

Introducción a la programación en Python

Escuela de Invierno y Análisis de Datos

Centro Hodós, Universidad Católica del Uruguay

Modalidad: Asincrónico/Virtual

Docente: Ornella Novino **Correo:** istar.novino@gmail.com

Discord Google Collab

Objetivos del curso:

El objetivo de este curso es introducir a los estudiantes en el lenguaje de programación Python para el análisis de datos mediante ejemplos prácticos. Este curso ofrece los conocimientos básicos de como trabajar en la carga, inspección y consulta de datos mediante las librerías más utilizadas tales como Numpy y Pandas, mientras que al mismo tiempo como visualizarlos mediante Matplotlib, Seaborn y Plotly.

Estructura del curso:

Descripción general del curso

- [Google Drive](#)
- [GitHub](#)

Módulo 1: Introducción a Python

1.1 Bienvenida y Google Colab

- **Objetivo:** Familiarizarse con Google Colab como entorno de desarrollo.
- **Contenidos:**
 - ¿Qué es Google Colab y por qué usarlo?
 - Celdas de código y texto (Markdown).
 - Atajos de teclado básicos.
 - Ejecutar código y comentar con `#`.

1.2 ¿Qué es Python?

- **Objetivo:** Comprender las características y usos de Python.
- **Contenidos:**
 - Definición y ventajas de Python.
 - Áreas de aplicación: desarrollo web, automatización, ciencia de datos, etc.
 - Mostrar mensajes con `print()`.

1.3 Operaciones básicas

- **Objetivo:** Aprender operaciones aritméticas y lógicas.

- **Contenidos:**
 - Operadores aritméticos: `+`, `-`, `*`, `/`, `//`, `%`, `**`.
 - Operadores relacionales: `==`, `!=`, `>`, `<`, `>=`, `<=`.
 - Operadores lógicos: `and`, `or`, `not`.

1.4 Variables y tipos de datos

- **Objetivo:** Conocer los tipos de datos y usar variables.
- **Contenidos:**
 - Declaración de variables.
 - Tipos de datos: `int`, `float`, `bool`, `str`.
 - Función `type()`.

1.5 Estructuras de datos

- **Objetivo:** Introducir listas, tuplas y diccionarios.
- **Contenidos:**
 - Listas: crear, acceder, modificar.
 - Tuplas: crear, acceder.
 - Diccionarios: crear, acceder.

1.6 Estructuras de control

- **Objetivo:** Controlar el flujo del código.
- **Contenidos:**
 - Condicionales: `if`, `elif`, `else`.
 - Bucles: `for`, `while`.

1.7 Funciones

- **Objetivo:** Comprender y crear funciones.
- **Contenidos:**
 - Funciones integradas: `print()`, `max()`, `min()`.
 - Definir funciones con `def`.

Ejercicio 1: Este ejercicio tiene como principal objetivo poner en práctica lo aprendido en el módulo 1. Realiza el ejercicio siguiendo la consigna, ten en cuenta que hay varias maneras de resolverlo. Por cualquier consulta no dudes escribir en el foro.

- [Google Drive](#)
- [GitHub](#)

Módulo 2: Análisis de datos con Python

2.1 Introducción al análisis de datos con Python

- **Objetivo:** Comprender la importancia del análisis de datos y cómo Python se utiliza en este campo.
- **Contenidos:**
 - ¿Qué es el análisis de datos?

- El rol de Python en el análisis de datos: ventajas y aplicaciones.
- Introducción a las librerías de análisis de datos: NumPy, Pandas, etc.

2.2 Librerías en Python

- **Objetivo:** Aprender a importar y utilizar librerías externas en Python.
- **Contenidos:**
 - ¿Qué son las librerías y para qué sirven?
 - Importar librerías: `import numpy`, `import numpy as np`.
 - Acceder a funciones y objetos de las librerías: `np.random.randint()`.

2.3 NumPy

- **Objetivo:** Introducir la librería NumPy para el manejo de arreglos numéricos.
- **Contenidos:**
 - ¿Qué es NumPy y por qué es importante para el análisis de datos?
 - Crear arreglos unidimensionales (vectores): `np.array([2, 6, 8, 10, 12, 14])`.
 - Operaciones con arreglos: `m_unidimensional ** 2`.
 - Crear arreglos bidimensionales (matrices): `np.array([[1, 2, 3], [4, 5, 6], [7, 8, 9]])`.
 - Sumar elementos de una matriz: `np.sum(matriz2)`.
 - Acceder a elementos de un arreglo (slicing): `matriz1[0]`, `matriz1[0:2]`.

2.4 Pandas

- **Objetivo:** Introducir la librería Pandas para la manipulación y análisis de datos tabulares.
- **Contenidos:**
 - ¿Qué es Pandas y por qué es útil para el análisis de datos?
 - Series y DataFrames: las estructuras de datos principales de Pandas.
 - Importar datos con Pandas:
 - `.read_csv()`: leer archivos CSV.
 - `.read_excel()`: leer archivos Excel.

2.5 Estadísticos descriptivos con Pandas

- **Objetivo:** Aprender a obtener estadísticas descriptivas de los datos con Pandas.
- **Contenidos:**
 - Calcular la media: `df['col'].mean()`.
 - Obtener estadísticas descriptivas: `df['col'].describe()`.
 - Agrupar datos con `groupby()`: `df.groupby('col1')['col2'].mean()`.

2.6 Filtros y selección de datos con Pandas

- **Objetivo:** Aprender a filtrar y seleccionar datos en un DataFrame de Pandas.
- **Contenidos:**
 - Seleccionar columnas: `ages = df["col"]`.
 - Filtrar filas por condición: `over_20 = df[df["col"] > 20]`.
 - Filtrar con múltiples condiciones: `df[(df['col'] == 'opcion1') & (df['col'] == 'opcion2')]`.

- Usar `isin()` para filtrar por valores específicos: `value = df[df["col"].isin(['opcion1', 'opcion2'])]`.

2.7 Transformar datos con Pandas

- **Objetivo:** Aprender a modificar y transformar datos en un DataFrame.
- **Contenidos:**
 - Modificar valores: `df["col"] = df["col"] * 2`.
 - Renombrar columnas: `df.rename(columns={"col0": "columna0", "col1": "columna1", "col2": "columna2"})`.

2.8 Análisis de datos de opinión pública

- **Objetivo:** Aplicar Pandas para analizar un conjunto de datos real de opinión pública en Uruguay.
- **Contenidos:**
 - Importar datos de un repositorio de GitHub.
 - Explorar y limpiar los datos.
 - Realizar análisis descriptivos y agrupaciones.
 - Filtrar y seleccionar datos relevantes.
 - Visualizar resultados con tablas y gráficos.

2.9 Análisis de encuestas

- **Objetivo:** Aplicar Pandas para analizar datos de encuestas de Stackoverflow.
- **Contenidos:**
 - Importar datos de encuestas.
 - Limpiar y preprocesar datos.
 - Separar respuestas múltiples en columnas.
 - Realizar análisis con `crosstab()`:
 - `pd.crosstab(df['col1'], df['col2'], margins = True)`.
 - Formatear y visualizar resultados.

Ejercicio 2: Este ejercicio tiene como principal objetivo poner en práctica lo aprendido en el módulo 2. Realiza el ejercicio siguiendo la consigna, ten en cuenta que hay varias maneras de resolverlo. Por cualquier consulta no dudes escribir en el foro.

- [Google Drive](#)
- [GitHub](#)

Módulo 3: Visualización de datos con Python

3.1 Introducción a la visualización de datos

- **Objetivo:** Comprender la importancia de la visualización de datos y su rol en el análisis de datos.
- **Contenidos:**
 - ¿Por qué visualizar datos?
 - Tipos de gráficos y sus usos.
 - Librerías de visualización en Python: Matplotlib, Seaborn, Plotly.

3.2 Matplotlib

- **Objetivo:** Aprender a crear visualizaciones básicas con Matplotlib.
- **Contenidos:**
 - Crear gráficos de barras: `plt.bar(choice['names'], choice['counts'])`.
 - Personalizar gráficos: colores, etiquetas, títulos.
 - Crear gráficos de torta: `plt.pie(choice['counts'], labels = choice['names'])`.

3.3 Seaborn

- **Objetivo:** Aprender a crear visualizaciones más avanzadas con Seaborn.
- **Contenidos:**
 - Cargar datasets de ejemplo: `df = sns.load_dataset('tips')`.
 - Crear gráficos de barras: `sns.barplot(data = df, x = 'time', y = 'total_bill')`.
 - Crear gráficos de conteo: `sns.countplot(x = 'sex', data = df)`.
 - Personalizar gráficos: temas, estilos, etiquetas.
 - Crear gráficos de líneas: `sns.lineplot(data = covid2021, x = 'mes', y = 'cantCasosNuevosOriginal')`.

3.4 Plotly

- **Objetivo:** Aprender a crear visualizaciones interactivas con Plotly.
- **Contenidos:**
 - Cargar datasets de ejemplo: `df = px.data.tips()`.
 - Crear histogramas: `px.histogram(df, x = 'sex', y = "total_bill")`.
 - Crear gráficos de barras: `px.bar(df, x = 'year', y = 'pop')`.
 - Crear gráficos de líneas: `px.line(df, x = 'year', y = 'lifeExp', color = 'country')`.
 - Personalizar gráficos: etiquetas, títulos, colores, interactividad.

Ejercicio 3: Este ejercicio tiene como principal objetivo poner en práctica lo aprendido en el módulo 3. Realiza el ejercicio siguiendo la consigna, ten en cuenta que hay varias maneras de resolverlo. Por cualquier consulta no dudes escribir en el foro.

- [Google Drive](#)
- [GitHub](#)

Módulo 4: Análisis de texto - Webscrapping

4.1 Introducción al Web Scraping

- **Objetivo:** Comprender el concepto de web scraping y su utilidad.
- **Contenidos:**
 - ¿Qué es web scraping?
 - Aplicaciones del web scraping.
 - Ética y legalidad del web scraping.
 - Introducción a las librerías `requests` y `BeautifulSoup`.

4.2 Scraping simple con BeautifulSoup

- **Objetivo:** Aprender a extraer información básica de una página web.
- **Contenidos:**
 - Realizar una petición GET con `requests: page = requests.get(url)`.
 - Parsear el HTML con BeautifulSoup: `soup = BeautifulSoup(page.content, 'html.parser')`.
 - Encontrar elementos HTML por tag y clase: `soup.find_all('h3', class_='list-item-header')`.
 - Extraer texto de elementos HTML.
 - Almacenar la información extraída.

4.3 Scraping complejo y DataFrames

- **Objetivo:** Aprender a extraer información más compleja y estructurarla en un DataFrame.
- **Contenidos:**
 - Extraer información de tablas HTML.
 - Iterar sobre elementos HTML.
 - Convertir datos a tipos de datos adecuados.
 - Crear un DataFrame con Pandas: `pd.DataFrame(movies_list, columns=['Title', 'Year', 'Rating'])`.
 - Guardar el DataFrame en un archivo CSV.

4.4 Introducción al análisis de texto

- **Objetivo:** Comprender los conceptos básicos del análisis de texto.
- **Contenidos:**
 - ¿Qué es el análisis de texto?
 - Aplicaciones del análisis de texto.
 - Tokenización de texto: `word_tokenize(lyrics)`.
 - Distribución de frecuencia de palabras: `nlTK.FreqDist(tokens)`.

4.5 Nube de palabras

- **Objetivo:** Aprender a crear una nube de palabras a partir de un texto.
- **Contenidos:**
 - Eliminar puntuación y stop words.
 - Generar una nube de palabras con `WordCloud`.
 - Visualizar la nube de palabras con `Matplotlib`.

4.6 Análisis de sentimientos

- **Objetivo:** Aprender a realizar análisis de sentimientos en un texto.
- **Contenidos:**
 - Usar `SentimentIntensityAnalyzer` para obtener la polaridad del sentimiento.
 - Interpretar los resultados del análisis de sentimientos.

Ejercicio 4: Este ejercicio tiene como principal objetivo poner en práctica lo aprendido en el módulo 4. Realiza el ejercicio siguiendo la consigna, ten en cuenta que hay varias maneras de resolverlo. Por cualquier

consulta no dudes escribir en el foro.

- [Google Drive](#)
 - [GitHub](#)
-