In [ ]:
```
import pandas as pd
import numpy as np
```

In [118…
```
df = pd.read_excel("C:\\Users\\user\\3D Objects\\EDA_Shark_Tank_India.xlsx")
df.head(10)
```

Out[118…

	episode_number	pitch_number	brand_name	idea	deal	pitcher_ask_amoun
0	1	1	BluePine Industries	Frozen Momos	1	50.
1	1	2	Booz scooters	Renting e-bike for mobility in private spaces	1	40.
2	1	3	Heart up my Sleeves	Detachable Sleeves	1	25.
3	2	4	Tagz Foods	Healthy Potato Chips	1	70.
4	2	5	Head and Heart	Brain Development Course	0	50.
5	2	6	Agro tourism	Tourism	0	50.
6	3	7	Qzense Labs	Food Freshness Detector	0	100.
7	3	8	Peeschute	Disposable Urine Bag	1	75.
8	3	9	NOCD	Energy Drink	1	50.
9	4	10	Cosiq	Intelligent Skincare	1	50.

10 rows × 28 columns

In [119…
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 117 entries, 0 to 116
Data columns (total 28 columns):
 #   Column                 Non-Null Count  Dtype  
---  ------                 --------------  -----  
 0   episode_number         117 non-null    int64  
 1   pitch_number           117 non-null    int64  
 2   brand_name             117 non-null    object 
 3   idea                   117 non-null    object 
 4   deal                   117 non-null    int64  
 5   pitcher_ask_amount     117 non-null    float64
 6   ask_equity             117 non-null    float64
 7   ask_valuation          117 non-null    float64
 8   deal_amount            117 non-null    float64
 9   deal_equity            117 non-null    float64
 10  deal_valuation         117 non-null    float64
 11  ashneer_present        117 non-null    int64  
 12  anupam_present         117 non-null    int64  
 13  aman_present           117 non-null    int64  
 14  namita_present         117 non-null    int64  
 15  vineeta_present        117 non-null    int64  
 16  peyush_present         117 non-null    int64  
 17  ghazal_present         117 non-null    int64  
 18  ashneer_deal           117 non-null    int64  
 19  anupam_deal            117 non-null    int64  
 20  aman_deal              117 non-null    int64  
 21  namita_deal            117 non-null    int64  
 22  vineeta_deal           117 non-null    int64  
 23  peyush_deal            117 non-null    int64  
 24  ghazal_deal            117 non-null    int64  
 25  total_sharks_invested  117 non-null    int64  
 26  amount_per_shark       117 non-null    float64
 27  equity_per_shark       117 non-null    float64
dtypes: float64(8), int64(18), object(2)
memory usage: 25.7+ KB
```

In [184…

```
df.describe().transpose()
```

Out[184…

	count	mean	std	min	25%	50%	7
episode_number	117.0	18.735043	10.070778	1.00000	10.00	19.00	27
pitch_number	117.0	59.000000	33.919021	1.00000	30.00	59.00	88
deal	117.0	0.555556	0.499041	0.00000	0.00	1.00	1
ask_amount	117.0	319.854709	2767.842777	0.00101	45.00	50.00	80
ask_equity	117.0	5.188034	3.892121	0.25000	2.50	5.00	7
ask_valuation	117.0	3852.462479	11931.601957	0.01000	666.67	1250.00	2857
deal_amount	117.0	31.982915	36.687391	0.00000	0.00	25.00	50
deal_equity	117.0	8.963504	13.106769	0.00000	0.00	3.00	15
deal_valuation	117.0	467.104872	919.988864	0.00000	0.00	100.00	500
ashneer_present	117.0	0.837607	0.370397	0.00000	1.00	1.00	1
anupam_present	117.0	1.000000	0.000000	1.00000	1.00	1.00	1
aman_present	117.0	0.837607	0.370397	0.00000	1.00	1.00	1
namita_present	117.0	0.905983	0.293108	0.00000	1.00	1.00	1
vineeta_present	117.0	0.564103	0.498007	0.00000	0.00	1.00	1
peyush_present	117.0	0.752137	0.433629	0.00000	1.00	1.00	1
ghazal_present	117.0	0.222222	0.417528	0.00000	0.00	0.00	0
ashneer_deal	117.0	0.179487	0.385410	0.00000	0.00	0.00	0
anupam_deal	117.0	0.205128	0.405532	0.00000	0.00	0.00	0
aman_deal	117.0	0.239316	0.428501	0.00000	0.00	0.00	0
namita_deal	117.0	0.188034	0.392420	0.00000	0.00	0.00	0
vineeta_deal	117.0	0.128205	0.335756	0.00000	0.00	0.00	0
peyush_deal	117.0	0.230769	0.423137	0.00000	0.00	0.00	0
ghazal_deal	117.0	0.059829	0.238190	0.00000	0.00	0.00	0
total_sharks_invested	117.0	1.230769	1.410457	0.00000	0.00	1.00	2
amount_per_shark	117.0	18.132481	23.588682	0.00000	0.00	10.00	25
equity_per_shark	117.0	5.583590	10.803799	0.00000	0.00	1.25	6

In [134…

```
df.columns
```

Out[134…
```
Index(['episode_number', 'pitch_number', 'brand_name', 'idea', 'deal',
       'pitcher_ask_amount', 'ask_equity', 'ask_valuation', 'deal_amount',
       'deal_equity', 'deal_valuation', 'ashneer_present', 'anupam_present',
       'aman_present', 'namita_present', 'vineeta_present', 'peyush_present',
       'ghazal_present', 'ashneer_deal', 'anupam_deal', 'aman_deal',
       'namita_deal', 'vineeta_deal', 'peyush_deal', 'ghazal_deal',
       'total_sharks_invested', 'amount_per_shark', 'equity_per_shark'],
      dtype='object')
```

In [ ]:
```
```

In [121…
```
ndf = df[['episode_number','brand_name','pitcher_ask_amount', 'ask_equity', 'as
ndf.head(10)
```

Out[121…

	episode_number	brand_name	pitcher_ask_amount	ask_equity	ask_valuation
0	1	BluePine Industries	50.0	5.00	1000.00
1	1	Booz scooters	40.0	15.00	266.67
2	1	Heart up my Sleeves	25.0	10.00	250.00
3	2	Tagz Foods	70.0	1.00	7000.00
4	2	Head and Heart	50.0	5.00	1000.00
5	2	Agro tourism	50.0	5.00	1000.00
6	3	Qzense Labs	100.0	0.25	40000.00
7	3	Peeschute	75.0	4.00	1875.00
8	3	NOCD	50.0	2.00	2500.00
9	4	Cosiq	50.0	7.50	666.67

In [122…
```
df.shape
```

Out[122…
```
(117, 28)
```

Q1. Find the number of episodes?

In [138…
```
print(ndf['episode_number'].unique())
```

```
[ 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
 25 26 27 28 29 30 31 32 33 34 35]
```

In [136…
```
print(ndf['episode_number'].nunique())
```

```
35
```

In [124…
```
ndf.rename(columns={'pitcher_ask_amount': 'ask_amount'},inplace="True")
ndf.head(10)
```

Out[124...

	episode_number	brand_name	ask_amount	ask_equity	ask_valuation
0	1	BluePine Industries	50.0	5.00	1000.00
1	1	Booz scooters	40.0	15.00	266.67
2	1	Heart up my Sleeves	25.0	10.00	250.00
3	2	Tagz Foods	70.0	1.00	7000.00
4	2	Head and Heart	50.0	5.00	1000.00
5	2	Agro tourism	50.0	5.00	1000.00
6	3	Qzense Labs	100.0	0.25	40000.00
7	3	Peeschute	75.0	4.00	1875.00
8	3	NOCD	50.0	2.00	2500.00
9	4	Cosiq	50.0	7.50	666.67

Q2. Max, min , mean of asked amount,asked equity, asked valuation?

In [128...

```
ndf[['ask_amount', 'ask_equity', 'ask_valuation']].agg(['max', 'min', 'mean'])
```

Out[128...

	ask_amount	ask_equity	ask_valuation
max	30000.000000	25.000000	120000.000000
min	0.001010	0.250000	0.010000
mean	319.854709	5.188034	3852.462479

Q3. Max & Min asked- equity,asked-valuation and asked amount episode-wise?

In [129...

```
gr = ndf.groupby('episode_number')[['ask_amount', 'ask_equity', 'ask_valuation'
gr
```

Out[129...

	ask_amount		ask_equity		ask_valuation	
	max	min	max	min	max	min
episode_number						
1	50.0	25.00000	15.0	5.00	1000.00	250.00
2	70.0	50.00000	5.0	1.00	7000.00	1000.00
3	100.0	50.00000	4.0	0.25	40000.00	1875.00
4	75.0	50.00000	10.0	4.00	1875.00	500.00
5	100.0	10.00000	20.0	1.00	10000.00	50.00
6	100.0	45.00000	10.0	1.00	10000.00	500.00
7	100.0	50.00000	7.5	1.00	7500.00	666.67
8	56.0	30.00000	7.5	2.50	2000.00	746.67
9	100.0	50.00000	5.0	2.50	2000.00	1000.00
10	30.0	25.00000	5.0	2.00	1500.00	500.00
11	30000.0	30.00000	25.0	5.00	120000.00	300.00
12	75.0	40.00000	5.0	3.00	1875.00	800.00
13	50.0	30.00000	10.0	2.00	2500.00	500.00
14	100.0	45.00000	5.0	3.00	3333.33	900.00
15	50.0	5.00000	5.0	3.00	1250.00	100.00
16	80.0	45.00000	7.0	2.00	2250.00	1071.43
17	150.0	50.00000	10.0	3.00	5000.00	500.00
18	100.0	50.00000	4.0	1.00	10000.00	1250.00
19	125.0	15.00000	8.0	1.25	10000.00	300.00
20	65.0	35.00000	5.0	2.00	3250.00	700.00
21	100.0	35.00000	10.0	5.00	1250.00	470.00
22	80.0	50.00000	5.0	2.00	3000.00	1500.00
23	100.0	30.00000	5.0	1.00	10000.00	600.00
24	40.0	20.00000	10.0	8.00	500.00	200.00
25	150.0	50.00000	4.0	2.00	7500.00	1250.00
26	65.0	50.00000	10.0	1.00	6500.00	500.00
27	100.0	0.00101	10.0	1.00	10000.00	0.01
28	90.0	50.00000	5.0	4.00	1875.00	1000.00
29	100.0	75.00000	5.0	3.00	2857.14	1500.00
30	300.0	50.00000	15.0	1.00	30000.00	500.00

	ask_amount		ask_equity		ask_valuation	
	max	min	max	min	max	min
episode_number						
31	75.0	50.00000	10.0	2.00	3750.00	500.00
32	200.0	35.00000	7.0	1.00	5000.00	583.33
33	40.0	35.00000	10.0	1.00	3500.00	400.00
34	100.0	30.00000	10.0	1.00	10000.00	400.00
35	100.0	40.00000	8.0	2.50	4000.00	500.00

Q4. Brand names in which 2 ,3 or 4 sharks are invested?

In [144…
```
brands_df = df[(df['total_sharks_invested'].isin([2, 3, 4]))]['brand_name']
brands_df.unique()
```

Out[144…
```
array(['BluePine Industries', 'Booz scooters', 'Heart up my Sleeves',
       'Cosiq', 'Bummer', 'Revamp Moto', 'Raising Superstars',
       'Beyond Snack', 'Vivalyf Innovations- Easy Life', 'Altor', 'Ariro',
       'Nuutjob', 'Meatyour', 'EventBeep', 'Farda', 'LOKA', 'Annie',
       'Caragreen', 'The Yarn Bazaar', 'The Renal Project', 'Cocofit',
       'Bamboo India', 'Beyond Water', "Let's Try", 'Aas Vidyalaya',
       'WeSTOCK', 'Get a Whey', 'The Quirky Nari', 'Hair Originals',
       'The Sass Bar', 'Watt Technovations', 'Humpy A2',
       'Gold Safe Solutions Ind.', 'Wakao Foods', 'Kabaddi Adda',
       'Tweek Labs', 'Proxgy', 'Nomad Food Project', 'Jain Shikanji'],
      dtype=object)
```

Q5: Episode-wise minimum and maximum number of sharks invested.

In [145…
```
episode = df.groupby('episode_number')['total_sharks_invested'].agg(['min', 'ma

episode
```

Out[145...

episode_number	min	max
1	2	3
2	0	1
3	0	1
4	0	2
5	0	2
6	0	5
7	0	2
8	1	2
9	0	2
10	3	3
11	0	2
12	0	3
13	2	4
14	0	2
15	1	3
16	0	2
17	0	5
18	0	1
19	0	4
20	0	5
21	1	3
22	0	3
23	0	1
24	0	2
25	0	5
26	0	1
27	0	4
28	0	3
29	0	3
30	0	2
31	0	1

	min	max
episode_number		
32	0	1
33	0	3
34	0	4
35	0	4

Q6: Brands that asked for 1 crore and got a deal.

In [152…
```
df.rename(columns={'pitcher_ask_amount': 'ask_amount'},inplace="True")
df.head(10)
```

Out[152…

	episode_number	pitch_number	brand_name	idea	deal	ask_amount	ask_
0	1	1	BluePine Industries	Frozen Momos	1	50.0	
1	1	2	Booz scooters	Renting e-bike for mobility in private spaces	1	40.0	
2	1	3	Heart up my Sleeves	Detachable Sleeves	1	25.0	
3	2	4	Tagz Foods	Healthy Potato Chips	1	70.0	
4	2	5	Head and Heart	Brain Development Course	0	50.0	
5	2	6	Agro tourism	Tourism	0	50.0	
6	3	7	Qzense Labs	Food Freshness Detector	0	100.0	
7	3	8	Peeschute	Disposable Urine Bag	1	75.0	
8	3	9	NOCD	Energy Drink	1	50.0	
9	4	10	Cosiq	Intelligent Skincare	1	50.0	

10 rows × 28 columns

In [153…
```
brands_deal = df[(df['ask_amount'] == 100) & (df['deal'] == 1)]['brand_name']

brands_deal
```

Out[153…
```
12            Revamp Moto
39      The Renal Project
55                   Otua
64             Get a Whey
71           Namhya Foods
79    Sunfox Technologies
87     Insurance Samadhan
Name: brand_name, dtype: object
```

Q7. Brand names where deal equity is between 0 to 50?

In [160…
```
deal = df[(df['deal_equity'] >= 0) & (df['deal_equity'] <= 50)]['brand_name']
print(deal.nunique())
print(deal.unique())
```

```
116
['BluePine Industries' 'Booz scooters' 'Heart up my Sleeves' 'Tagz Foods'
 'Head and Heart' 'Agro tourism' 'Qzense Labs' 'Peeschute' 'NOCD' 'Cosiq'
 'JhaJi Achaar' 'Bummer' 'Revamp Moto' 'Hungry Heads' 'Shrawani Engineers'
 'Skippi Pops' 'Menstrupedia' 'Hecolll' 'Raising Superstars' 'Torch-it'
 'La Kheer Deli' 'Beyond Snack' 'Vivalyf Innovations- Easy Life'
 'Motion Breeze' 'Altor' 'Ariro' 'Kabira Handmade' 'Nuutjob' 'Meatyour'
 'EventBeep' "Gopal's 56" 'ARRCOAT Surface Textures' 'Farda'
 'Auli Lifestyle' 'SweeDesi' 'LOKA' 'Annie' 'Caragreen' 'The Yarn Bazaar'
 'The Renal Project' 'Morikko Pure Foods' 'Good Good Piggy Bank'
 'Hammer Lifestyle' 'PNT' 'Cocofit' 'Bamboo India' 'Flying Furr'
 'Beyond Water' "Let's Try" 'Find Your Kicks India' 'Aas Vidyalaya'
 'Outbox' 'RoadBounce' "Mommy's Kitchen" 'India Hemp and Co' 'Otua'
 'Anthyesti' 'Ethik' 'WeSTOCK' 'KetoIndia' 'Magic lock' 'The State Plate'
 'Bakarmax' 'IN A CAN' 'Get a Whey' 'The Quirky Nari' 'Hair Originals'
 'Poo de Cologne' 'Moonshine Meads' 'Falhari' 'Namhya Foods'
 'Urban Monkey' 'Guardian Gears' 'Modern Myth' 'The Sass Bar'
 'KG Agrotech' 'Nuskha Kitchen' 'PawsIndia' 'Sunfox Technologies' 'Alpino'
 'Isak Fragrances' 'Julaa Automation' 'Rare Planet' 'Theka Coffee'
 'Watt Technovations' 'Aliste Technologies' 'Insurance Samadhan'
 'Humpy A2' 'Kunafa World' 'Gold Safe Solutions Ind.' 'Wakao Foods'
 'PDD Falcon' 'PlayBox TV' 'Sippline Drinking Shields' 'Kabaddi Adda'
 'Shades of Spring' 'Scholify' 'Scrapshala' 'Sabjikothi' 'AyuRythm'
 'Astrix' 'Thea and Sid' 'Experential Etc' 'GrowFitter' 'Med Tech'
 'Colour Me Mad' "Mavi's" 'Tweek Labs' 'Proxgy' 'Nomad Food Project'
 'Twee in One' 'Green Protein' 'On2Cook' 'Jain Shikanji' 'Woloo'
 'Elcare India']
```

Q8. Find the number of brands participated in each episode?

In [177…
```
b = df.groupby('episode_number')['brand_name'].count()
print(b.unique())
```

```
[3 4]
```

Q9. How many sharks participated in this show and What were their names?

In [182…
```
shark_columns = [col for col in df.columns if '_present' in col]
shark_names = [col.replace('_present', '') for col in shark_columns]
participated = len(shark_names)
```

```
print("Number of sharks participated:", participated)
print("Names of sharks:", shark_names)
```

```
Number of sharks participated: 7
Names of sharks: ['ashneer', 'anupam', 'aman', 'namita', 'vineeta', 'peyush', 'g
hazal']
```

OR

In [ ]:
```
shark_columns = [col for col in df.columns if '_deal' in col]
shark_names = [col.replace('_deal', '') for col in shark_columns]
participated = len(shark_names)
print("Number of sharks participated:",participated)
print("Names of sharks:", shark_names)
```

```
Number of sharks participated: 7
Names of sharks: ['ashneer', 'anupam', 'aman', 'namita', 'vineeta', 'peyush', 'g
hazal']
```

Q10. Find appearance of each sharks?

In [198…
```
appear = {shark: df[shark + '_present'].sum() for shark in shark_names}
print("Appearance of each shark:",appear)
```

```
Appearance of each shark: {'ashneer': 98, 'anupam': 117, 'aman': 98, 'namita': 1
06, 'vineeta': 66, 'peyush': 88, 'ghazal': 26}
```

Q11. How many entrepreneurs were present?

In [193…
```
n = df['pitch_number'].nunique()
print("Total number of entrepreneurs:", n)
```

```
Total number of entrepreneurs: 117
```

Q12. How many times each shark invested the deal?

In [197…
```
invest = {shark: df[shark + '_deal'].sum() for shark in shark_names}
print("Investment count for each shark:",invest)
```

```
Investment count for each shark: {'ashneer': 21, 'anupam': 24, 'aman': 28, 'nami
ta': 22, 'vineeta': 15, 'peyush': 27, 'ghazal': 7}
```

Q13. Find the equity percent that each sharks gets?

In [207…
```
p = {
    shark: (df[df[shark + '_deal'] == 1]['equity_per_shark']).sum()
    for shark in shark_names
}
p
```

Out[207…
```
{'ashneer': 93.249999999,
 'anupam': 166.35,
 'aman': 160.263333334,
 'namita': 134.78333333400002,
 'vineeta': 131.533333333,
 'peyush': 315.84999999999997,
 'ghazal': 46.7}
```

Q14. Find the total number of amount invested in this show?

In [211…
```
t = df['deal_amount'].sum()
print("Total amount invested in the show:", t.round())
```

```
Total amount invested in the show: 3742.0
```