In [1]:

```python
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
```

Dipak Mani

In [2]:

```python
df = pd.read_csv('Shark Tank India Dataset.csv')
df
```

Out[2]:

| | episode_number | pitch_number | brand_name | idea | deal | pitcher_ask_amount | a |
|---|---|---|---|---|---|---|---|
| **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 | |
| **...** | ... | ... | ... | ... | ... | ... | |
| **112** | 34 | 113 | Green Protein | Plant-Based Protein | 0 | 60.0 | |
| **113** | 34 | 114 | On2Cook | Fastest Cooking Device | 0 | 100.0 | |
| **114** | 35 | 115 | Jain Shikanji | Lemonade | 1 | 40.0 | |
| **115** | 35 | 116 | Woloo | Washroom Finder | 0 | 50.0 | |
| **116** | 35 | 117 | Elcare India | Carenting for Elders | 0 | 100.0 | |

117 rows × 28 columns

◀ ▬▬▬▬▬▬▬                                                                                        ▶
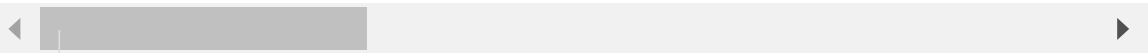
```
1    About Dataset
2  Context:
```

3   Shark Tank India is an Indian Hindi-language business reality television series
     that airs on Sony Entertainment Television. The show is the Indian franchise of
     the American show Shark Tank. It shows entrepreneurs making business presentations
     to a panel of investors or sharks, who decide whether to invest in their company.
     The first season of Shark Tank India premiered on 20 December 2021, and concluded
     on 4 February 2022.

4

5   Sharks Intro
6   1- Ashneer Grover

7

8   Managing Director and Co-founder of BharatPe
9   2- Aman Gupta

10

11   Co-founder and Chief Marketing Officer of boAt
12   3- Anupam Mittal

13

14   Founder and CEO of Shaadi.com and People Group
15   4- Ghazal Alagh

16

17   Co-founder and Chief Mama of MamaEarth
18   5- Namita Thapar

19

20   Executive Director of Emcure Pharmaceuticals
21   6- Peyush Bansal

22

23   Co-founder and CEO of Lenskart
24   7- Vineeta Singh

25

26   CEO and co-founder of SUGAR Cosmetics
27   Data Description
28   Episode_number - Number of the episode
29   Pitch_number - Number of the Pitch
30   Brand_name - Name of the brand Idea
31   Idea - behind the brand building Deal
32   Deal - done or not ; 1 - YES, 0 - NO
33   Pitcher_ask_amount - Amount asked by the pitchers
34   Ask_equity - Equity offered by the pitchers
35   Ask_valuation - Valuation asked by pitchers
36   Deal_amount - Final Deal Amount
37   Deal_equity - Final Deal equity percentage
38   Deal_valuation - Final Valuation of Company after Deal
39   Ashneer_present - Ashneer was present during the pitching ; 1 - YES, 0 - NO
40   Anupam_present - Anupam was present during the pitching ; 1 - YES, 0 - NO
41   Aman_present - Aman was present during the pitching ; 1 - YES, 0 - NO
42   Namita_present - Namita was present during the pitching ; 1 - YES, 0 - NO
43   Vineeta_present - Vineeta was present during the pitching ; 1 - YES, 0 - NO
44   Peyush_present - Peyush was present during the pitching ; 1 - YES, 0 - NO
45   Ghazal_present - Ghazal was present during the pitching ; 1 - YES, 0 - NO
46   Ashneer_deal - Ashneer is a part of Final Deal ; 1 - YES, 0 - NO
47   Anupam_deal - Anupam is a part of Final Deal ; 1 - YES, 0 - NO
48   Aman_deal - Aman is a part of Final Deal ; 1 - YES, 0 - NO
49   Namita_deal - Namita is a part of Final Deal ; 1 - YES, 0 - NO
50   Vineeta_deal - Vineeta is a part of Final Deal ; 1 - YES, 0 - NO
51   Peyush_deal - Peyush is a part of Final Deal ; 1 - YES, 0 - NO
52   Ghazal_deal - Ghazal is a part of Final Deal ; 1 - YES, 0 - NO
53   Total_sharks_invested - Number of total sharks invested in the Company
54   Amount_per_shark - Amount per shark invested
55   Equity_per_shark - Final Equity gained per Shark
56   Acknowledgement:
57   Shark Tank India | Wikipedia

Workflow

1: UNDERSTANDING OF DATA

2: CLEANING OF DATA

3: INSIGHTS

1. Understanding the Data

In [3]:

```python
# Number of rows and columns of data
df.shape
```

Out[3]:

```
(117, 28)
```

In [4]:

```python
# Fetching columns name
df.columns
```

Out[4]:

```
Index(['episode_number', 'pitch_number', 'brand_name', 'idea', 'deal',
       'pitcher_ask_amount', 'ask_equity', 'ask_valuation', 'deal_amoun
t',
       'deal_equity', 'deal_valuation', 'ashneer_present', 'anupam_presen
t',
       'aman_present', 'namita_present', 'vineeta_present', 'peyush_prese
nt',
       '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 [5]:

```python
# We checking column data type
df.dtypes
```

Out[5]:

```
episode_number           int64
pitch_number             int64
brand_name              object
idea                    object
deal                     int64
pitcher_ask_amount     float64
ask_equity             float64
ask_valuation          float64
deal_amount            float64
deal_equity            float64
deal_valuation         float64
ashneer_present          int64
anupam_present           int64
aman_present             int64
namita_present           int64
vineeta_present          int64
peyush_present           int64
ghazal_present           int64
ashneer_deal             int64
anupam_deal              int64
aman_deal                int64
namita_deal              int64
vineeta_deal             int64
peyush_deal              int64
ghazal_deal              int64
total_sharks_invested    int64
amount_per_shark       float64
equity_per_shark       float64
dtype: object
```

In [6]:

```
1 df.select_dtypes(include=['object'])
```

Out[6]:

	brand_name	idea
0	BluePine Industries	Frozen Momos
1	Booz scooters	Renting e-bike for mobility in private spaces
2	Heart up my Sleeves	Detachable Sleeves
3	Tagz Foods	Healthy Potato Chips
4	Head and Heart	Brain Development Course
...	...	...
112	Green Protein	Plant-Based Protein
113	On2Cook	Fastest Cooking Device
114	Jain Shikanji	Lemonade
115	Woloo	Washroom Finder
116	Elcare India	Carenting for Elders

117 rows × 2 columns

There are two columns are categorical

In [7]:

```
1 df.select_dtypes(include=['int', 'float'])
```

Out[7]:

	episode_number	pitch_number	deal	pitcher_ask_amount	ask_equity	ask_valuation	d
0	1	1	1	50.0	5.0	1000.00	
1	1	2	1	40.0	15.0	266.67	
2	1	3	1	25.0	10.0	250.00	
3	2	4	1	70.0	1.0	7000.00	
4	2	5	0	50.0	5.0	1000.00	
...	...	...	...	...	...	...	
112	34	113	0	60.0	2.0	3000.00	
113	34	114	0	100.0	1.0	10000.00	
114	35	115	1	40.0	8.0	500.00	
115	35	116	0	50.0	4.0	1250.00	
116	35	117	0	100.0	2.5	4000.00	

117 rows × 26 columns
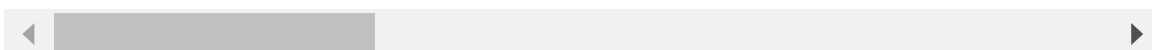
In [8]:

```
1  # Viewing top 5 rows
2  df.head()
```

Out[8]:

	episode_number	pitch_number	brand_name	idea	deal	pitcher_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 rows × 28 columns

In [9]:

```
1  #Viewing data of first 10 columns
2  df.iloc[0:5,0:10]
```

Out[9]:

	episode_number	pitch_number	brand_name	idea	deal	pitcher_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	

Here it shows the types of columns ,all columns are numerical except brand_name and idea are categorical
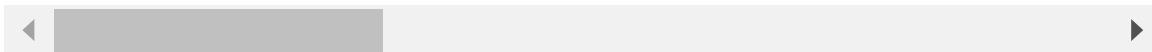
In [10]:

```
1  # Viewing bottom 5 rows
2
3  df.tail()
```

Out[10]:

	episode_number	pitch_number	brand_name	idea	deal	pitcher_ask_amount	ask
112	34	113	Green Protein	Plant-Based Protein	0	60.0	
113	34	114	On2Cook	Fastest Cooking Device	0	100.0	
114	35	115	Jain Shikanji	Lemonade	1	40.0	
115	35	116	Woloo	Washroom Finder	0	50.0	
116	35	117	Elcare India	Carenting for Elders	0	100.0	

5 rows × 28 columns

In [11]:

```
1  #Viewing data of 10 to 20 columns
2
3  df.iloc[5:10,10:21]
```

Out[11]:

	deal_valuation	ashneer_present	anupam_present	aman_present	namita_present	vineeta
5	0.00	1	1	1	1	
6	0.00	1	1	1	1	
7	1250.00	1	1	1	1	
8	133.33	1	1	1	1	
9	200.00	1	1	1	1	

In [12]:

```python
# Basic data information

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 [13]:

```
1  # Describe count,mean,std deviation
2  df.describe()
```

Out[13]:

	episode_number	pitch_number	deal	pitcher_ask_amount	ask_equity	ask_va
count	117.000000	117.000000	117.000000	117.000000	117.000000	117.0
mean	18.735043	59.000000	0.555556	319.854709	5.188034	3852.4
std	10.070778	33.919021	0.499041	2767.842777	3.892121	11931.0
min	1.000000	1.000000	0.000000	0.001010	0.250000	0.0
25%	10.000000	30.000000	0.000000	45.000000	2.500000	666.0
50%	19.000000	59.000000	1.000000	50.000000	5.000000	1250.0
75%	27.000000	88.000000	1.000000	80.000000	7.500000	2857.
max	35.000000	117.000000	1.000000	30000.000000	25.000000	120000.0

8 rows × 26 columns
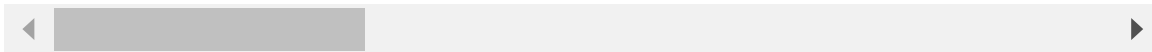
2: Cleaning the data

Checking missing values if yes then handle them accordingly

from above output , we can say that the data is simple as it does not have missing values

In [14]:

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

Out[14]:

```
episode_number          0
pitch_number            0
brand_name              0
idea                    0
deal                    0
pitcher_ask_amount      0
ask_equity              0
ask_valuation           0
deal_amount             0
deal_equity             0
deal_valuation          0
ashneer_present         0
anupam_present          0
aman_present            0
namita_present          0
vineeta_present         0
peyush_present          0
ghazal_present          0
ashneer_deal            0
anupam_deal             0
aman_deal               0
namita_deal             0
vineeta_deal            0
peyush_deal             0
ghazal_deal             0
total_sharks_invested   0
amount_per_shark        0
equity_per_shark        0
dtype: int64
```

Checking duplicacy of data

In [15]:

```
1  df.duplicated()
```

Out[15]:

```
0      False
1      False
2      False
3      False
4      False
       ...
112    False
113    False
114    False
115    False
116    False
Length: 117, dtype: bool
```

above output shows that all rows are unique so no need to drop any rows all rows are considered as
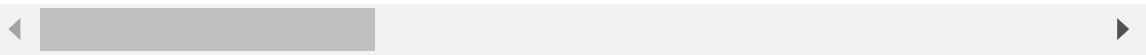
relevant¶

3.INSIGHTS

In [16]:

```
1  df.head(2)
```

Out[16]:

	episode_number	pitch_number	brand_name	idea	deal	pitcher_ask_amount	ask_equ
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	1

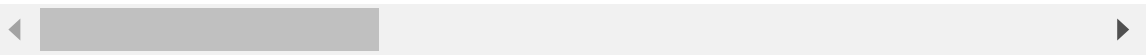2 rows × 28 columns

How many Dublicated rows present

In [17]:

```
1  df[df.duplicated()]
```

Out[17]:

	episode_number	pitch_number	brand_name	idea	deal	pitcher_ask_amount	ask_equity

0 rows × 28 columns

Q1. Find the number of episodes

In [18]:

```
1  df["episode_number"].nunique()
```

Out[18]:

35

In [19]:

```
1  df['deal'].value_counts()
```

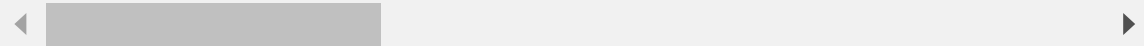Out[19]:

```
1    65
0    52
Name: deal, dtype: int64
```

In [20]:

```
1  df[df['deal_amount']==df['deal_amount'].max()]
```

Out[20]:

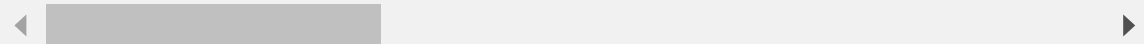|	episode_number	pitch_number	brand_name	idea	deal	pitcher_ask_amount	ask_eq
50	17	51	Aas Vidyalaya	EdTech App	1	150.0	

1 rows × 28 columns

In [21]:

```
1  df[df['ask_valuation']==df['ask_valuation'].max()]
```

Out[21]:

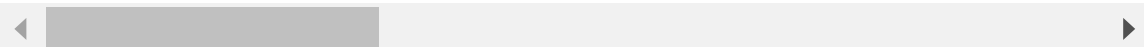|	episode_number	pitch_number	brand_name	idea	deal	pitcher_ask_amount	ask_equ
30	11	31	Gopal's 56	Fiber Ice Cream	0	30000.0	2

1 rows × 28 columns

In [22]:

```
1  df[df['ask_equity']==df['ask_equity'].min()]
```

Out[22]:

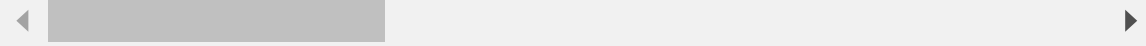|	episode_number	pitch_number	brand_name	idea	deal	pitcher_ask_amount	ask_e
6	3	7	Qzense Labs	Food Freshness Detector	0	100.0	

1 rows × 28 columns

In [23]:

```
1  df[df['deal_valuation']==df['deal_valuation'].max()]
```

Out[23]:

	episode_number	pitch_number	brand_name	idea	deal	pitcher_ask_amount	ask_equity
12	5	13	Revamp Moto	E-Bike	1	100.0	1.0

1 rows × 28 columns

◁ [▬▬▬▬▬▬▬▬▬   ] ▷

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

In [24]:

```
1  df[["pitcher_ask_amount","ask_equity","ask_valuation"]].agg([max,min,"mean"])
```

Out[24]:

	pitcher_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 [25]:

```
1  df.groupby(['episode_number'])[['pitcher_ask_amount', 'ask_equity','ask_valuation']
```

Out[25]:

episode_number	pitcher_ask_amount max	pitcher_ask_amount min	ask_equity max	ask_equity min	ask_valuation max	ask_valuation min
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
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 [26]:

```python
#dataframe showing those brands name who get sharks between 2 to 4:
df[(df["total_sharks_invested"] >=2) & (df["total_sharks_invested"] <=5)][["brand_n
```

Out[26]:

	brand_name	total_sharks_invested
0	BluePine Industries	3
1	Booz scooters	2
2	Heart up my Sleeves	2
9	Cosiq	2
11	Bummer	2
12	Revamp Moto	2
15	Skippi Pops	5
18	Raising Superstars	2
21	Beyond Snack	2
22	Vivalyf Innovations- Easy Life	2
24	Altor	2
25	Ariro	2
27	Nuutjob	3
28	Meatyour	3
29	EventBeep	3
32	Farda	2
35	LOKA	3
36	Annie	3
37	Caragreen	2
38	The Yarn Bazaar	4
39	The Renal Project	2
44	Cocofit	3
45	Bamboo India	2
47	Beyond Water	2
48	Let's Try	2
49	Find Your Kicks India	5
50	Aas Vidyalaya	3
58	WeSTOCK	4
63	IN A CAN	5
64	Get a Whey	3
66	The Quirky Nari	2
67	Hair Originals	3
75	The Sass Bar	2
79	Sunfox Technologies	5
85	Watt Technovations	4
88	Humpy A2	3
90	Gold Safe Solutions Ind.	3

	brand_name	total_sharks_invested
91	Wakao Foods	3
95	Kabaddi Adda	2
108	Tweek Labs	3
109	Proxgy	2
110	Nomad Food Project	4
114	Jain Shikanji	4

Q5. Episode wise minimum and maximum of sharks invested ?

In [27]:

```python
1  df.groupby(["episode_number"])["total_sharks_invested"].agg([max,min])
```

Out[27]:

episode_number	max	min
1	3	2
2	1	0
3	1	0
4	2	0
5	2	0
6	5	0
7	2	0
8	2	1
9	2	0
10	3	3
11	2	0
12	3	0
13	4	2
14	2	0
15	3	1
16	2	0
17	5	0
18	1	0
19	4	0
20	5	0
21	3	1
22	3	0
23	1	0
24	2	0
25	5	0
26	1	0
27	4	0
28	3	0
29	3	0
30	2	0
31	1	0
32	1	0
33	3	0
34	4	0
35	4	0

Q6. Minimum and Maximum sharks invested in each brand ?

In [28]:

```
1  df.groupby(["brand_name"])["total_sharks_invested"].agg([max,min])
```

Out[28]:

	max	min
brand_name		
ARRCOAT Surface Textures	1	1
Aas Vidyalaya	3	3
Agro tourism	0	0
Aliste Technologies	0	0
Alpino	0	0
...	...	...
Vivalyf Innovations- Easy Life	2	2
Wakao Foods	3	3
Watt Technovations	4	4
WeSTOCK	4	4
Woloo	0	0

117 rows × 2 columns

Q7. Brand names who ask for 1 crore and got a deal ?

In [29]:

```
1  # dataframe showing those brand names in which pitcher_ask_amt and deal amount == 1(
2  df[(df["pitcher_ask_amount"] == 100) & (df["deal_amount"] == 100)][["brand_name","p
```

Out[29]:

	brand_name	pitcher_ask_amount	deal_amount
12	Revamp Moto	100.0	100.0
18	Raising Superstars	100.0	100.0
39	The Renal Project	100.0	100.0
64	Get a Whey	100.0	100.0
79	Sunfox Technologies	100.0	100.0
87	Insurance Samadhan	100.0	100.0

Q7. Brand names who asked equity between 0 to 50 ?

In [30]:

```python
df[(df["ask_equity"] >= 0) & (df["ask_equity"] <=50)][["brand_name","ask_equity"]]
```

Out[30]:

	brand_name	ask_equity
0	BluePine Industries	5.0
1	Booz scooters	15.0
2	Heart up my Sleeves	10.0
3	Tagz Foods	1.0
4	Head and Heart	5.0
...	...	...
112	Green Protein	2.0
113	On2Cook	1.0
114	Jain Shikanji	8.0
115	Woloo	4.0
116	Elcare India	2.5

117 rows × 2 columns

Q8. Brand names where deal equity is between 0 to 50 ?

In [31]:

```python
# dataframe showing those brand names in which deal_equity is between 0 to 50:
df[(df["deal_equity"] >= 0) & (df["deal_equity"] <=50)]
```

Out[31]:

	episode_number	pitch_number	brand_name	idea	deal	pitcher_ask_amount	a
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	
...	...	...	...	...	...	...	
112	34	113	Green Protein	Plant-Based Protein	0	60.0	
113	34	114	On2Cook	Fastest Cooking Device	0	100.0	
114	35	115	Jain Shikanji	Lemonade	1	40.0	
115	35	116	Woloo	Washroom Finder	0	50.0	
116	35	117	Elcare India	Carenting for Elders	0	100.0	

116 rows × 28 columns
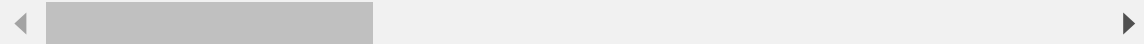
Q9. Find the number of brands participated in each episode ?

In [32]:

```python
#Insight :
#max no of brands participated per episode: 4
#Min. number of brands participated per episode : 3

print(df["episode_number"].value_counts())
```

```
18    4
30    4
17    4
16    4
22    4
23    4
27    4
31    4
32    4
33    4
34    4
19    4
29    3
28    3
20    3
26    3
25    3
24    3
21    3
1     3
2     3
15    3
14    3
13    3
12    3
11    3
10    3
9     3
8     3
7     3
6     3
5     3
4     3
3     3
35    3
Name: episode_number, dtype: int64
```

Q10. How many sharks participated in this show and What were their names ?

In [33]:

```
1  print("sharks name as below:\n\n",df.columns[11:17],"\n\n")
2  print("total number of sharks/investers in this series",len(df.columns[11:17]))
```

sharks name as below:

 Index(['ashneer_present', 'anupam_present', 'aman_present', 'namita_pres
ent',
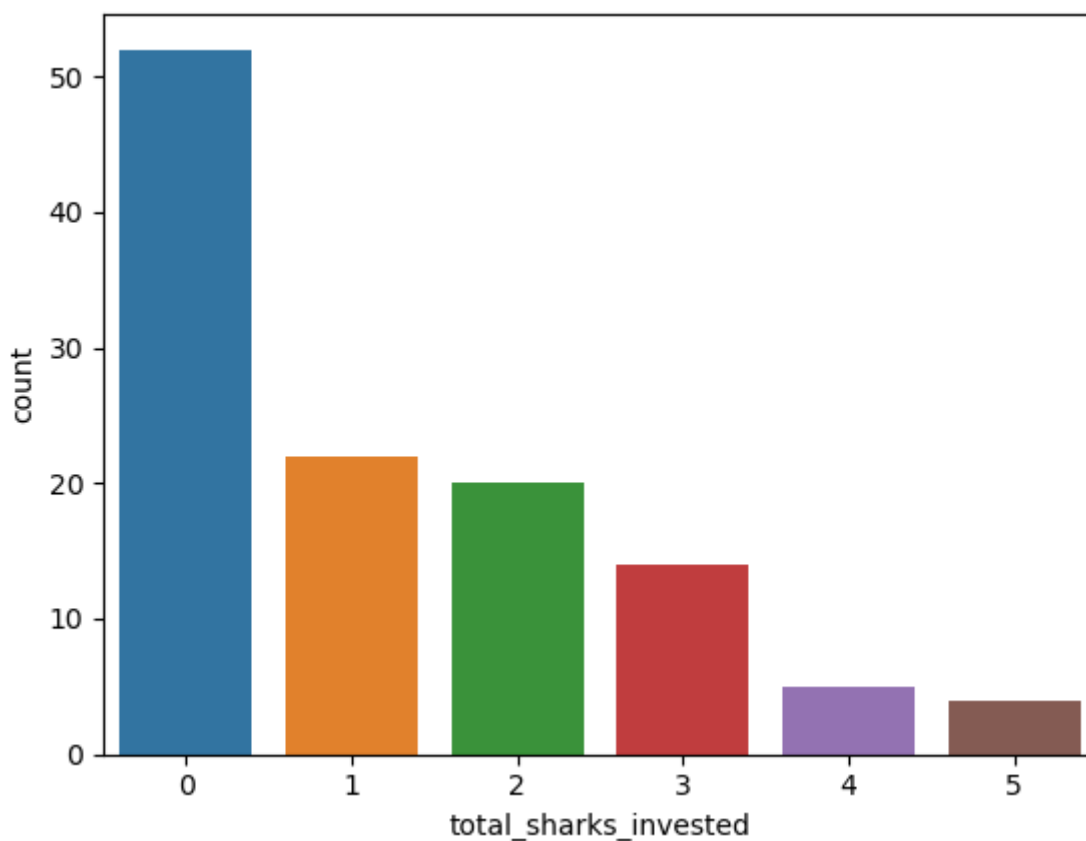   'vineeta_present', 'peyush_present'],
      dtype='object')

total number of sharks/investers in this series 6

In [34]:

```
1  import seaborn as sns
2  sns.countplot(x='total_sharks_invested',data=df)
```

Out[34]:

<AxesSubplot:xlabel='total_sharks_invested', ylabel='count'>

Q11. Find appearance of each sharks?

In [35]:

```python
l1=[]
for i in df.columns[11:17]:
    l1.append(df[i].sum())

#sharks_appearance=pd.DataFrame(d)
#sharks_appearance
sharks_appearance_data={"sharks":["ashneer_present","anupam_present","aman_present"
sharks_appearance=pd.DataFrame(sharks_appearance_data)
plt.bar(sharks_appearance["sharks"],sharks_appearance["sharks_appearance_"])
plt.xticks(rotation=90)
plt.title("sharks appearance in series")
```

Out[35]:

Text(0.5, 1.0, 'sharks appearance in series')
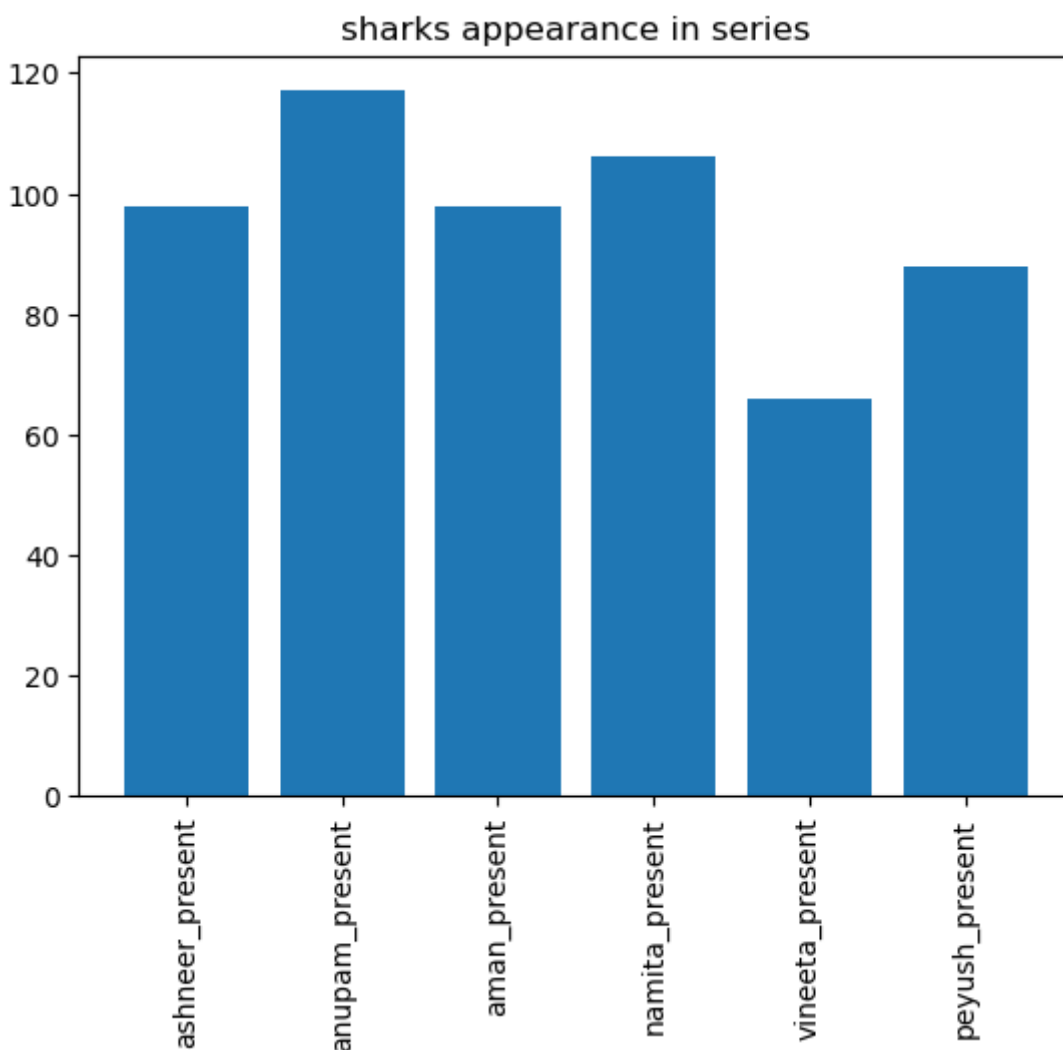
insight: Shark named anupam presented in all the episodes Shark named Ghazal presented least

Q12. How many entrepreneurs were present ?

In [36]:

```
1  df["brand_name"].nunique()
```

Out[36]:

117

insight : total number of entrepreneurs=117 who presented their ideas

In [37]:

```
1  df.columns[18:25]
```

Out[37]:

```
Index(['ashneer_deal', 'anupam_deal', 'aman_deal', 'namita_deal',
       'vineeta_deal', 'peyush_deal', 'ghazal_deal'],
      dtype='object')
```

Q13. How many times each investers invested the deal ?

lowest deal count of the invester named : Ghazal Highest deal count of the invester named : anumpam

In [38]:

```
1  for i in df.columns[18:25]:
2      s=df[i].sum()
3      print(i, "deal count :",s)
```

```
ashneer_deal deal count : 21
anupam_deal deal count : 24
aman_deal deal count : 28
namita_deal deal count : 22
vineeta_deal deal count : 15
peyush_deal deal count : 27
ghazal_deal deal count : 7
```

In [39]:

```python
l = []
for i in df.columns[18:25]:
    x=df[df[i]==1]
    s=x["amount_per_shark"].sum()
    l.append(s)
    print(i,"invested",s,"amount")
amount = sorted(l,reverse=True)
amount
```

ashneer_deal invested 494.33333333 amount
anupam_deal invested 533.83360253 amount
aman_deal invested 887.500016693 amount
namita_deal invested 648.333602533 amount
vineeta_deal invested 328.3333333300001 amount
peyush_deal invested 719.6669191630001 amount
ghazal_deal invested 130.0002525 amount

Out[39]:

```
[887.500016693,
 719.6669191630001,
 648.333602533,
 533.83360253,
 494.33333333,
 328.3333333300001,
 130.0002525]
```

Insight : total amount invested by each sharks in descending order is as follows : aman : invested 887.500016693 amount

peyush :invested 719.6669191630001 amount

namita : invested 648.333602533 amount

anupam : invested 533.83360253 amount

ashneer invested 494.33333333 amount

vineeta : invested 328.3333333300001 amount

ghazal :invested 130.0002525 amount

so highest amount i.e 887.500016693 invested by shark name Mr.A aman Lowest amount invested i.e 130.0002525 amount by shark name Ghazal

Q.14 Find the equity percent that each sharks gets ?

In [40]:

```python
l = []
for i in df.columns[18:25]:
    x=df[df[i]==1]
    s=x["equity_per_shark"].sum()
    l.append(s)
    print(i,"gets total",s,"equity")
amount=sorted(l,reverse=True)
amount
```

```
ashneer_deal gets total 93.249999999 equity
anupam_deal gets total 166.35 equity
aman_deal gets total 160.263333334 equity
namita_deal gets total 134.78333333400002 equity
vineeta_deal gets total 131.533333333 equity
peyush_deal gets total 315.84999999999997 equity
ghazal_deal gets total 46.7 equity
```

Out[40]:

```
[315.84999999999997,
 166.35,
 160.263333334,
 134.78333333400002,
 131.533333333,
 93.249999999,
 46.7]
```

Insight : Highest percentage of equity owned by Peyush and lowest by Ghazal.

As we can see that Ghazal invested least amount of money and have least percentage of equity but on other hand Aman invested highest but Peyush got highest percentage of equity.

Out of 117 enterpreneurs how many ' enterpreneurs' deals were accepted¶

In [41]:

```python
df['deal'].value_counts()
```

Out[41]:

```
1    65
0    52
Name: deal, dtype: int64
```

insights:

```
    deals are accepted more rejected deals
    total number of brands whose deals had been acepted are 65 out of 117 and their
    names and ideas are given below¶
```

In [42]:

```python
df1=df[df["deal"]==1]
df1[["brand_name","idea"]]
```

Out[42]:

	brand_name	idea
0	BluePine Industries	Frozen Momos
1	Booz scooters	Renting e-bike for mobility in private spaces
2	Heart up my Sleeves	Detachable Sleeves
3	Tagz Foods	Healthy Potato Chips
7	Peeschute	Disposable Urine Bag
...	...	...
106	Colour Me Mad	Insoles
108	Tweek Labs	Sportswear
109	Proxgy	VR
110	Nomad Food Project	Bacon Jams
114	Jain Shikanji	Lemonade

65 rows × 2 columns

now, we are making new dataset named x only with successful deals

In [43]:

```
1  x = df[df["deal"]==1]
2  x
```

Out[43]:

	episode_number	pitch_number	brand_name	idea	deal	pitcher_ask_amount	as
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	
7	3	8	Peeschute	Disposable Urine Bag	1	75.0	
...	...	...	...	...	...	...	
106	33	107	Colour Me Mad	Insoles	1	40.0	
108	33	109	Tweek Labs	Sportswear	1	40.0	
109	33	110	Proxgy	VR	1	35.0	
110	34	111	Nomad Food Project	Bacon Jams	1	40.0	
114	35	115	Jain Shikanji	Lemonade	1	40.0	

65 rows × 28 columns

Q15. Find the total number of amount invested in this show?

In [44]:

```
1  x["deal_amount"].sum()
```

Out[44]:

```
3642.00106
```

Q16. Find the highest equity a shark gets ?

In [45]:

```
1  print("Highest % of equity  per shark ",x["equity_per_shark"].max())
```

```
Highest % of equity  per shark  75.0
```

Q17. Find the most attracted ideas accepted by sharks ?

In [46]:

```
1  df['total_sharks_invested'].unique()
```

Out[46]:

```
array([3, 2, 1, 0, 5, 4], dtype=int64)
```

In [47]:

```
1  df[df['total_sharks_invested']>3][['brand_name','idea']]
```

Out[47]:

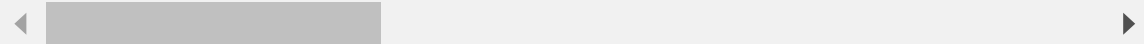|	brand_name	idea
15	Skippi Pops	Ice-Pops
38	The Yarn Bazaar	Yarn-Trading App
49	Find Your Kicks India	Sneaker Resale
58	WeSTOCK	Livestock health monitoring AI
63	IN A CAN	Can Cocktails
79	Sunfox Technologies	Portable ECG Device
85	Watt Technovations	Ventilated PPE Kits
110	Nomad Food Project	Bacon Jams
114	Jain Shikanji	Lemonade

In [48]:

```
1  df[df['ask_valuation']==df['ask_valuation'].max()]
```

Out[48]:

	episode_number	pitch_number	brand_name	idea	deal	pitcher_ask_amount	ask_equ
30	11	31	Gopal's 56	Fiber Ice Cream	0	30000.0	2

1 rows × 28 columns

In [49]:

```
1  df[df['ask_equity']==df['ask_equity'].min()]
```

Out[49]:

	episode_number	pitch_number	brand_name	idea	deal	pitcher_ask_amount	ask_e
6	3	7	Qzense Labs	Food Freshness Detector	0	100.0	

1 rows × 28 columns

◁ ▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬▬ ▷

Q18. How many companies in front Owner speak ?

In [50]:

```
1  list = ['anupam_present','aman_present','namita_present','vineeta_present','peyush_
2  for i in list:
3      pres = df[i].sum()
4      print(i[:-8],"present in front of",pres,"companies" )
```

```
anupam present in front of 117 companies
aman present in front of 98 companies
namita present in front of 106 companies
vineeta present in front of 66 companies
peyush present in front of 88 companies
ghazal present in front of 26 companies
ashneer present in front of 98 companies
```

In [51]:

```python
got_in=df[df['deal_amount']>0.0].shape
not_in=df[df['deal_amount']==0.0].shape

print("The number of companies who got the investment: ", got_in[0])
print("The number of companies who didn't got the investment: ", not_in[0])
print("\n---------\n")
list1=["Got Investment", "Not Recieved Investment"]
plt.pie([got_in[0],not_in[0]], labels=list1, autopct='%.1f%%')
plt.title('Number of comapnies which got investment or not:')
plt.show()
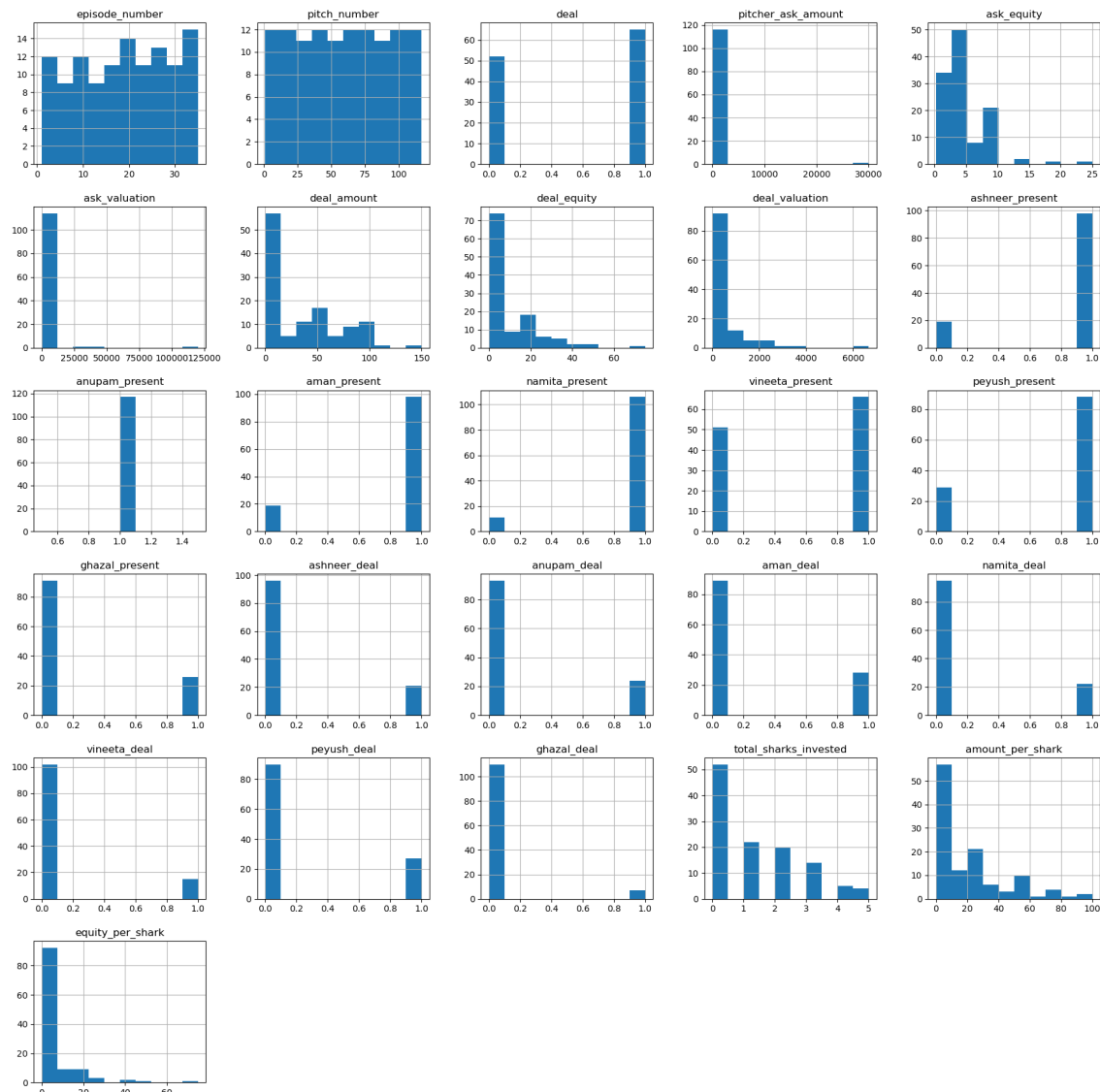```

```
The number of companies who got the investment:  65
The number of companies who didn't got the investment:  52


---------
```
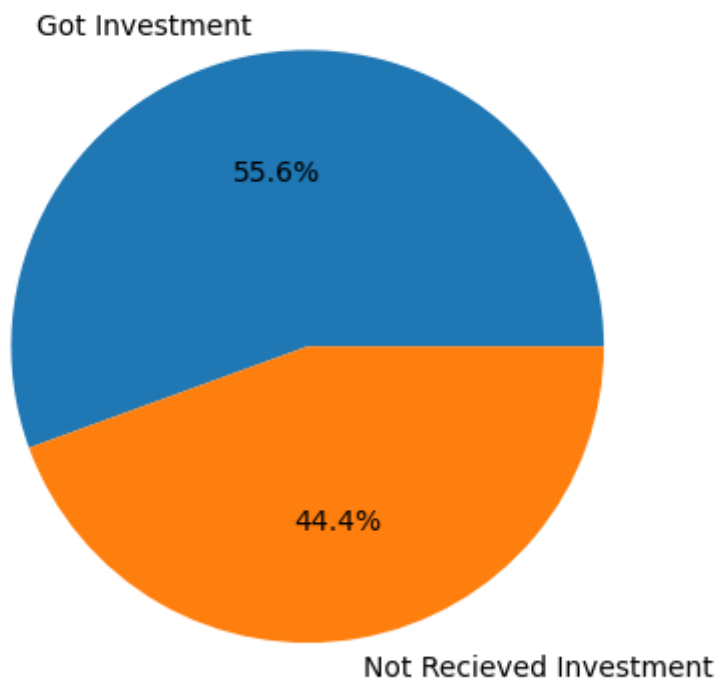
In [52]:

```python
df.hist(figsize=(22, 22), bins=10)
plt.show()
```


Q19. Percentage of companies got investment or not?

Five shark deal

In [53]:

```python
# Total shark deal
total_shark = df[df["total_sharks_invested"]==5]
figure=px.bar(total_shark, x='brand_name', y='deal_amount',title="Five Shark Deal B
                template="plotly_dark")
figure.show()
```

From above plot, we get to know that there are all total 4 five shark deal brands and also with the pitcher deal amount and the deal amount of the brand.

Highest Pitch Ask Amount:

In [54]:

```python
#Highest Pitch Ask Amount
high=df[df["pitcher_ask_amount"]>100]
figure=px.bar(high, x='brand_name', y='pitcher_ask_amount',title="Highest Pitch Ask
                  template="plotly_dark")
figure.show()
```

Ask Equity and Deal Equity:

Most of the brand asking for the high amount had not received the investment from the shark, we clearly got to know by the blue color plot which shows that no deal happened except "Aas Vidyalaya".

In [55]:

```python
#Ask Equity and Deal Equity of Highest Pitch Ask AMount Brand
figure=px.bar(high, x='brand_name', y='ask_equity',title="Ask Equity & Deal Equity
                  template="plotly_dark")
figure.show()
```

Least Pitch Amount Asked:

Among the lowest pitch amout made by the brands, only "Cocofit" and "Watt Technovations" got the investment.

The lowest amount pitch which sharks had made deal was Cocofit and Watt Technovations.

In [56]:

```python
# Least Pitch Amount Asked
low = df[df["pitcher_ask_amount"]<20]
figure=px.bar(low, x='brand_name', y='pitcher_ask_amount',title="Lowest Pitch Ask A
                  template="plotly_dark")
figure.show()
```

Q20. No. of brands the sharks have invested:?

In [57]:

```
1  #No. of brands the sharks have invested:
2  num_deal_shark=[df.ashneer_deal.sum(), df.anupam_deal.sum(), df.aman_deal.sum(), df
3  all_sharks=["Ashneer", "Anupam", "Aman", "Namita", "Vineeta", "Peyush", "Ghazal"]
4  figure=px.bar(total_shark, x=all_sharks, y=num_deal_shark,title="Number of deals do
5               template="plotly_dark")
6  figure.show()
```

Above 50% Equity taken by the shark in which brand:

In [58]:

```
1  #Above 50% Equity taken by the shark in which brand?:
2  df_equity=df[df['equity_per_shark']>=50]
3  figure=px.bar(df_equity, x='brand_name', y='equity_per_shark',title="Above 50% Equi
4               template="plotly_dark")
5  figure.show()
```

In [ ]:

```
1
```

In [ ]:

```
1
```