



## MANUAL DE PROCEDIMIENTO

**CARRERA: COMPUTACIÓN**

**PRÁCTICA: Trabajo en clases**

**EQUIPO: COMPUTADOR**

**CÁTEDRA O MATERIA  
RELACIONADA**

**REVISIÓN N°: 1**

**Responsable / Equipo.:**

**Simulación**

**EDICIÓN: 1**

**Accesorios:**

COMPUTADOR / INTERNET

**DOCENTE:**

Diego Quisi

**NÚMERO DE ESTUDIANTES  
POR EQUIPO O PRÁCTICA:**

1

**Fecha: 14/11/2021**

**Tema: Graficas**

**Realizado por:** Alex Reinoso

**Objetivo:**

Utilizar Matplot y Plotly para realizar las gráficas.

### Ejercicio en Clase

**1. Con los datos de Fifa, organizar a los jugadores de acuerdo al peso en las siguientes escalas y generar un cuadro tipo PIE**

Debajo 125 Lbs.

125-150

150-175

175 o superior

**Desarrollo:**

```
jugador125 =  
fifa.loc[fifa['Weight'].str.replace("lbs", '.0').astype(float)  
< 125].count()[0]  
jugador150 =  
fifa.loc[(fifa['Weight'].str.replace("lbs", '.0').astype(float)  
>= 125) &  
(fifa['Weight'].str.replace("lbs", '.0').astype(float) <  
150)].count()[0]
```

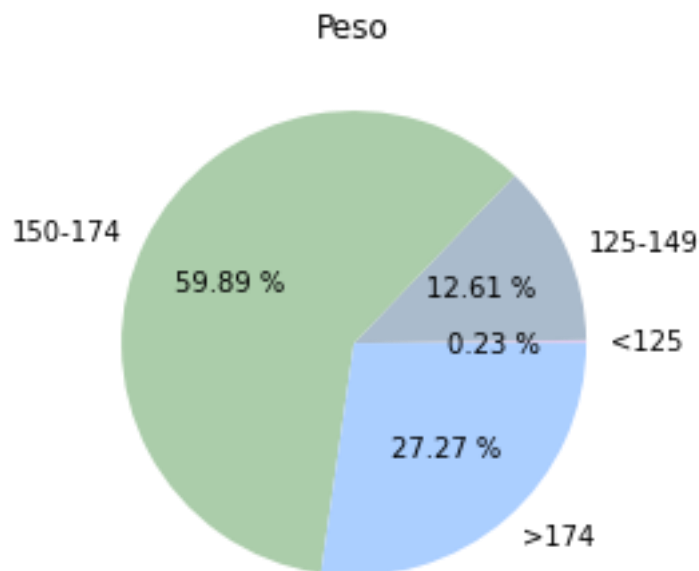
```

jugador175 =
fifa.loc[(fifa['Weight'].str.replace("lbs",'.0').astype(float)
>= 150) &
(fifa['Weight'].str.replace("lbs",'.0').astype(float) <
175)].count()[0]
mayor175 =
fifa.loc[(fifa['Weight'].str.replace("lbs",'.0').astype(float)
>= 175)].count()[0]

plt.figure()

etiquetas = ['<125', '125-149', '150-174', '>174']
colores = ['#ddcdef', '#aabbcc', '#abcdaa', '#abcfff']
plt.pie([jugador125, jugador150, jugador175, mayor175],
labels=etiquetas, colors=colores, autopct='%.2f %%')
plt.title('Peso')
plt.show()

```



**2. Generar un gráfico de barras (histograma) de acuerdo a su habilidad (Overall) en base a los siguientes segmentos contando el número de jugadores**

40  
50  
60

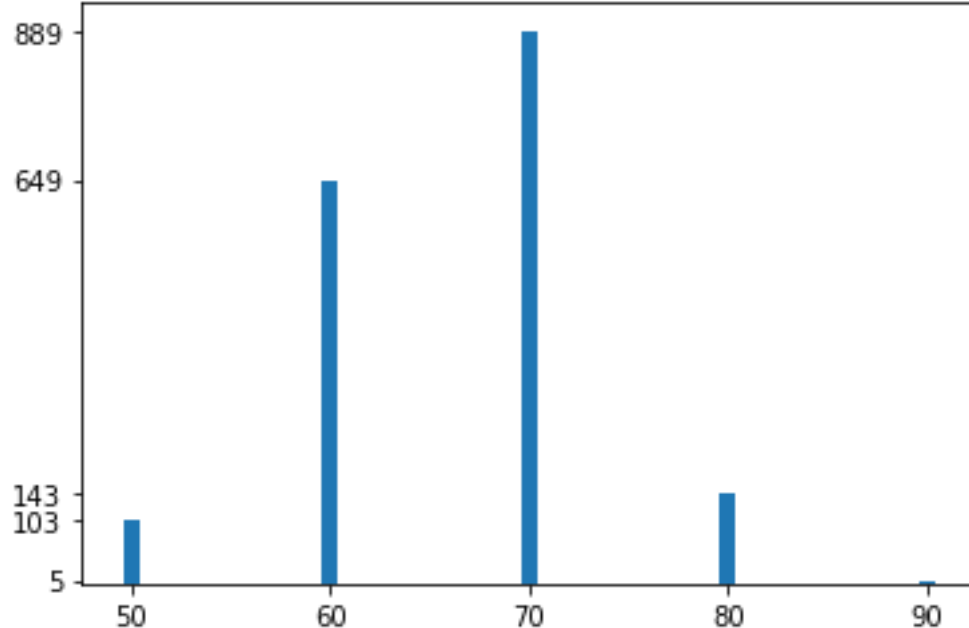
70  
80  
90

### Desarrollo:

```
df_overall = fifa.loc[(fifa['Overall'].astype(float) == 40) |  
(fifa['Overall'].astype(float) == 50) |  
(fifa['Overall'].astype(float) == 60) |  
(fifa['Overall'].astype(float) == 70) |  
(fifa['Overall'].astype(float) == 80) |  
(fifa['Overall'].astype(float) == 90)]  
df_overall = pd.DataFrame(df_overall['Overall'])  
df_overall.tail()
```

Overall	
18113	50
18114	50
18115	50
18116	50
18117	50

```
labels, counts = np.unique(df_overall, return_counts=True)  
plt.bar(labels, counts, align='center')  
plt.gca().set_xticks(labels)  
plt.gca().set_yticks(counts)  
plt.show()
```



### 3. Investigar alguna herramienta o software que permita generar reportes con Python utilizando CodeLabs.

#### Papermill

Esta es una librería de Python la cual permite parametrizar y ejecutar libros de Jupyter.

Esta herramienta permite ejecutar diferentes libros de jupyter dependiendo en la ejecución de uno, sin la necesidad de copiar y pegar los datos del uno al otro.

Es considerado el estándar de oro en cuanto al análisis exploratorio de datos, así también como una herramienta sólida para documentar proyectos de ciencia de datos.

#### Conclusiones

#### Referencias Bibliográficas

<https://papermill.readthedocs.io/en/latest/>

<https://towardsdatascience.com/introduction-to-papermill-2c61f66bea30>

A handwritten signature in blue ink, consisting of the letters 'A', 'R', and 'U' in a stylized, cursive-like font. The 'A' is formed by two intersecting diagonal lines, the 'R' is a simple vertical line with a small loop at the top, and the 'U' is a simple U-shape.

---

**Est. Alex Reinoso**