido con lo anteriormente nombrado se otorgará un Certificado de Aprobación.

CREACIÓN DE LAS AULAS VIRTUALES Y MATRICULACIÓN DE USUARIOS

Las aulas virtuales de este curso se habilitarán conforme los mecanismos institucionales utilizados en la UNNE. La apertura de las aulas virtuales y la matriculación de los usuarios será función de los Coordinadores del curso, previo registro del pago del arancel correspondiente.

PLANTEL DOCENTE

En cuanto a la coordinación y dictado de la misma, la Diplomatura en Herramientas Digitales para la Gestión y Análisis de Negocios estará a cargo del Cr. Sergio Gómez Pamies, docente de la Facultad de Ciencias Económicas, especialista en cada uno de los módulos y temáticas a desarrollar.

Funciones de la coordinación y de los docentes:

a. Coordinador: será responsable de la gestión académica y administrativa del curso, lo que implica desarrollar las siguientes tareas:

- Articular con la Secretaría de Extensión y Ejercicio Profesional la gestión administrativa, financiera y de difusión del curso.
- Convocar a los docentes de los diferentes módulos, definiendo pautas de trabajo.
- Propiciar y evaluar que los módulos cuenten con los recursos didácticos necesarios para el desarrollo adecuado del proceso de enseñanza-aprendizaje.
- Atender las consultas de los estudiantes, generando mecanismos de resolución adecuados y eficientes.
- Gestionar la certificación de los estudiantes.

b. Docentes: tendrán como función la gestión académica de los módulos bajo su responsabilidad, lo que implica desarrollar las siguientes tareas:

- Elaborar la programación del módulo.
- Elaborar los diferentes materiales didácticos que se utilizarán durante el cursado.

- Organizar y disponer los diferentes materiales didácticos en el aula virtual.
- Evaluar y calificar la participación de los estudiantes.
- Atender las consultas de los estudiantes.
- Elevar el informe final del módulo.

REQUISITOS DE ADMISIÓN

Para acceder a esta Diplomatura, los postulantes deberán presentar:

- a. Fotocopia del título secundario (o cualquiera superior a este) y fotocopia del DNI. En este punto, la documentación puede ser enviada digitalmente.
- b. Al tratarse de una propuesta de formación con modalidad virtual, los postulantes deben poseer PC con conexión a Internet para el desarrollo de las actividades del curso.
- c. Será requisito para confirmar el cupo, el abono de la matrícula.

FUENTE DE FINANCIAMIENTO

Esta Diplomatura en Herramientas Digitales para la Gestión y Análisis de Negocios se autofinanciará mediante un arancel que los estudiantes deberán abonar.

El abono del arancel habilita a los/as estudiantes a:

- a. Participar del encuentro sincrónico
- b. Descargar los materiales de lectura y acceder a los contenidos audiovisuales dispuestos en el aula virtual.
- c. Realizar los ejercicios prácticos de autoevaluación, los casos de estudio y participar en los diferentes foros.
- d. Realizar las actividades finales de evaluación para acreditar la aprobación de los módulos y recibir la certificación correspondiente.

Hoja de firmas