

## 2.2 PREPARACIÓN DE LOS DATOS EXTRACCIÓN DE CONOCIMIENTOS DE BASES DE DATOS INTEGRANTES

REYES ROMERO GUSTAVO

RODRIGUEZ VELASCO ELIAZAR

DE JESUS CABRERA CARLOS ARMANDO

Ejercicio: Formar un equipo de 3 integrantes máximo y realizar las actividades.

Crear una libreta de jupyter en Colab y realizar el análisis del dataset anexo.

Crear un repositorio de github con el nombre de la materia.

Almacenar la libreta de jupyter en el repositorio creado.

Realizar el reporte de su practica.

la primera parte de este analisis son los datos para eso debemos de subir el archivo  
ecommerce\_costumers.csv

una vez el archivo ecommerce\_costumers.csv este cargado nos dirigimos a importar las librerias  
que vamos a ocupar

### ▼ Importar librerias

```
1 import tensorflow as tf
2 import numpy as np
3 from tensorflow import keras
4 import pandas as pd
5 from google.colab import files
6 import matplotlib.pyplot as plt
7 print(np.__version__)
8 print(pd.__version__)
```

```
📄 1.19.5
    1.1.5
```

### ▼ Ver datos

leer datos e importarlos

```
1 files.upload()
```

Elegir archivos ecommerce\_customers.csv

- **ecommerce\_customers.csv**(application/vnd.ms-excel) - 71638 bytes, last modified: 8/10/2021 - 100% done

Saving ecommerce\_customers.csv to ecommerce\_customers.csv

```
{'ecommerce_customers.csv': b'Email,Address,Avatar,Avg. Session Length,Time on Ap
```

ver datos

```
1 analisis = pd.read_csv('ecommerce_customers.csv',delimiter=',',header='infer'  
2 analisis
```

```

      Address      Avatar      Avg. Session      Time on App      Time on Website      Length of Membership      Yearly Amount
1 !wget https://github.com/eliazar-rodriguez-velasco/Extraccion_conocimientos_b
2 df_customers = pd.read_csv('ecommerce_customers.csv',delimiter=',',header='in
3

--2021-10-11 02:34:23-- https://github.com/eliazar-rodriguez-velasco/Extraccion
Resolving github.com (github.com)... 140.82.113.3
Connecting to github.com (github.com)|140.82.113.3|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: unspecified [text/html]
Saving to: 'ecommerce_customers.csv.1'

ecommerce_customers      [ <=> ] 365.53K --.-KB/s    in 0.02s

2021-10-11 02:34:23 (17.0 MB/s) - 'ecommerce_customers.csv.1' saved [374302]

```



```

1 print(analysis.columns)

Index(['Email', 'Address', 'Avatar', 'Avg. Session Length', 'Time on App',
      'Time on Website', 'Length of Membership', 'Yearly Amount Spent'],
      dtype='object')

```

es Motorway

Haz doble clic (o pulsa Intro) para editar

```

... ..
1 # tipo de datos
2 df_customers.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 500 entries, 0 to 499
Data columns (total 8 columns):
#   Column                Non-Null Count  Dtype
---  ---
0   Email                 500 non-null    object
1   Address               500 non-null    object
2   Avatar               500 non-null    object
3   Avg. Session Length   500 non-null    float64
4   Time on App           500 non-null    float64
5   Time on Website       499 non-null    float64
6   Length of Membership  499 non-null    float64
7   Yearly Amount Spent   500 non-null    float64
dtypes: float64(5), object(3)
memory usage: 31.4+ KB

```

al momento de platicar con el equipo se llego a la conclusion que de acuerdo al problema que se plantea en el ejercicio que dice: la compañía esta tratando de decidir si enfocar sus esfuerzos con la aplicacion movil o el sitio web. se tomaron los siguientes campos como mas importantes avatar: el cual va relacionado con los diferentes tiempos o datos

time on app y time on website que son todos los tiempos relacionados con los diferentes

```
1 pd.DataFrame(analysis, columns = ["Avatar", "Time on App", "Time on Website"])
```

	Avatar	Time on App	Time on Website
0	Violet	12.655651	39.577668
1	DarkGreen	11.109461	37.268959
2	Bisque	11.330278	37.110597
3	SaddleBrown	13.717514	36.721283
4	MediumAquaMarine	12.795189	37.536653
...	...	...	...
495	Tan	13.566160	36.417985
496	PaleVioletRed	11.695736	37.190268
497	Cornsilk	11.499409	38.332576
498	Teal	12.391423	36.840086
499	DarkMagenta	12.418808	35.771016

500 rows × 3 columns

una vez analizados los campos se realizo el proceso para obtener el promedio el cual nos indica cual de las dos aplicaciones es la que se esta usando mas.

```
1
2 analisis['Time on Website'].mean()
```

37.059788368076156

```
1 analisis['Time on App'].mean()
2
```

12.052487936928012

de igual forma como otro dato se muestran mas campos como el maximo y el minimo que en este momoento solo no importa el time on app y time on website

el minimo en time on app fue de 8.5 y el maximo de 15.12 y en time on website el minimo fue de 33.91 y el maximo de 40.00

```
1 df_customers.describe()
```

	Avg. Session Length	Time on App	Time on Website	Length of Membership	Yearly Amount Spent
<b>count</b>	500.000000	500.000000	499.000000	499.000000	500.000000
<b>mean</b>	33.053194	12.052488	37.059788	3.535567	499.314038
<b>std</b>	0.992563	0.994216	1.011396	0.999169	79.314782
<b>min</b>	29.532429	8.508152	33.913847	0.269901	256.670582
<b>25%</b>	32.341822	11.388153	36.346012	2.933598	445.038277
<b>50%</b>	33.082008	11.983231	37.067090	3.535088	498.887875

RESULTADOS al realizar la actividad se decidio que los esfuerzos deberian enfocar mas en la aplicacion web ya que es donde hay mas cantidad de usuarios y como lo dice la tabla el maximo en web es de 40 tomando un poco mas de el doble que en movil. pero al final pensamos como punto de vista que tambien se debe de enfocar en la app movil ya que podrian abarcar mas usuarios

✓ 0 s completado a las 23:35

● ✕