DSBDA Assignment 3

Descriptive Statistics – Measures of Central Tendency

Peform the following operations on any open source dataset.

1. Provide summary statistic (mean, median, minimum, maximum, standard deviation) for a dataset with numberic variables grouped by one of the qualitative variable, For example, if your categorical variable is age groups and quantitative variable is income, then provide summary statistics of income grouped by the age groups. Create a list that contains a numberic value for each response to the categorical variable.
2. Write a python program to display some basic statistical details like percentile, mean, standard deviation etc of the species of “Iris-setosa”, “Iris-veriscolor” of iris-dataset.

```
import pandas, numpy
```

```
data_frame = pandas.read_csv('DataSets/netfilx_movies.csv')
```

```
data_frame
```

	show_id	type	title	director	cast	country	date_added	release_year	rating	duration	listed_in
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	PG-13	90 min	Documentaries
1	s2	TV Show	Blood & Water	NaN	Ama Qamata, Khosi Ngema, Gail Mabalane, Thaban...	South Africa	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, TV Dramas, TV Mysteries
2	s3	TV Show	Ganglands	Julien Leclercq	Sami Bouajila, Tracy Gotoas, Samuel Jouy, Nabi...	NaN	September 24, 2021	2021	TV-MA	1 Season	Crime TV Shows, International TV Shows, TV Act...
3	s4	TV Show	Jailbirds New Orleans	NaN	NaN	NaN	September 24, 2021	2021	TV-MA	1 Season	Docuseries, Reality TV
4	s5	TV Show	Kota Factory	NaN	Mayur More, Jitendra Kumar, Ranjan Raj, Alam K...	India	September 24, 2021	2021	TV-MA	2 Seasons	International TV Shows, Romantic TV Shows, TV ...

Dataset:- netflix_movies Source: kaggle

Example of group by:-

```
data_frame['type'].value_counts()
```

```
Movie      6131
TV Show    2676
Name: type, dtype: int64
```

```
movie_type = data_frame.groupby('type')
```

```
movie_type.get_group('Movie')
```

```
movie_type.get_group('Movie')
```

	show_id	type	title	director	cast	country	date_added	release_year	r
0	s1	Movie	Dick Johnson Is Dead	Kirsten Johnson	NaN	United States	September 25, 2021	2020	
6	s7	Movie	My Little Pony: A New Generation	Robert Cullen, José Luis Ucha	Vanessa Hudgens, Kimiko Glenn, James Marsden, ...	NaN	September 24, 2021	2021	
7	s8	Movie	Sankofa	Haile Gerima	Kofi Ghanaba, Oyafunmike Ogunlano, Alexandra D...	United States, Ghana, Burkina Faso, United Kin...	September 24, 2021	1993	
9	s10	Movie	The Starling	Theodore Melfi	Melissa McCarthy, Chris O'Dowd, Kevin Kline, T...	United States	September 24, 2021	2021	

Performing Operations on grouped by dataset

```
smartphones = pandas.read_csv('DataSets/flipkart_smartphones.csv')
```

```
smartphones
```

	brand	model	colour	original_price	discounted_price	ratings	rating_count	reviews	memory	storage	processor	rear_camera
0	VIVO	VIVO T1 44W	Starry Sky	19990	14499	4.5	87331	6044	4.0	128.0	Qualcomm Snapdragon 680	50MP + 2MP + 2MP
1	APPLE	APPLE IPHONE 11	White	48900	47199	4.6	184191	10818	NaN	128.0	A Bionic Chip	12MP + 12MP
2	VIVO	VIVO T1 44W	Midnight Galaxy	20990	15999	4.4	51365	3750	6.0	128.0	Qualcomm Snapdragon 680	50MP + 2MP + 2MP
3	XIAOMI	POCO M4 5G	Power Black	15999	11999	4.2	53448	4185	4.0	64.0	Mediatek Dimensity 700	50MP + 2MP
4	XIAOMI	REDMI 10	Caribbean Green	14999	9299	4.3	187787	12084	4.0	64.0	Qualcomm Snapdragon 680	50MP + 2MP

```
smartphones_df = smartphones.groupby('brand')
```

```
xiaomi = smartphones_df.get_group('XIAOMI')
```

```
xiaomi
```

	brand	model	colour	original_price	discounted_price	ratings	rating_count	reviews	memory	storage	processor	rear_camera
3	XIAOMI	POCO M4 5G	Power Black	15999	11999	4.2	53448	4185	4.0	64.0	Mediatek Dimensity 700	50MP + 2MP
4	XIAOMI	REDMI 10	Caribbean Green	14999	9299	4.3	187787	12084	4.0	64.0	Qualcomm Snapdragon 680	50MP + 2MP
5	XIAOMI	POCO M4 5G	Cool Blue	15999	11999	4.2	53448	4185	4.0	64.0	Mediatek Dimensity 700	50MP + 2MP
6	XIAOMI	POCO C31	Shadow Gray	11999	7499	4.3	202727	11672	4.0	64.0	MediaTek Helio G35	13MP + 2MP + 2MP

```
xiaomi['original_price'].mean()
```

```
20870.837988826817
```

```
xiaomi['original_price'].describe()
```

```
count      179.000000
mean     20870.837989
std      10802.224432
min       8499.000000
25%      15999.000000
50%      17999.000000
75%      23999.000000
max      84999.000000
Name: original_price, dtype: float64
```

```
xiaomi[['original_price', 'discounted_price']].describe()
```

	original_price	discounted_price
count	179.000000	179.000000
mean	20870.837989	16345.938547
std	10802.224432	8326.211551
min	8499.000000	5999.000000
25%	15999.000000	10624.000000
50%	17999.000000	14699.000000
75%	23999.000000	20444.500000
max	84999.000000	51999.000000

Operations on Iris – Data set

```
data_frame = pandas.read_csv('DataSets/iris.csv')
```

```
data_frame
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
...	...	...	...	...	...
145	6.7	3.0	5.2	2.3	Iris-virginica
146	6.3	2.5	5.0	1.9	Iris-virginica
147	6.5	3.0	5.2	2.0	Iris-virginica
148	6.2	3.4	5.4	2.3	Iris-virginica
149	5.9	3.0	5.1	1.8	Iris-virginica

150 rows × 5 columns

```
species = data_frame.groupby('species')
```

```
Iris_Setosa = species.get_group('Iris-setosa')
Iris_Setosa
```

	sepal_length	sepal_width	petal_length	petal_width	species
0	5.1	3.5	1.4	0.2	Iris-setosa
1	4.9	3.0	1.4	0.2	Iris-setosa
2	4.7	3.2	1.3	0.2	Iris-setosa
3	4.6	3.1	1.5	0.2	Iris-setosa
4	5.0	3.6	1.4	0.2	Iris-setosa
5	5.4	3.9	1.7	0.4	Iris-setosa
6	4.6	3.4	1.4	0.3	Iris-setosa
7	5.0	3.4	1.5	0.2	Iris-setosa
8	4.4	2.9	1.4	0.2	Iris-setosa
9	4.9	3.1	1.5	0.1	Iris-setosa
10	5.4	3.7	1.5	0.2	Iris-setosa

```
Iris_Setosa[['petal_length', 'petal_width', 'sepal_length', 'sepal_width']].describe()
```

	petal_length	petal_width	sepal_length	sepal_width
count	50.000000	50.00000	50.00000	50.000000
mean	1.464000	0.24400	5.00600	3.418000
std	0.173511	0.10721	0.35249	0.381024
min	1.000000	0.10000	4.30000	2.300000
25%	1.400000	0.20000	4.80000	3.125000
50%	1.500000	0.20000	5.00000	3.400000
75%	1.575000	0.30000	5.20000	3.675000
max	1.900000	0.60000	5.80000	4.400000

```
Iris_Setosa['petal_length'].quantile()
```

```
1.5
```

```
species.describe()
```

	sepal_length								sepal_width							
	count	mean	std	min	25%	50%	75%	max	count	mean	std	min	25%	50%	75%	r
species																
Iris-setosa	50.0	5.006	0.352490	4.3	4.800	5.0	5.2	5.8	50.0	3.418	0.381024	2.3	3.125	3.4	3.675	
Iris-versicolor	50.0	5.936	0.516171	4.9	5.600	5.9	6.3	7.0	50.0	2.770	0.313798	2.0	2.525	2.8	3.000	
Iris-virginica	50.0	6.588	0.635880	4.9	6.225	6.5	6.9	7.9	50.0	2.974	0.322497	2.2	2.800	3.0	3.175	

```
species.quantile()
```

	sepal_length	sepal_width	petal_length	petal_width
species				
Iris-setosa	5.0	3.4	1.50	0.2
Iris-versicolor	5.9	2.8	4.35	1.3
Iris-virginica	6.5	3.0	5.55	2.0