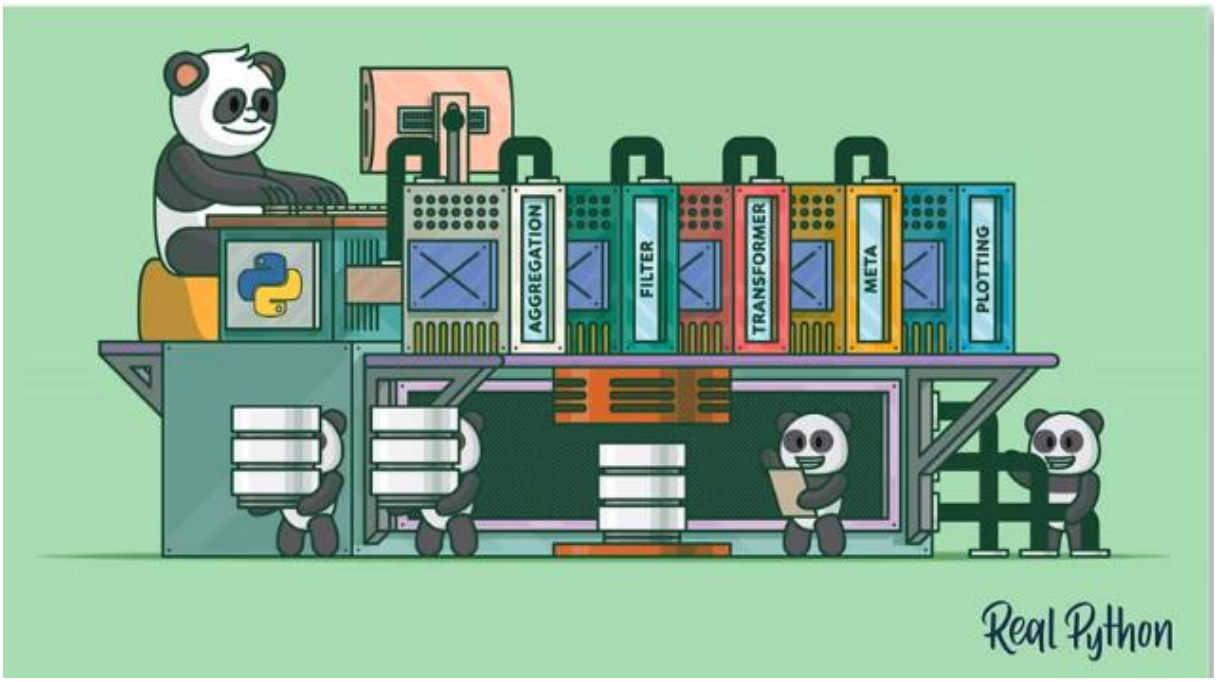
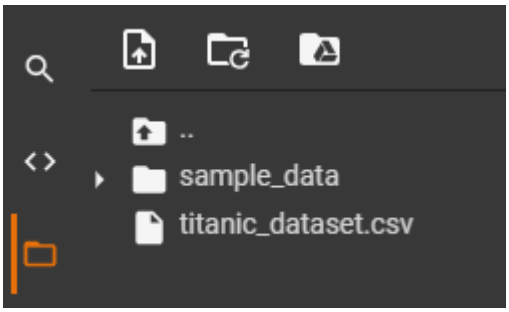
Pandas ile Veri Analizi



Çalışacağımız dosyayı yükleme:



titanic_dataset.csv dosyasına sağ tıklayıp, yolunu kopyalama: read_csv diyerek verimizi dataframe olarak çalışmamızda kullanabiliyoruz.

```
[1]  import pandas as pd
     import numpy as np

     path ='/content/titanic_dataset.csv'
     df = pd.read_csv(path)
```

Datamızın satır ve sütun sayısını öğrenmek için df.shape() komutu:

```
df.shape

(891, 12)
```

head () fonksiyonu ile varsayılan olarak bize datamızın ilk 5 verisini gösteriyor. 10 adet görmek istediğimiz de df.head(10) yazarak görüntüleyebiliyoruz.

```
df.head(10)
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	male	22.0	1	0	A/5 21171	7.2500	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	female	38.0	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss. Laina	female	26.0	0	0	STON/O2. 3101282	7.9250	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	female	35.0	1	0	113803	53.1000	C123	S
4	5	0	3	Allen, Mr. William Henry	male	35.0	0	0	373450	8.0500	NaN	S
5	6	0	3	Moran, Mr. James	male	NaN	0	0	330877	8.4583	NaN	Q
6	7	0	1	McCarthy, Mr. Timothy J	male	54.0	0	0	17463	51.8625	E46	S
7	8	0	3	Palsson, Master. Gosta Leonard	male	2.0	3	1	349909	21.0750	NaN	S
8	9	1	3	Johnson, Mrs. Oscar W (Elisabeth Vilhelmina Berg)	female	27.0	0	2	347742	11.1333	NaN	S
9	10	1	2	Nasser, Mrs. Nicholas (Adele Achem)	female	14.0	1	0	237736	30.0708	NaN	C

Sondan veri göstermek için: df.tail()

```
df.tail()
```

	PassengerId	Survived	Pclass	Name	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
886	887	0	2	Montvila, Rev. Juozas	male	27.0	0	0	211536	13.00	NaN	S
887	888	1	1	Graham, Miss. Margaret Edith	female	19.0	0	0	112053	30.00	B42	S
888	889	0	3	Johnston, Miss. Catherine Helen "Carrie"	female	NaN	1	2	W./C. 6607	23.45	NaN	S
889	890	1	1	Behr, Mr. Karl Howell	male	26.0	0	0	111369	30.00	C148	C
890	891	0	3	Dooley, Mr. Patrick	male	32.0	0	0	370376	7.75	NaN	Q

Verimizde genel bilgileri edinme: df.info()

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 891 entries, 0 to 890
Data columns (total 12 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   PassengerId  891 non-null    int64  
 1   Survived     891 non-null    int64  
 2   Pclass       891 non-null    int64  
 3   Name         891 non-null    object 
 4   Sex          891 non-null    object 
 5   Age          714 non-null    float64
 6   SibSp        891 non-null    int64  
 7   Parch        891 non-null    int64  
 8   Ticket       891 non-null    object 
 9   Fare         891 non-null    float64
 10  Cabin        204 non-null    object 
 11  Embarked     889 non-null    object 
dtypes: float64(2), int64(5), object(5)
memory usage: 83.7+ KB
```

info() satır ve sütun sayısını, her bir sütunda null, yani boş olmayan girdi sayısını, her sütunda ne türde veri olduğunu ve DataFramemimizin ne kadar bellek kullandığı gibi veri kümemizle ilgili temel ayrıntıları gösterir.

df'teki eksik değerlerimizin sayısını sıralayarak görüntüleyelim:

isnull(): Bu fonksiyon her bir kolonda boş bir değer olup olmadığını kontrol ediyor. Ardından sum() ekleyerek bunların sayısını ve ardından sort_values() ile bunları sıralayabiliyoruz.

```
df.isnull().sum().sort_values(ascending = False)
```

```
Cabin          687
Age            177
Embarked         2
Fare             0
Ticket           0
Parch            0
SibSp            0
Sex              0
Name             0
Pclass           0
Survived         0
PassengerId      0
dtype: int64
```

Embarked değişkeni ile başlayalım. Bu değişken yolcuların gemiye nerede bindiğini gösteren bir değişkendir. Kategori dağılımı için value_counts() kullanıldı.

```
df['Embarked'].value_counts()
```

```
S    644
C    168
Q     77
Name: Embarked, dtype: int64
```

Boş olan satırları silmek bazen veri kümemizde bilgi eksikliğine neden olabilir. Bu nedenle bu boşluğa başka bir değerle, genellikle sütunun ortalaması, modu veya medyanı ile doldurulur. Boş olan değerleri doldurabilmek için fillna() komutu kullanılır.

```
df['Embarked'].fillna(df['Embarked'].mode()[0], inplace = True)
```

Age kolonumuza bakalım. Age kolonumuz nümerik olduğu için mean veya median değerlerini kullanarak boş değerleri doldurabiliriz.

```
[13] df['Age'].mean()

     29.69911764705882

[14] df['Age'].median()

     28.0
```

Burada mean ve median değerleri birbirine yakın olduğu için mean kullanıyoruz. Age kolonumuzdaki değerleri ortalama değer ile doldurduk.

```
df['Age'].fillna(df['Age'].mean(), inplace = True)
```

Bazı kolonları verimizden silme:

```
[16] df.drop(['Name', 'PassengerId', 'Ticket', 'Cabin'], axis = 1, inplace=True)
```

Boş değerleri tekrar kontrol etme: Verimiz üzerinde boş değer kalmadı.

```
df.isnull().sum()

Survived    0
Pclass      0
Sex         0
Age         0
SibSp       0
Parch       0
Fare        0
Embarked    0
dtype: int64
```

KAYNAKÇA

Bilgeiş “Herkes için Yapay Zeka I” eğitimi.

KODLUYORUZ
geleceği kodluyoruz >_

EMpower
Enriching young lives in emerging markets