

## Tarea para SBD04

### Título de la tarea: Manejo de DataFrames con Python/Pandas

### Curso de Especialización: Inteligencia Artificial y Big Data

### Módulo profesional: Sistemas de Big Data

### ¿Qué te pedimos que hagas?

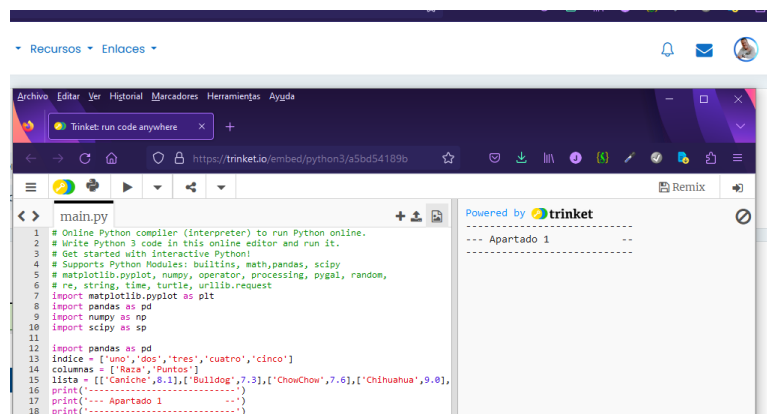
A continuación tienes la primera parte del programa de Enrique.

Accede al [intérprete de Python en trinket.io](https://trinket.io/embed/python3/a5bd54189b), pega el código fuente que te entregamos y ejecútalo (con el botón del triángulo negro de la parte superior).

```
import pandas as pd

indice = ['uno', 'dos', 'tres', 'cuatro', 'cinco']
columnas = ['Raza', 'Puntos']
lista = [[ 'Caniche', 8.1], [ 'Bulldog', 7.3], [ 'ChowChow', 7.6], [ 'Chihuahua', 9.0], [ 'Labrador', 9.3]]

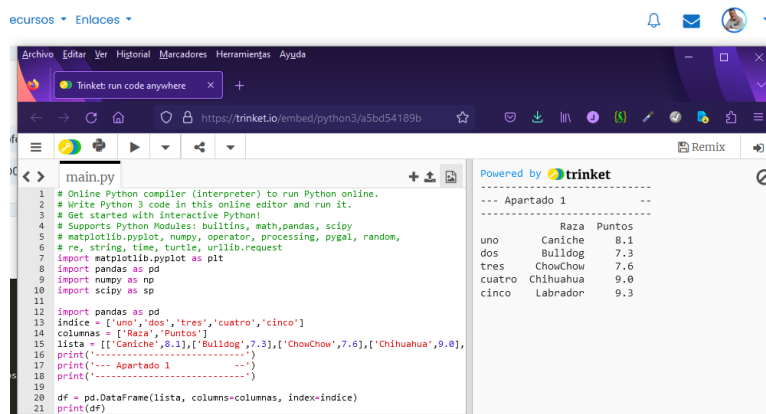
print('-----')
print('--- Apartado 1 ---')
print('-----')
```



- **Apartado 1:**

Crea un DataFrame llamado **df** a partir de la lista de razas y puntuaciones otorgadas por Enrique, usando el índice, las columnas que se te proporcionan.

```
df = pd.DataFrame(lista, columns=columnas, index=indice)
print(df)
```



- **Apartado 2:**

Ana le entrega sus puntuaciones a Enrique y éste las transcribe en forma de la siguiente lista:

```
puntos_ana = [61,75,82,95,99]
```

Añade la columna con etiqueta 'PuntosAna' al DataFrame.

```
puntos_ana = [61,75,82,95,99]
df['PuntosAna'] = puntos_ana
print(df)
```

```
main.py
1 # Online Python compiler (interpreter) to run Python online.
2 # Write Python 3 code in this online editor and run it.
3 # Get started with interactive Python!
4 # Supports Python Modules: builtins, math, pandas, scipy
5 # matplotlib.pyplot, numpy, operator, processing, pygame, random,
6 # re, string, time, turtle, urllib.request
7 import matplotlib.pyplot as plt
8 import pandas as pd
9 import numpy as np
10 import scipy as sp
11
12 import pandas as pd
13 indice = ['uno','dos','tres','cuatro','cinco']
14 columnas = ['Raza','Puntos']
15 lista = [['Caniche',8.1],['Bulldog',7.3],['ChowChow',7.6],['Chihuahua',9.0],
16          ['Labrador',9.3]]
17 print('----- Apartado 1 -----')
18 print('-----')
19 df = pd.DataFrame(lista, columns=columnas, index=indice)
20 print(df)
21
22 # Apartado 2
23 print('----- Apartado 2 -----')
24 print('-----')
25 #df = pd.DataFrame(lista, columns=columnas, index=indice)
26 puntos_ana = [61,75,82,95,99]
27 df['PuntosAna'] = puntos_ana
28 print(df)
```

	Raza	Puntos	PuntosAna
uno	Caniche	8.1	61
dos	Bulldog	7.3	75
tres	ChowChow	7.6	82
cuatro	Chihuahua	9.0	95
cinco	Labrador	9.3	99

- **Apartado 3:**

Como las puntuaciones de Ana han entrado en valores de 0 a 100 y las de Enrique estaban de 0 a 10, divide la columna 'PuntosAna' entre 10 de modo que todas las puntuaciones queden en la misma escala.

```
df['PuntosAna'] = np.array(puntos_ana)/10
print(df)
```

```
main.py
32 # Apartado 3
33 print('----- Apartado 3 -----')
34 print('-----')
35 df['PuntosAna'] = np.array(puntos_ana)/10
36 print(df)
```

	Raza	Puntos	PuntosAna
uno	Caniche	8.1	6.1
dos	Bulldog	7.3	7.5
tres	ChowChow	7.6	8.2
cuatro	Chihuahua	9.0	9.5
cinco	Labrador	9.3	9.9

- **Apartado 4:**

Enrique y Ana han estado viendo una revista de animales y han decidido añadir otras dos razas más, por lo que han creado la siguiente lista ya con sus puntuaciones integradas.

```
lista2 = [['Samoyedo',9.2,8.9],['Pinscher',8.1,6.7]]
```

Añade esas dos nuevas filas al DataFrame. Fíjate en que se han quedado sin ideas para los índices y han decidido usar 'nuevo1' y 'nuevo2' para las nuevas filas.

```

lista2 = [['Samoyedo',9.2,8.9],['Pinscher',8.1,6.7]]
df2 = pd.DataFrame(lista2, columns=['Raza','Puntos','PuntosAna'],
index=['nuevo1','nuevo2'])
df = df.append(df2)
print(df)

```

```

uno Caniche 8.1 6.1
dos Bulldog 7.3 7.5
tres ChowChow 7.6 8.2
cuatro Chihuahua 9.0 9.5
cinco Labrador 9.3 9.9
--- Apartado 4 ---
Raza Puntos PuntosAna
uno Caniche 8.1 6.1
dos Bulldog 7.3 7.5
tres ChowChow 7.6 8.2
cuatro Chihuahua 9.0 9.5
cinco Labrador 9.3 9.9
nuevo1 Samoyedo 9.2 8.9
nuevo2 Pinscher 8.1 6.7

```

En este ejercicio, el editor avisa de un futuro paso a desuso en la función append().

- **Apartado 5:**

Crea la columna 'Media' para poder ver la puntuación media para cada raza.

```

x1 = df[['Puntos', 'PuntosAna']].mean(axis=1)
df['Media'] = x1.sort_index()
print(df)

```

```

tres ChowChow 7.6 8.2
cuatro Chihuahua 9.0 9.5
cinco Labrador 9.3 9.9
nuevo1 Samoyedo 9.2 8.9
nuevo2 Pinscher 8.1 6.7
--- Apartado 5 ---
Raza Puntos PuntosAna Media
uno Caniche 8.1 6.1 7.10
dos Bulldog 7.3 7.5 7.40
tres ChowChow 7.6 8.2 7.90
cuatro Chihuahua 9.0 9.5 9.25
cinco Labrador 9.3 9.9 9.60
nuevo1 Samoyedo 9.2 8.9 9.05
nuevo2 Pinscher 8.1 6.7 7.40

```

- **Apartado 6:**

Enrique y Ana deciden que no tendrán un bulldog. Elimina la fila cuyo índice es 'dos'.

```

df = df.drop('dos')
print(df)

```

```

dos Bulldog 7.3 7.5 7.40
tres ChowChow 7.6 8.2 7.90
cuatro Chihuahua 9.0 9.5 9.25
cinco Labrador 9.3 9.9 9.60
nuevo1 Samoyedo 9.2 8.9 9.05
nuevo2 Pinscher 8.1 6.7 7.40
--- Apartado 6 ---
Raza Puntos PuntosAna Media
uno Caniche 8.1 6.1 7.10
tres ChowChow 7.6 8.2 7.90
cuatro Chihuahua 9.0 9.5 9.25
cinco Labrador 9.3 9.9 9.60
nuevo1 Samoyedo 9.2 8.9 9.05
nuevo2 Pinscher 8.1 6.7 7.40

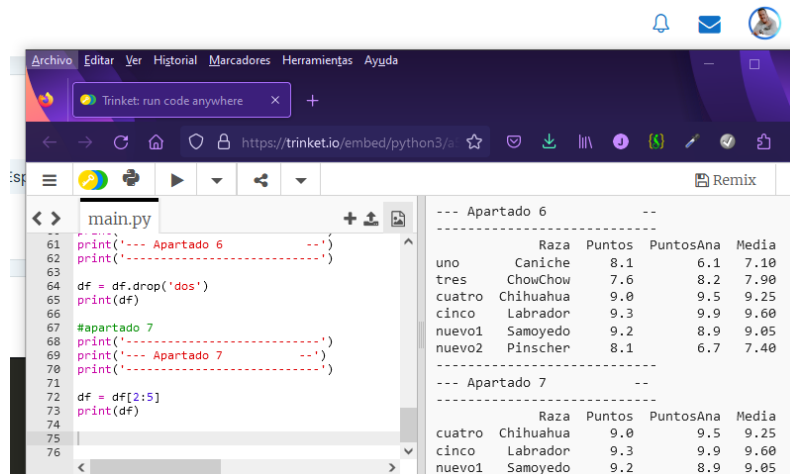
```

- **Apartado 7:**

Viendo las puntuaciones medias, Enrique y Ana deciden que la decisión va a estar entre chihuahua, labrador y samoyedo, por lo que deciden obtener sólo esas filas del DataFrame (en concreto desde la posición 2 hasta la 5 -no incluía-).

```
df = df[2:5]
```

```
print(df)
```



```

main.py
61 print('--- Apartado 6 ---')
62 print('')
63
64 df = df.drop('dos')
65 print(df)
66
67 #apartado 7
68 print('--- Apartado 7 ---')
69 print('')
70 print('')
71
72 df = df[2:5]
73 print(df)
74
75
76

```

```

--- Apartado 6 ---

```

	Raza	Puntos	PuntosAna	Media
uno	Caniche	8.1	6.1	7.10
tres	ChowChow	7.6	8.2	7.90
cuatro	Chihuahua	9.0	9.5	9.25
cinco	Labrador	9.3	9.9	9.60
nuevo1	Samoyedo	9.2	8.9	9.05
nuevo2	Pinscher	8.1	6.7	7.40

```

--- Apartado 7 ---

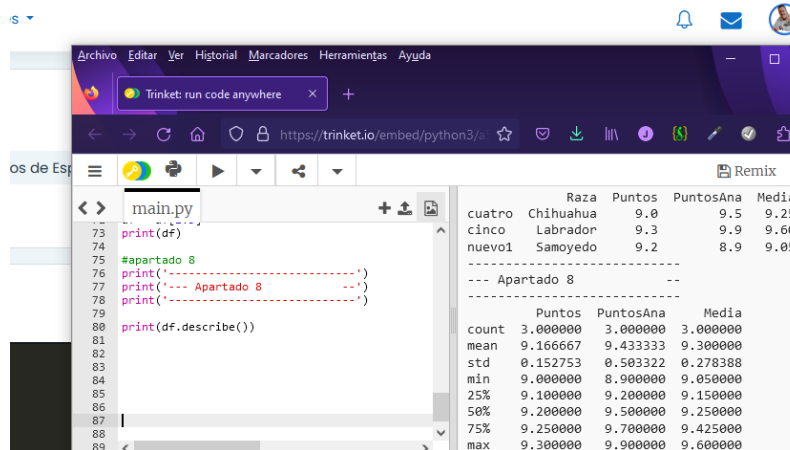
```

	Raza	Puntos	PuntosAna	Media
cuatro	Chihuahua	9.0	9.5	9.25
cinco	Labrador	9.3	9.9	9.60
nuevo1	Samoyedo	9.2	8.9	9.05

- **Apartado 8:**

Por último, sólo por curiosidad, deciden ver información estadística sobre las filas que les han quedado.

```
print(df.describe())
```



```

main.py
73 print(df)
74
75 #apartado 8
76 print('--- Apartado 8 ---')
77 print('')
78 print('')
79
80 print(df.describe())
81
82
83
84
85
86
87
88
89

```

```

--- Apartado 8 ---

```

	Puntos	PuntosAna	Media
count	3.000000	3.000000	3.000000
mean	9.166667	9.433333	9.300000
std	0.152753	0.503322	0.278388
min	9.000000	8.900000	9.050000
25%	9.100000	9.200000	9.150000
50%	9.200000	9.500000	9.250000
75%	9.250000	9.700000	9.425000
max	9.300000	9.900000	9.600000