

UNIVERSIDAD MAYOR REAL Y PONTIFICIA DE SAN FRANCISCO XAVIER

Facultad de Tecnología



Practico 01 SIS420

Tabla de estaturas y pesos en python

Universitario (a): Hector Daniel Vargas Gonzales

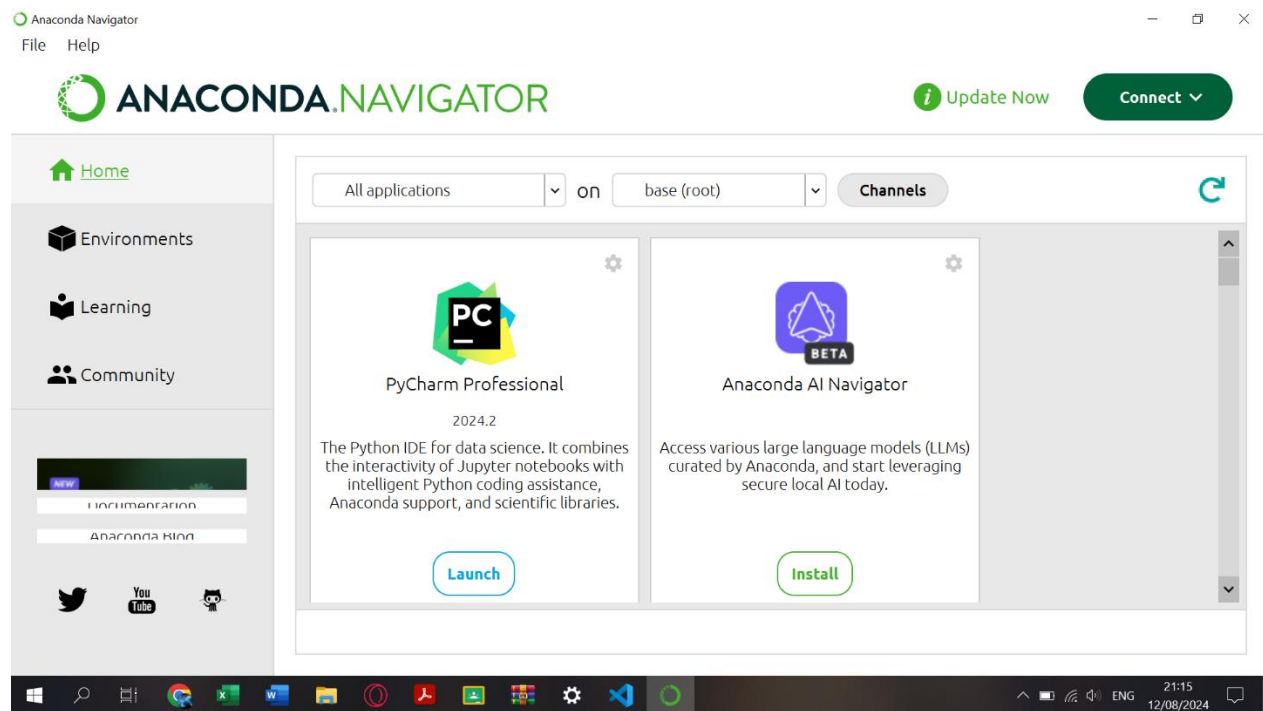
Carrera: Ingenieria de Sistemas

Docente: Carlos Walter Pacheco Lora

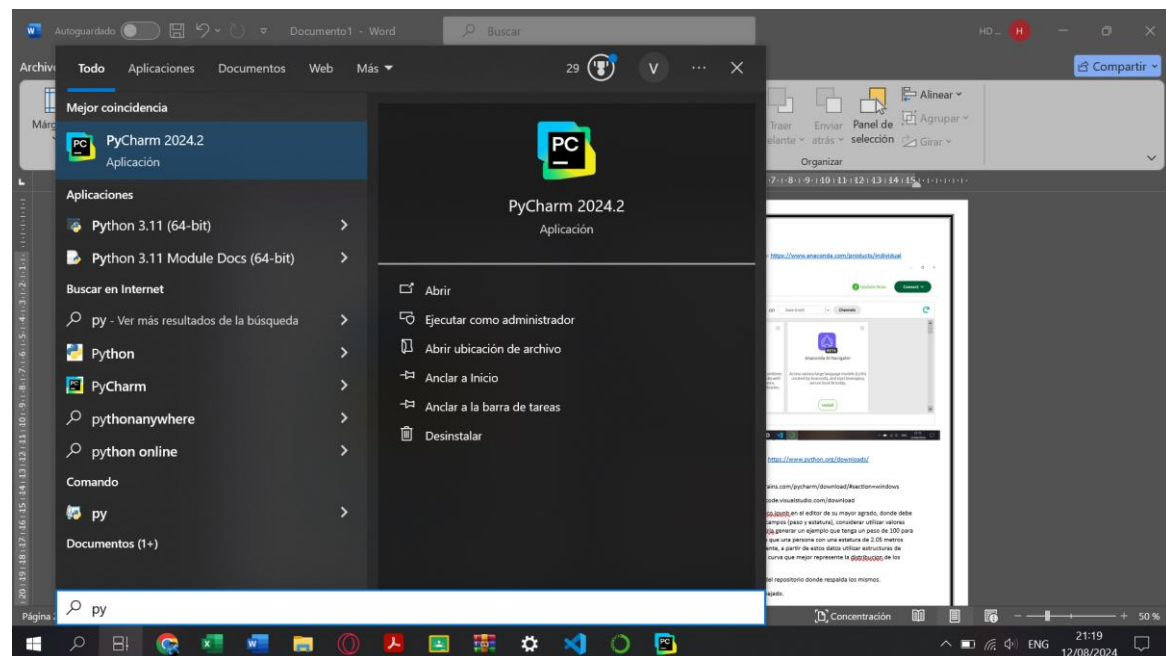
Materia: Inteligencia Artificial

Fecha: 12 de ago. de 24

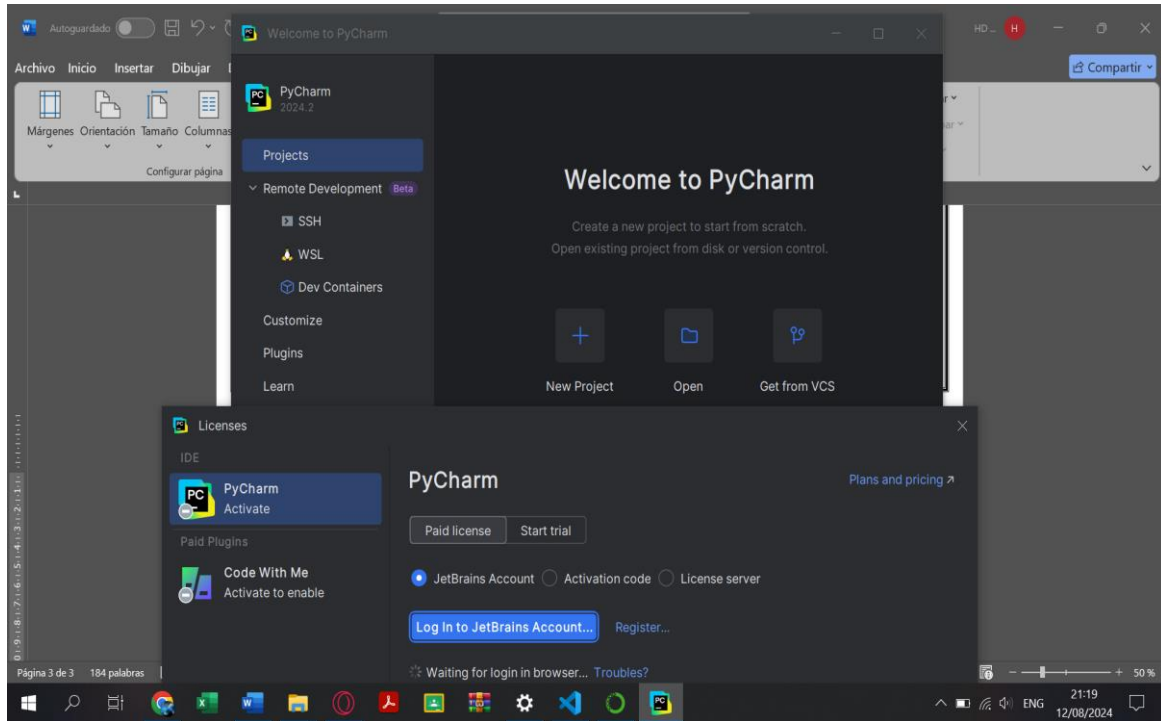
1. Instalar Anaconda edición individual de: <https://www.anaconda.com/products/individual>



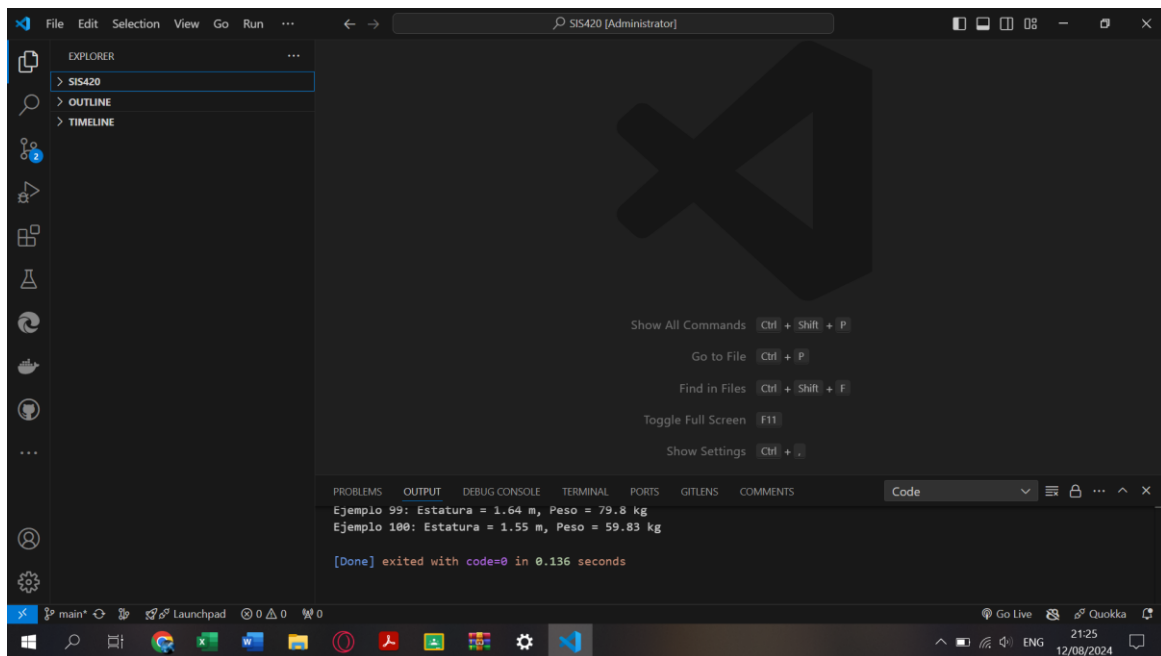
2. Instalar la ultima version de Python de: <https://www.python.org/downloads/>



3. Instalar Pycharm de: <https://www.jetbrains.com/pycharm/download/#section=windows>



4. Instalar Visual Studio Code de: <https://code.visualstudio.com/download>



5. Crear un archivo practico01.py y practico.ipynb en el editor de su mayor agrado, donde debe crear una tabla de 100 ejemplos con dos campos (peso y estatura), considerar utilizar valores aleatorios controlados, es decir no se podría generar un ejemplo que tenga un peso de 100 para una estatura de 0.80 metros o a la inversa que una persona con una estatura de 2.05 metros tenga un peso de 30 kilos y así sucesivamente, a partir de estos datos utilizar estructuras de control repetitivo para lograr encontrar la curva que mejor represente la distribución de los puntos generados.

Practico01.py

```
import random
def generar_datos(n=100):
    datos = []
    for _ in range(n):
        estatura = round(random.uniform(1.5, 2.0), 2)
        peso = round(random.uniform(50, 100), 2)
        if estatura < 1.6 and peso > 80:
            peso = round(random.uniform(50, 80), 2)
        elif estatura > 1.9 and peso < 70:
            peso = round(random.uniform(70, 100), 2)
        datos.append((estatura, peso))
    return datos
datos = generar_datos()
for i, (estatura, peso) in enumerate(datos[:100]):
    print(f"Ejemplo {i+1}: Estatura = {estatura} m, Peso = {peso} kg")
```

Practico.ipynb

```
import random
import matplotlib.pyplot as plt
def generar_datos(n=100):
    datos = []
    for _ in range(n):
        estatura = round(random.uniform(1.5, 2.0), 2) # Estatura en metros
        peso = round(random.uniform(50, 100), 2) # Peso en kg
        # Controlamos que los datos sean razonables
        if estatura < 1.6 and peso > 80:
            peso = round(random.uniform(50, 80), 2)
        elif estatura > 1.9 and peso < 70:
            peso = round(random.uniform(70, 100), 2)
        datos.append((estatura, peso))
    return datos
datos = generar_datos()
estaturas, pesos = zip(*datos)
plt.scatter(estaturas, pesos)
plt.xlabel('Estatura (m)')
```

```
plt.ylabel('Peso (kg)')
plt.title('Distribución de Peso vs Estatura')
plt.show()
```

6. Adjuntar los archivos creados ademas del repositorio donde respalda los mismos.

<https://github.com/HD-VG/SIS420.git>

7. Incluir un resumen que explique lo trabajado.

Una instalado todas las aplicaciones necesarias como visual studio code y Python se creo el archivo Practico01.py en el que se realizo lo siguiente:

- Importar la librería random
- Se crea una función llamada generar datos con un paramatro de n=100
- Se crea un array llamado datos que inicialmente esta vacio
- Se crea un ciclo en el cual el cual repetirá por n veces el ciclo el _ indica el contador, al ser un _ no se lo puede usar en el código como un valor
- Luego se crea 2 variables estatura y peso los cuales son una función round es decir que se agregan valores aleatorios entre los paramatros establlecidos
- Seguidanmente se tiene in if el cual asegura que las edades como los pesos esten entre rangos lógicos y asi trabajr con valores reales
- Después se añade al final del contenedor datos ambas variables obtenidas aleatoriamente y se retornan los datos de la función.
- Luego se llama a la funcon generar_datos() y se muestran los datos generados en la terminal

En el archivo practico.ipynb se importo la librería matplotlib.pyplot usando como variable plt, se instalo en vsc esta librería una ves obtenidos los datos con los pasos anteriores se pasa a usar el graficador el en eje x esta la estatura y el eje y esta el peso y por ultimo se dibija la grafica