1. YÜKLƏMƏ : pip install pandas
2. import pandas as pd yazaraq qısaldılmış formada istifadə edə bilərik
3. Versiya sorğusu: pd.\_\_version \_\_
4. obj = pd.Series([21,’Vugar’,’Alma’,’2020’]) daha sonra bunu print(obj) desək ekrana səlqəli şəkildə yazdırar.

#Output:

0 21

1 Vugar

2 Alma

3 2020

dtype: object

Sol tərəfdə gördüyünüz rəqəmlər index`lər dir, index`lər 0-dan başlayar. Əgər print(obj[0]) yazarsaq objectdəki 1-ci dəyəri verər, yəni **‘21’**.

Ən sonda yazılan ‘dtype’ Series`a girilən dəyərin tipini göstərir. Gördüyümüz kimi bizdə də object idi tipi.

1. Əgər series tipindəki bu object`i, **array** şəklində yəni ***list*** tipində almaq istəyiriksə: print(obj.values) yazmalıyıq.

#Output:

array([21,’Vugar’,’Alma’,’2020’], dtype = object)

1. Əgər indexləri dəyişmək istəsək:

obj = pd.Series([21,’Vugar’,’Alma’,’2020’], index = [‘a’,’b’,’c’,’d’])

#Output:

a 21

b Vugar

c Alma

d 2020

dtype: object

Artıq burdan sonra dəyərləri özümüzün təyin etdiyi index`lər vasitəsi ilədə çağıra bilərik print(obj[‘a’]) .

Əgər object`in index formasını öyrənmək istəyiriksə print(obj.index) yazacayıq.

1. **Dictonary** verilənlər tipini series ilə yazdıraq. Bunu tələbələrin topladıqları ballar üzərindən edək:

bal = {“Vugar”:402, ”Anar”:355, ”Ali”:532, ”Huseyn”:555}

nt = pd.series(bal)

print(nt)

#Output:

Vugar 402

Anar 355

Ali 532

Huseyn 55

Dtype: int64

Gördüyünüz kimi burdada index`lərin yerini tələbələrin adları aldı. Artıq bizdə tələbələrin adını verərək balını ala bilərik: print(nt[‘Anar’])

Biz burda artıq bir çox riyazi hesablamalarda edə bilərik. Məsəl üçün 500 baldan çox toplayan tələbələri görmək üçün:

print([nt > 500]) yaza bilƏrik və yaxudda tələbələrin ballarını istədiyimiz ədədə vura-bölə, yəni istədiyimiz riyazi əməliyyatı edə bilərik.

nt / 100 + 5 yazaraq bütün tələbələrin balını 100`ə bölüp 5 artırırıq. Yuxarıda müəyyən etdiyimiz istənilən dəyəri sonradan da dəyişədə bilərik məsəl üçün Ali`nin balını 400 edək:

nt[‘Ali’] = 400 buradan sonra Ali`nin balı dəyişmişdir. Bu

əməliyyatı birdən çox dəyərin üzərində edə bilərik. Məsəl üçün balı 500-dən kiçik bütün tələbələrin balını 501 edək.

nt[nt < 500] = 501

1. Pandas modulunda əksik olan veri təyin etmək mümkündü, Məsəl üçün yuxardakı tələbələrin bal cədvəlinin series`i üzərində bunu yoxlayaq:

nt.isnull()

#Output:

Vugar False

Anar False

Ali False

Huseyn False

Dtype: Bool

Bütün dəyərlər **False** olaraq gəldi, bu o deməkdirki əksik veri yoxdu.

1. Bilirik ki bəzi fayıllar CSV formatında saxlanılır. CSV(Comma Separated Values) yəni **vegül ilə ayrılmış dəyərlər** fayılıdır. Bu fayllar tez-tez müxtəlif tətbiqlər arasında məlumat alış-verişi üçün istifadə olunur. Verilənlər bazaları ümumiyyətlə CSV sənədlərini dəstəkləyər.

Gəlin indi bizdə bir **.csv** uzantılı bir fayılı pandas vasitəsi ilə oxuyaq*(csv fayıllarını internetdən tapıb yükləyə bilərsiniz).* Hər hansı bir **.csv** uzantılı bir fayılı pandasda oxumaq üçün:

fayl = pd.read\_csv(“example.csv”)

*example.csv* artıq fayl adlı dəyərin içinə import edildi. Bunu görmək istəsək əgər print(fayl) yaza bilərik!

#Output:

Id ... Homepage

0 50 Cent: Bulletproof - Australia ... NaN

1 Assassin's Creed - Saudi Arabia ... http://www.assassinscreed.com

2 Battlefield III - Iran ... http://www.battlefield.com/battlefield3

3 Battlefield IV - China ... http://www.battlefield.com/battlefield-4

.. ... ... ...

131 Tom Clancy's Splinter Cell III: Chaos Theory -... ... NaN

[136 rows x 14 columns]

Eyni zamanda burda *example.csv* yerinə .**csv** uzantılı bir fayılın internet üzərindəki linkinidə qoya bilərik

fayl = pd.read\_csv(“https://query.data.world/s/r2qw2qivktjizdbpue7a4qvx6wkfqd”)

İmport etdiyimiz faylı artıq istifadə edə bilərik.

1. Əgər print(fayl.head()) yazarsaq ilk 5 sıranı göstərər.

#Output:

Id ... Homepage

0 50 Cent: Bulletproof - Australia ...

1 Assassin's Creed - Saudi Arabia ...

2 Battlefield III - Iran ...

3 Battlefield IV - China ...

4 BlazBlue II: Continuum Shift - United Arab Emi... ...

[5 rows x 14 columns]

1. Əgər print(fayl.dtypes) yazsaq **fayl** içindəki dəyişkən tiplərini göstərər.

#Output:

Id int64

Game object

Series object

Country object

Details object

Ban Category object

Ban Status object

Wikipedia Profile float64

Image float64

Summary object

Developer object

Publisher object

Genre object

Homepage object

dtype: object

1. Burdan bir dəyişkən seçək və onun təsviri dəyərlərini alaq. Bunun üçün **describe()** methodundan istifadə edəcəyik. Bu method **Series** üçün istifadə edilir, **DataFrame**(*ilərdə keçəcəyik)* üçün istifadə edilmir. Gəlin baxaq, təsviri dəyərlərini öyrənmək üçün **Genre** dəyişkənini seçək: print(fayl.Genre.describe())

#Output:

count 134

unique 43

top First-person shooter

freq 26

Name: Genre, dtype: object

134 dənə dəyər varmış, 43 dənə tək dəyər var, ən çox *First-person shooter* tipi mövcuddu və bu da 26 kəz təkrar edir.

1. Əgər dəyişkənin içərsindəki dəyərləri görmək istəyərsək: print(fayl.Genre.value\_counts())

#Output:

First-person shooter 26

Action|Adventure 16

Stealth 8

Third-person shooter 7

Fighting 7

Crime|Open world|Role-playing|Action|Adventure 6

Survival|Horror 5

Stealth|Psychological horror 5

Role-playing 4

Adventure 3

Eroge 3

Sports 2

... ...

1. Əgər hər bir dəyərin faizini öyrənmək istəyiriksə: print(fayl.Genre.value\_counts(normalize = True))

First-person shooter 0.194030

Action|Adventure 0.119403

Stealth 0.059701

... ...

1. Dedik ki dəyişkənin içərsindəki dəyərləri görmək istəyərsək: print(fayl.Genre.value\_counts()) yazırıq, indisə burdan gələn dəyərlərin hansı veri tipində olduğunu öyrənək. Bunun üçün:

type(print(fayl.Genre.value\_counts()))

#Output

<class 'pandas.core.series.Series'>

Gördüyünüz kimi Series veri tipinə aiddi! Buda o deməkdir ki biz bu dəyişkən üçün **series methodlarından** istifadə edə bilərik. Məsəl üçün yuxarda dediyimizi .head() methodu`nu burda da istifadə edək və ilk 5 dəyəri görək: print(fayl.Genre.value\_counts().head())

#Output

First-person shooter 26

Action|Adventure 16

Stealth 8

Third-person shooter 7

Fighting 7

Name: Genre, dtype: int64

1. Təkrarlanan dəyərləri tək halda görmək üçün **.unique()**: print(fayl.Genre.unique())

#Output

['Action' 'Historical fiction|Stealth|Action|Adventure'

'First-person shooter' 'Fighting' 'Sports' 'Racing|Sports'

'Action|Adventure' 'Combat|Open world' 'Combat|Racing'

'Platform|Adventure' 'Strategy' "Action|Beat 'em up"

'Hack and slash|Action|Adventure' 'Third-person shooter'

'Survival|Horror|Action|Role-playing' 'Survival|Horror' 'Role-playing'

'Adventure' 'Full Motion']

Bunlar **Genre** dəyişkəni içindəki növlərdir.

1. Neçə dənə növ olduğunu öyrənmək üçün isə **.nunique()**:

print(fayl.Genre.nunique())

1. İki dəyişkənin qarşılıqlı dəyərlərini cədvəl halında görmək istəyərsək **.crosstab()** dan istifadə edəcəyik:

info = pd.crosstab(fayl.Genre, fayl.Details)

print(info)

#Output:

Details "banned" due to existing pornography laws ... because of an 18-year old Thai player who killed a taxi-driver similar to the gameplay in GTA

Genre ...

Action 0 ... 0

Action|Adventure 0 ... 0

Action|Beat 'em up 0 ... 0

Action|Role-playing 0 ... 0

Action|Third-person shooter 0 ... 0

Adventure 0 ... 0

Beat 'em up 0 ... 0

Beat 'em up|Action|Adventure 0 ... 0

Collectible Cards 0 ... 0

Combat|Open world 0 ... 0

Combat|Racing 0 ... 0

Crime|Open world|Role-playing|Action|Adventure 0 ... 1

... ... ...

1. Əgər rəqəmsal bir dəyişkənin ortalama qiymətini almaq istərsək:

print(fayl.Global\_Sales.mean())

1. Əgər bu dəyişkənin dəyərlərinin hesablamasını görmək istəyərsək:

print(fayl.Global\_Sales.value\_counts())

#Output

0.02 1071

0.03 811

0.04 645

0.05 632

0.01 618

...

9.09 1

12.27 1

16.38 1

20.22 1

22.00 1

Name: Global\_Sales, Length: 623, dtype: int64

Bu cədvəldən belə bir nəticə çıxarırıq ki, bu satışa 0.02 milyon ilə 0.03 milyonu keçən **1071** dənə oyun vardır. Satışı 22 milyonu keçən **1** dənə oyun vardır.

1. İndi isə Series tipindəki bir veri necə görsəlləşdirilir gəlin onu öyrənək.

Grafiki sətr içində görmək üçün: %matplotlib inline fayl.Year.plot(kind=”hist”)