

# Programación para Física y Astronomía

Departamento de Física.

---

Corodinadora: C Loyola

Profesores C Femenías / F Bugini / D Basantes

Primer Semestre 2025

Universidad Andrés Bello

Departamento de Física y Astronomía



Introducción y Contexto

Resumen de Contenidos Clave

Ejercicios de Repaso

Actividad en Clase

Conclusiones y Preparación

# Introducción y Contexto

---

- **Semana 8:**
  - POO básica (clases, atributos, métodos, `__init__`).
  - Integración con NumPy/pandas para datos, problemas evaluados.
- **Semana 9:**
  - Herencia, métodos especiales (`__str__`, `__repr__`).
  - Ejercicios sobre **Body/Star, Particle/ChargedParticle**, etc.
  - También tuvimos evaluaciones parciales en grupos (POO + datos).
- **Objetivo de hoy:** Repasar de forma integral todos los contenidos recientes, resolver ejercicios previos a la **Solemne II** (próxima sesión).

- **Revisar y ejercitar** los conceptos clave:
  - POO (clases, herencia, métodos).
  - NumPy avanzado (manipulación de arreglos, linalg).
  - Matplotlib (subplots, histogramas, 3D).
  - pandas (lectura de CSV, manejo de DataFrame básico).
- **Resolver** ejercicios integrales que unan estos temas.
- **Prepararnos** para la **Solemne II** (Semana 10, Sesión 2), revisando dudas y áreas de dificultad.

## Resumen de Contenidos Clave

---

- Clases y Objetos: `class Nombre :` y `obj = Nombre(...)`.
- `__init__` (constructor) para inicializar atributos.
- Herencia:  
`class Derivada(BaseClass):...super().__init__.`
- Métodos especiales: `__str__`, `__repr__`, etc.
- Ejemplos: `Particle`, `ChargedParticle`, `Body`, `Star`.

- `ndarray` con creación (`np.array`), `zeros`, `ones`, `arange`, `linspace`.
- `reshape`, `transpose`, `concatenate` (`hstack`, `vstack`).
- `np.random` (valores aleatorios), `np.linalg` (inversa, determinante, eigenvalores).
- Broadcasting y operaciones vectorizadas (suma, resta, multiplicación, etc. sin bucles).



- `plt.plot(x, y)`, `plt.scatter()`, `plt.hist()`, `plt.bar()`.
- `subplots()` para múltiples paneles (axs).
- Gráficos 3D con `Axes3D` o `subplots(projection='3d')`.
- Personalización:  
`xlabel`, `ylabel`, `title`, `legend`, `colorbar`.
- `plt.tight_layout()` para ajustar márgenes.

- `import pandas as pd.`
- Lectura de CSV: `df = pd.read_csv("archivo.csv").`
- `df.head()`, `df.describe()`, `df.columns`.
- Selección de columnas: `df['col']`, `df[['col1','col2']]`.
- Iteración de filas: `df.iterrows()` o `df.itertuples()`.
- Integración con **NumPy** y **Matplotlib** para análisis y graficación.

## Ejercicios de Repaso

---

### Enunciado

- Crear una clase `Point` con atributos (`name`, `x`, `y`).
- Método `distance(self, other)` que devuelva  $\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$ .
- Instanciar 5 objetos `Point` (pueden ser aleatorios).
- Crear `np.zeros((5,5))` y rellenar con las distancias entre cada par de `Point`.
- Visualizar la matriz en `plt.imshow` con `colorbar()`.

**Objetivo:** Reforzar `clase simple`, `distancia` y `matriz NxN` con NumPy + visualización.

### Enunciado

- Crear clase base `Particle` con (`mass`, `x`, `y`) y método `kinetic_energy(vx,vy)`.
- Crear clase derivada `ChargedParticle` con `charge` y método `potential_energy(E_field)`.
- Instanciar 3-4 `ChargedParticle` con datos aleatorios (usando `np.random`).
- Graficar **scatter** en 2D (eje X, Y) coloreado según `charge` (p. ej. `c=charge + colormap`).
- Mostrar **title**, **xlabel**, **ylabel**.

**Objetivo:** Reforzar **herencia**, uso de `np.random` y **scatter** con color mapping.

# Ejercicios de Repaso ∈ Ejercicio 3: DataFrame, Análisis y Subplots

## Enunciado

- Suponiendo un CSV (`measurements.csv`) con columnas `name`, `value`, `category`.
- Cargar con `pandas.read_csv`.
- Crear un **subplot** con 2 paneles:
  - Panel 1: **histograma** de la columna `value`.
  - Panel 2: **barra** de `value` por `name`, separando o coloreando por `category` (si cabe).
- Mostrar `df.describe()` para ver estadísticas rápidas.

**Objetivo:** Integrar `pandas` (lectura + describe) con **subplots** de Matplotlib.

## Ejercicios de Repaso ∈ Ejercicio 4 (Opcional): Cargar Objetos desde CSV

### Enunciado

- CSV con `name`, `mass`, `charge`, `x`, `y`.
- Clase `ChargedParticle` (hereda de `Particle`).
- Iterar filas del DF para instanciar objetos en una lista `particles`.
- Calcular la energía cinética total asumiendo `vx`, `vy` aleatorios o fijos.
- **Visualizar** la distribución de `charge` en un histograma.

**Objetivo:** Profundizar la **creación masiva** de objetos con datos CSV, combinando **NumPy** y **Matplotlib**.

## Actividad en Clase

---



- **Parejas o tríos**, elijan 2+ ejercicios o combínenlos.
- **Objetivo**: Repasar antes de la Solemne II.
- Pueden usar **Colab** o local, comentando sus pasos y mostrando resultados (gráficas, matrices).
- Comparen resultados y listas de **dudas** para plantearlas al final.

- Revisen **módulos** importados: `import numpy as np,`  
`import matplotlib.pyplot as plt,`  
`import pandas as pd.`
- Prueben **logging** básico o **print debugging** si algo sale mal.
- Usen **super()** y verifiquen si **\_\_str\_\_** es útil para imprimir objetos.
- Si hacen **matrices NxN**, consideren `np.fill_diagonal(mat, 0)` para poner ceros en la diagonal, si es relevante.

- ¿Preguntas sobre POO, herencia, métodos especiales?
- ¿Dificultades con `np.linalg` o `numpy.random`?
- ¿Matplotlib subplots, 3D, personalización?
- `pandas`: `df.describe`, `df.iterrows`, merges, etc.

Levanten la mano para aclarar cualquier duda.

## Conclusiones y Preparación

---

- Compartan cómo resolvieron **distancias NxN**, **scatter con color**, **subplots**, etc.
- Si usaron **pandas**, ¿cómo filtraron datos o manipularon columnas?
- ¿Qué inconvenientes aparecieron en la parte POO?

- Repasamos de forma integral **POO**, NumPy, Matplotlib y pandas con ejercicios.
- Identificamos **dudas** y resolvimos problemas típicos.
- Todo esto apunta a la **Solemne II** (próxima sesión), donde se evaluarán estos bloques de contenidos.

## Conclusiones y Preparación ∈ Preparación para la Solemne II (Semana 10, Sesión 2)

- **Revisar** apuntes y ejercicios hechos:
  - Sintaxis de **class**, herencia, `__init__`, `__str__`.
  - Principales funciones NumPy: **reshape**, **random**, **linalg**.
  - **Matplotlib**: subplots, scatter, hist, 3D, personalización básica.
  - **pandas** para CSV y `df.plot` (opcional).
- **Practicar** resolviendo miniproblemas con tiempo.
- Cualquier duda final, ¡pregunten vía foros o en la próxima clase (antes de la Solemne)!

- **Python Tutorial - Clases** (Repaso POO).
- **NumPy Docs** (subsecciones de random, linalg).
- **Matplotlib Docs** - Ejemplos de subplots y 3D.
- **pandas Docs** - sección de 10 minutes to pandas.



# Muchas gracias y éxito en su práctica

- Recuerden estudiar para la **Solemne II** (Semana 10, Sesión 2).
- Cualquier duda, sigan participando en foros o consultas.
- ¡Nos vemos en la próxima sesión con la evaluación!