

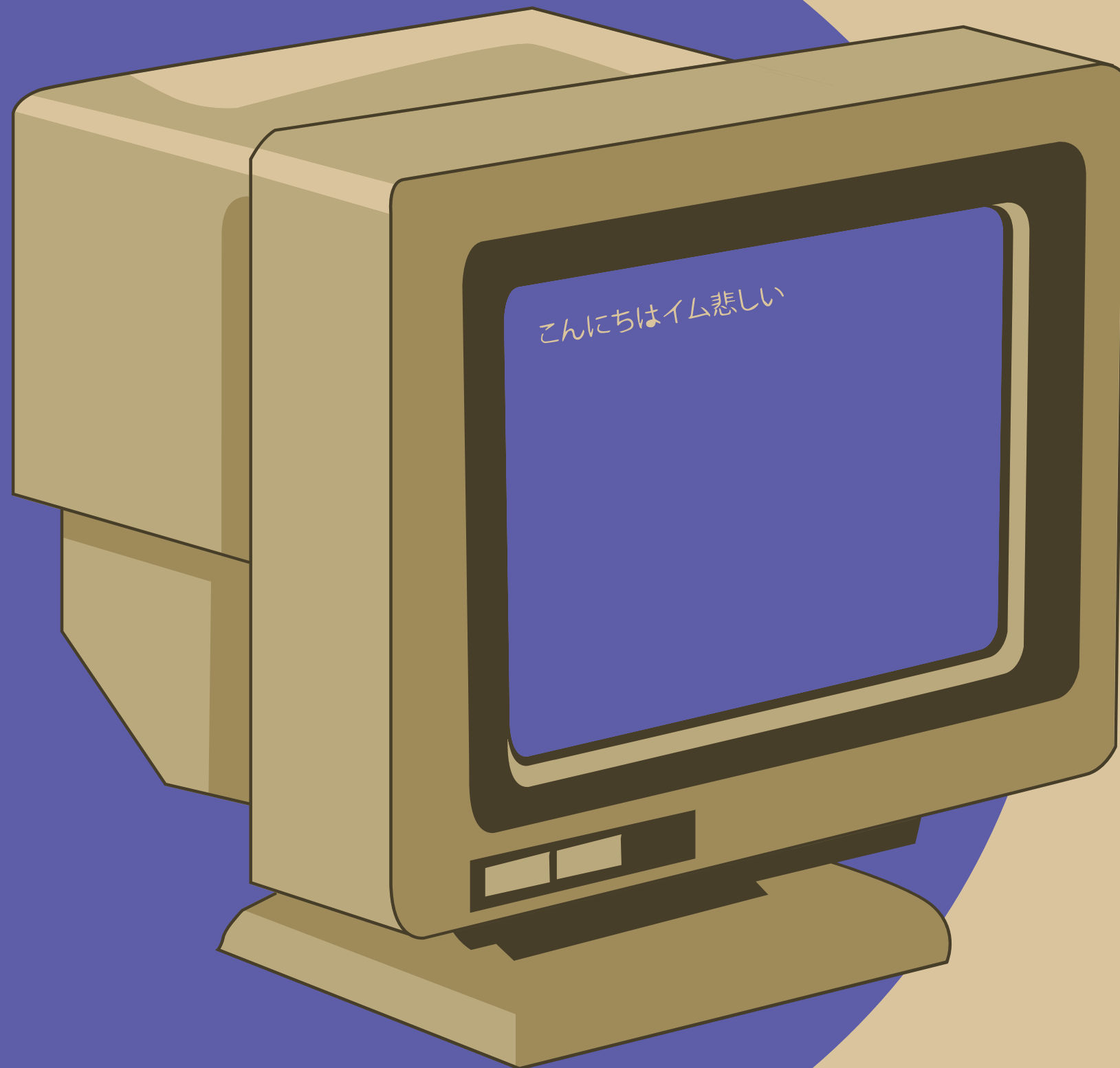


PROGRAMACION COMPUTACIONAL III

LIBRERIAS EXTERNAS DE PYTHON

INTEGRANTES:

- IVETTE AZUCENA MENDIOLA REQUENO.
- FLAVIO ELIAS REYES PAZ
- BRYAN ENRIQUE TORRES ALVAREZ.





INTRODUCCION

Pandas es una biblioteca de Python diseñada para facilitar el análisis y manipulación de datos, proporcionando estructuras eficientes y herramientas para trabajar con datos tabulares y etiquetados. Es ampliamente utilizada en campos como la ciencia de datos, análisis financiero y procesamiento de grandes volúmenes de datos.





CONCEPTOS CLAVES

- Series: Estructura unidimensional, similar a una columna de una tabla, donde cada valor tiene un índice asociado.
- DataFrame: Estructura bidimensional (como una hoja de cálculo), que organiza los datos en filas y columnas, siendo el objeto principal de trabajo en pandas.

python

```
import pandas as pd

# Crear una Serie
serie = pd.Series([10, 20, 30, 40], index=['a', 'b', 'c', 'd'])
print(serie)
```

python

```
# Crear un DataFrame
data = {'Nombre': ['Ana', 'Luis', 'Carmen', 'Juan'],
        'Edad': [28, 34, 29, 42],
        'Ciudad': ['Madrid', 'Barcelona', 'Sevilla', 'Bilbao']}

df = pd.DataFrame(data)
print(df)
```



FUNCIONES PRINCIPALES

```
df = pd.read_csv('archivo.csv')
```

```
df.to_csv('archivo_salida.csv', index=False)
```

- Lectura y escritura de datos desde CSV:

```
df['Nombre']
```

```
df[['Nombre', 'Edad']]
```

- Selección de Datos

```
# Seleccionar las primeras 2 filas
```

```
df.iloc[:2]
```

```
# Filtrar por condición
```

```
df[df['Edad'] > 30]
```



PROGRAMACION COMPUTACIONAL III

- Agrupación y agregación.
- Operaciones con filas y columnas.
- Descripción grafica de los datos.

```
# Agrupar por ciudad y calcular la edad promedio  
df.groupby('Ciudad')['Edad'].mean()
```

```
# Agregar una nueva columna 'Años para jubilación' basada en la columna 'Edad'  
df['Años para jubilación'] = 65 - df['Edad']  
  
# Eliminar una columna  
df.drop(columns=['Ciudad'])
```

```
df.describe()
```



CASOS DE USO

1.

Análisis financiero: Calcular indicadores como la media móvil o la varianza a partir de datos bursátiles históricos.

2.

Preparación de datos para Machine Learning: Limpieza y transformación de datos antes de aplicarlos en modelos predictivos.

3.

Exploración de datos: Visualización y análisis de grandes conjuntos de datos para descubrir patrones y tendencias.

```
# Cargar datos de precios
df = pd.read_csv('datos_financieros.csv')

# Calcular media móvil de 50 días
df['Media_Movil_50'] = df['Cierre'].rolling(window=50).mean()
```

```
# Identificar valores nulos
df.isnull().sum()

# Rellenar valores nulos con la media
df['Columna'].fillna(df['Columna'].mean(), inplace=True)
```

```
# Contar cuántas personas viven en cada ciudad
df['Ciudad'].value_counts()

# Visualizar el rango de edades
df['Edad'].plot(kind='hist')
```



DATOS RELEVANTES.



Arquitectura y diseño eficiente: pandas está construido sobre NumPy, lo que le permite realizar operaciones rápidas y eficientes en grandes conjuntos de datos numéricos.

Características Claves:

Alto rendimiento: pandas maneja grandes volúmenes de datos con un consumo eficiente de memoria y tiempo, lo que la hace adecuada tanto para conjuntos de datos pequeños como para big data.

Flexible y adaptable: Se puede trabajar con distintos tipos de datos, como números, texto, fechas y categorías, y transformar los datos de diferentes formas (filtrado, pivoteo, fusión, etc.).

Operaciones vectorizadas: pandas aplica operaciones matemáticas y lógicas sobre columnas completas o subconjuntos de datos de forma optimizada, eliminando la necesidad de escribir bucles.



DATOS RELEVANTES.



Ventajas y Desventajas

Ventajas:

- Gran capacidad de manejo de datos heterogéneos y estructurados.
- Amplias funcionalidades de limpieza y transformación.
- Facilidad de integración con otras bibliotecas del ecosistema de Python.

Desventajas:

- El rendimiento en conjuntos de datos extremadamente grandes puede volverse limitado en comparación con bibliotecas especializadas como Dask o herramientas de big data.



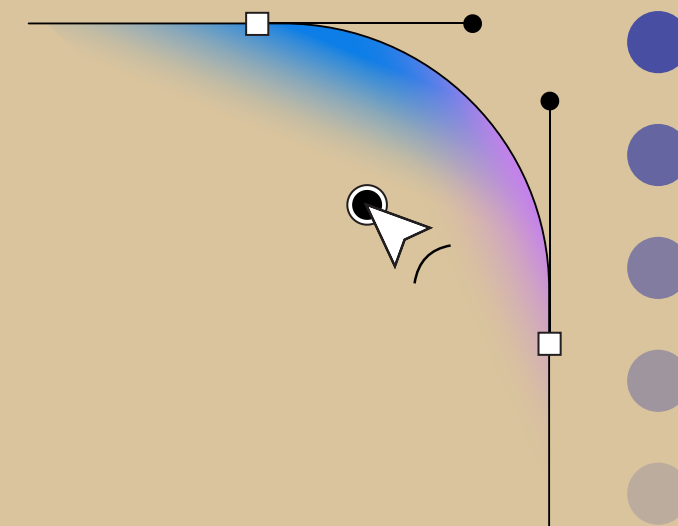
INSTALACION

Instalación con pip (para entornos de Python estándar), pip es el gestor de paquetes por defecto en Python,. Para instalar pandas con pip, sigue estos pasos:

```
pip install pandas
```

1. Abre una terminal o ventana de comandos en tu sistema operativo.

Esto instalará la versión más reciente de pandas disponible y sus dependencias, como numpy (una biblioteca esencial para operaciones matemáticas eficientes).





CONCLUSION

Pandas es una herramienta poderosa que ofrece una gran flexibilidad para el manejo, análisis y visualización de datos. Es un componente esencial del ecosistema de Python para científicos de datos y analistas debido a su capacidad de manejar datos complejos y realizar tareas como lectura/escritura de datos, limpieza, transformación, y análisis estadístico de manera eficiente.





EJEMPLO



**THANK
YOU!**

