Exploratory Data Analysis of Shark Tank Dataset (Season 1)

Welcome to my Shark Tank project in Jupyter Notebook! In this project, I have embarked on a fascinating journey of exploring and analyzing the Shark Tank Season 1 dataset, which I obtainied from Kaggle. This dataset is a treasure trove of information about startup number, brand name, description, deal offered, start ups request amount, deal equity, amount agreed by sharks, equity per shark and more providing a comprehensive snapshot of the shark tank season 1.

Throughout this project, I have employed the powerful techniques of Exploratory Data Analysis (EDA) and feature engineering to uncover valuable insights and patters hidden within the dataset.

The collected data has been stored in the Comma Separated Value file "Shark_Tank_India_S1.csv". Each startup is uniquely identified by its startup_number. Every startup contains the following variables:-

- episode_number: Episode number out of 36 episodes
- startup_number: Startup Number out of 121 startups
- brand_name: Name of the startup
- description: Brief introduction of the startup
- startup_ask_amount_lakhs: Startup's demand amount from the sharks
- startup_ask_percentage:
- startup_ask_valuation: Valuation calculated by the startup
- deal_amount_lakhs: Amount agreed by the sharks to invest
- deal_equity: Equity agreed by both the startup and the sharks for the deal
- deal_valuation: Valuation by the company in future
- loan_element_present: If loan is taken or not
- loan_amount: Loan amount if taken
- rannvijay_present: Ranvijay is present in that episode or not
- abish_present: Abish is present in that episode or not
- aman_present: Aman is present in that episode or not
- aman_invested: Aman has invested or not
- anupam_present: Anupam is present in that episode or not
- anupam_invested: Anupam has invested or not
- ashneer_present: Ashneer is present in that episode or not
- ashneer_invested: Ashneer has invested or not
- ghazal_present: Ghazal is present in that episode or not
- ghazal_invested: Ghazal has invested or not
- namita_present: Namita is present in that episode or not
- namita_invested: Namita has invested or not

- peyush_present: Peyush is present in that episode or not
- peyush_invested: Peyush has invested or not
- vineeta_present: Vineeta is present in that episode or not
- vineeta_invested: Vineeta has invested or not
- sharks_offering: Total Sharks present in a single deal
- amount_per_shark: Deal amount divided per sharks
- equity_per_shark: Equity divided per sharks

In [1]:
```python
# Importing the necessary libraries for the project

import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```python
# Reading the data as df saved as csv

df = pd.read_csv('Shark_Tank_India_S1.csv')
```

In [3]:
```python
df
```

Out[3]:

	episode_number	startup_number	brand_name	description	deal_offered	startup_ask_amount_
0	1	1	BluePine Industries	Frozen Momos	1	
1	1	2	Booz scooters	Renting e-bike for mobility in private spaces	1	
2	1	3	Heart up my Sleeves	Detachable Sleeves	1	
3	2	4	Tagz Foods	Healthy Potato Chips	1	
4	2	5	Head and Heart	Brain Development Course	0	
...	...	...	...	...	...	...
116	35	117	Elcare India	Carenting for Elders	0	
117	36	118	Sneakare	Shoe care and storage solutions	1	
118	36	119	French Crown	Clothing Industry	0	
119	36	120	Store My Goods	Storage solutions	1	
120	36	121	Devnagri	Translation platform	0	

121 rows × 32 columns

```
In [4]:    # Checking the columns present in the data
           df.columns
```

```
Out[4]:    Index(['episode_number', 'startup_number', 'brand_name', 'description',
                  'deal_offered', 'startup_ask_amount_lakhs', 'startup_ask_percentage',
                  'startup_ask_valuation', 'deal_amount_lakhs', 'deal_equity',
                  'deal_valuation', 'loan_element_present', 'loan_amount',
                  'rannvijay_present', 'abish_present', 'aman_present', 'aman_invested',
                  'anupam_present', 'anupam_invested', 'ashneer_present',
                  'ashneer_invested', 'ghazal_present', 'ghazal_invested',
                  'namita_present', 'namita_invested', 'peyush_present',
                  'peyush_invested', 'vineeta_present', 'vineeta_invested',
                  'sharks_offering', 'amount_per_shark', 'equity_per_shark'],
                 dtype='object')
```

```
In [5]:    df.drop(['rannvijay_present', 'abish_present'], axis=1, inplace=True)
```

- Rannvijay and Abish were not among the sharks in Shark Tank season 1

```
In [6]:    # Checking 5 samples of data
           df.sample(5)
```

Out[6]:

	episode_number	startup_number	brand_name	description	deal_offered	startup_ask_amount_lak
58	19	59	WeSTOCK	Livestock health monitoring AI	1	5(
11	4	12	Bummer	Underwear	1	7!
49	17	50	Find Your Kicks India	Sneaker Resale	1	5(
41	14	42	Good Good Piggy Bank	Digital Piggy Bank	0	4!
59	19	60	KetoIndia	Customised Keto Diets for various medical issues	0	12!

5 rows × 30 columns
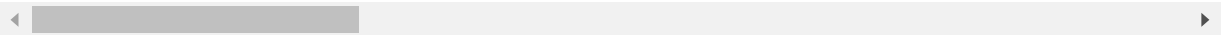
```
In [7]:    # Checking the basic information of the data

           df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 121 entries, 0 to 120
Data columns (total 30 columns):
 #   Column                    Non-Null Count   Dtype
---  ------                    --------------   -----
 0   episode_number            121 non-null     int64
```

```
 1   startup_number            121 non-null    int64
 2   brand_name                121 non-null    object
 3   description               121 non-null    object
 4   deal_offered              121 non-null    int64
 5   startup_ask_amount_lakhs  121 non-null    float64
 6   startup_ask_percentage    121 non-null    float64
 7   startup_ask_valuation     121 non-null    float64
 8   deal_amount_lakhs         121 non-null    float64
 9   deal_equity               121 non-null    float64
 10  deal_valuation            121 non-null    float64
 11  loan_element_present      121 non-null    int64
 12  loan_amount               121 non-null    int64
 13  aman_present              121 non-null    int64
 14  aman_invested             121 non-null    int64
 15  anupam_present            121 non-null    int64
 16  anupam_invested           121 non-null    int64
 17  ashneer_present           121 non-null    int64
 18  ashneer_invested          121 non-null    int64
 19  ghazal_present            121 non-null    int64
 20  ghazal_invested           121 non-null    int64
 21  namita_present            121 non-null    int64
 22  namita_invested           121 non-null    int64
 23  peyush_present            121 non-null    int64
 24  peyush_invested           121 non-null    int64
 25  vineeta_present           121 non-null    int64
 26  vineeta_invested          121 non-null    int64
 27  sharks_offering           121 non-null    int64
 28  amount_per_shark          121 non-null    float64
 29  equity_per_shark          121 non-null    float64
dtypes: float64(8), int64(20), object(2)
memory usage: 28.5+ KB
```

In [8]:
```python
# Checking the shape of the data (rows and columns)

df.shape
```

Out[8]:   (121, 30)

1. Finding the over all statistics of the data

In [9]:
```python
df.describe(include='all')
```

Out[9]:

	episode_number	startup_number	brand_name	description	deal_offered	startup_ask_amoun
count	121.000000	121.000000	121	121	121.000000	121.
unique	NaN	NaN	121	121	NaN	
top	NaN	NaN	BluePine Industries	Frozen Momos	NaN	
freq	NaN	NaN	1	1	NaN	
mean	19.305785	61.000000	NaN	NaN	0.561983	312.
std	10.375326	35.073732	NaN	NaN	0.498206	2721.
min	1.000000	1.000000	NaN	NaN	0.000000	0.
25%	11.000000	31.000000	NaN	NaN	0.000000	45.
50%	19.000000	61.000000	NaN	NaN	1.000000	50.

	episode_number	startup_number	brand_name	description	deal_offered	startup_ask_amoun
75%	28.000000	91.000000	NaN	NaN	1.000000	80.
max	36.000000	121.000000	NaN	NaN	1.000000	30000.

11 rows × 30 columns

2. Checking for Null Values in the data

In [10]:
```python
df.isnull().sum()
```

Out[10]:
```
episode_number              0
startup_number              0
brand_name                  0
description                 0
deal_offered                0
startup_ask_amount_lakhs    0
startup_ask_percentage      0
startup_ask_valuation       0
deal_amount_lakhs           0
deal_equity                 0
deal_valuation              0
loan_element_present        0
loan_amount                 0
aman_present                0
aman_invested               0
anupam_present              0
anupam_invested             0
ashneer_present             0
ashneer_invested            0
ghazal_present              0
ghazal_invested             0
namita_present              0
namita_invested             0
peyush_present              0
peyush_invested             0
vineeta_present             0
vineeta_invested            0
sharks_offering             0
amount_per_shark            0
equity_per_shark            0
dtype: int64
```

3. Counting the present days for each of the Sharks

In [11]:
```python
df['aman_present'].value_counts()
```

Out[11]:
```
1    102
0     19
Name: aman_present, dtype: int64
```

In [12]:
```python
df['anupam_present'].value_counts()
```

Out[12]:
```
1    121
Name: anupam_present, dtype: int64
```

In [13]:
```python
df['ashneer_present'].value_counts()
```

```
Out[13]:    1    98
            0    23
            Name: ashneer_present, dtype: int64
```

In [14]:
```python
df['ghazal_present'].value_counts()
```

```
Out[14]:    0    95
            1    26
            Name: ghazal_present, dtype: int64
```

In [15]:
```python
df['namita_present'].value_counts()
```

```
Out[15]:    1    110
            0     11
            Name: namita_present, dtype: int64
```

In [16]:
```python
df['peyush_present'].value_counts()
```

```
Out[16]:    1    92
            0    29
            Name: peyush_present, dtype: int64
```

In [17]:
```python
df['vineeta_present'].value_counts()
```

```
Out[17]:    1    70
            0    51
            Name: vineeta_present, dtype: int64
```

In [18]:
```python
aman_present = len(df[df['aman_present'] == 1])
anupam_present = len(df[df['anupam_present'] == 1])
vineeta_present = len(df[df['vineeta_present'] == 1])
peyush_present = len(df[df['peyush_present'] == 1])
namita_present = len(df[df['namita_present'] == 1])
ghazal_present = len(df[df['ghazal_present'] == 1])
ashneer_present = len(df[df['ashneer_present'] == 1])
```

4. Plotting the present days of each of the Sharks

a. Bar Graph Plotting

In [19]:
```python
fig = plt.figure(figsize = (10,5))
col_map = plt.get_cmap('tab20b')

present = [aman_present, anupam_present, vineeta_present,
           peyush_present, namita_present, ghazal_present, ashneer_present]
names = ['Aman', 'Anupam', 'Vineeta', 'Peyush', 'Namita', 'Ghazal', 'Ashneer']
plt.bar(names, present, color = col_map.colors, width=0.6)
plt.xlabel("Candidate Name", fontdict={'fontsize':15})
plt.ylabel("Startup Nos.", fontdict={'fontsize':15})
plt.title("Candidate Name Vs Startup Nos.",fontdict={'fontsize':20, 'weight': 550})
plt.show()
```
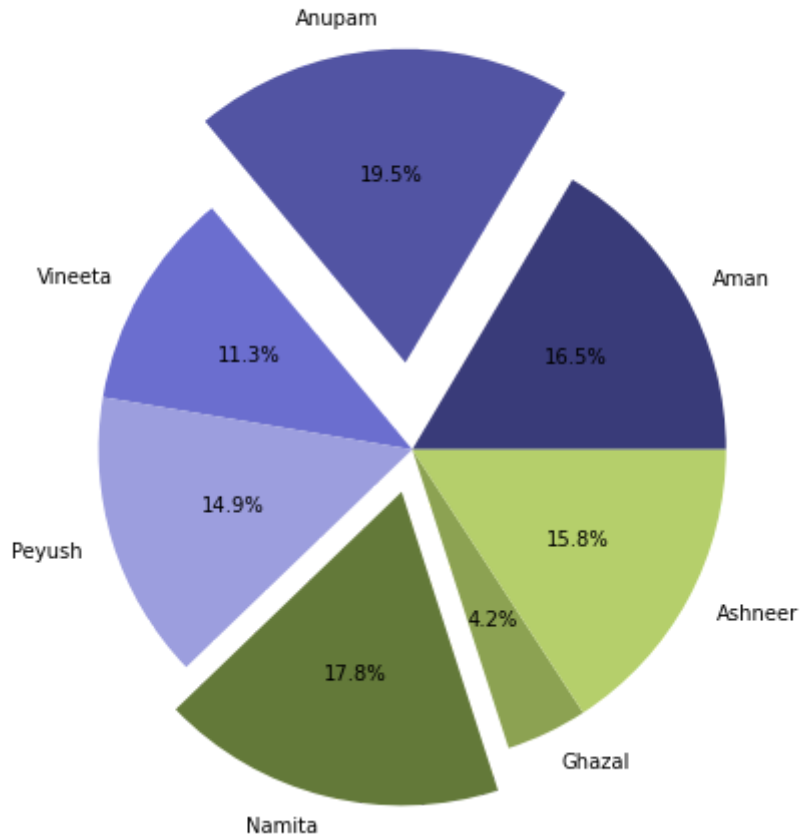
Candidate Name Vs Startup Nos.
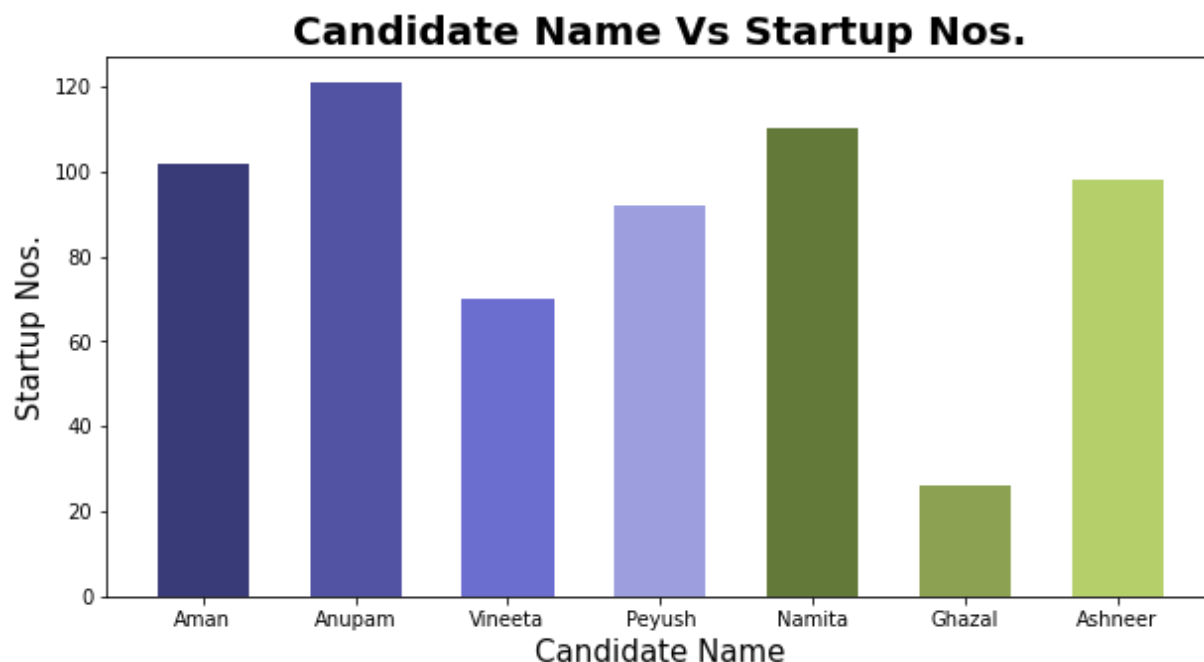
b. Pie Chart Plotting

```
In [20]:
# font = {'family': 'serif',
#         'color':  'darkred',
#         'weight': 'normal',
#         'size': 16,
#         }
col_map = plt.get_cmap('tab20b')
plt.pie(present, labels = names, radius = 1.8, autopct = '%0.1f%%',
        explode = [0,0.50,0,0,0.25,0,0], colors = col_map.colors)
plt.title("Percentage of Sharks Present in Startup",y=1.7, fontdict={'fontsize':20,
plt.show()
```

Percentage of Sharks Present in Startup

- Anupam was present in 19.5% of the 36 episodes followed by Namita with 17.8%.

```
In [21]:    # df.columns
```

5. Counting the investments by each Shark

```
In [22]:    df['aman_invested'].value_counts()
```

```
Out[22]:    0    92
            1    29
            Name: aman_invested, dtype: int64
```

```
In [23]:    df['anupam_invested'].value_counts()
```

```
Out[23]:    0    97
            1    24
            Name: anupam_invested, dtype: int64
```

```
In [24]:    df['ashneer_invested'].value_counts()
```

```
Out[24]:    0    100
            1     21
            Name: ashneer_invested, dtype: int64
```

```
In [25]:    df['ghazal_invested'].value_counts()
```

```
Out[25]:   0     114
           1       7
           Name: ghazal_invested, dtype: int64
```

```
In [26]:   df['namita_invested'].value_counts()
```

```
Out[26]:   0     97
           1     24
           Name: namita_invested, dtype: int64
```

```
In [27]:   df['peyush_invested'].value_counts()
```

```
Out[27]:   0     93
           1     28
           Name: peyush_invested, dtype: int64
```

```
In [28]:   df['vineeta_invested'].value_counts()
```

```
Out[28]:   0     105
           1      16
           Name: vineeta_invested, dtype: int64
```

```
In [29]:   aman_invested = len(df[df['aman_invested'] == 1])
           anupam_invested = len(df[df['anupam_invested'] == 1])
           ashneer_invested = len(df[df['ashneer_invested'] == 1])
           ghazal_invested = len(df[df['ghazal_invested'] == 1])
           namita_invested = len(df[df['namita_invested'] == 1])
           peyush_invested = len(df[df['peyush_invested'] == 1])
           vineeta_invested = len(df[df['vineeta_invested'] == 1])
```

6. Plotting the INVESTMENTS by each Shark

a. Bar Graph Plotting

```
In [30]:   fig = plt.figure(figsize=(10,5))
           col_map = plt.get_cmap('tab20c')

           invested = [aman_invested, anupam_invested, ashneer_invested,
                       ghazal_invested, namita_invested, peyush_invested, vineeta_invested]
           names = ['Aman', 'Anupam', 'Ashneer', 'Ghazal', 'Namita', 'Peyush', 'Vineeta']
           plt.bar(names, invested, color = col_map.colors, width=0.6)
           plt.title("Sharks Vs Deals Offered", fontdict={'fontsize':20, 'weight': 550})
           plt.xlabel("Sharks", fontdict={'fontsize':15})
           plt.ylabel("Deals offered", fontdict={'fontsize':15})

           plt.show()
```
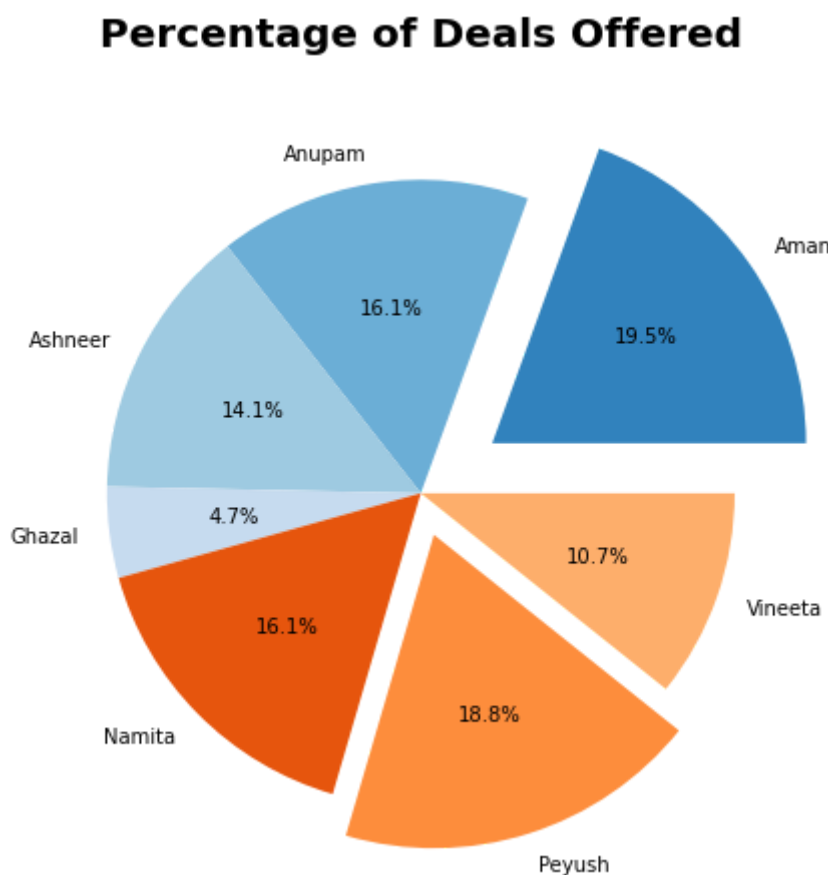
Sharks Vs Deals Offered

b. Pie Chart Plotting

In [31]:
```python
col_map = plt.get_cmap('tab20c')
plt.pie(invested, labels = names, radius = 1.8, autopct = '%0.1f%%',
        explode = [0.50, 0,0,0,0,0.25,0], colors = col_map.colors)
plt.title("Percentage of Deals Offered" ,y=1.5, fontdict={'fontsize':20, 'weight': 5
plt.show()
```

Percentage of Deals Offered

- Aman invested in most of the startups followed by Peyush, Anupam and Namita.

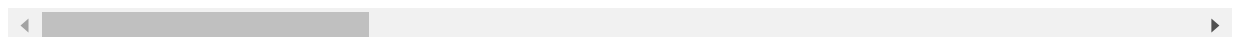## 7. Finding the top 10 Deals Offered by Sharks

In [32]:
```python
# based on deal amount
top_deals = df.sort_values('deal_amount_lakhs', ascending = False).head(10)
```

In [33]:
```python
top_deals
```

Out[33]:

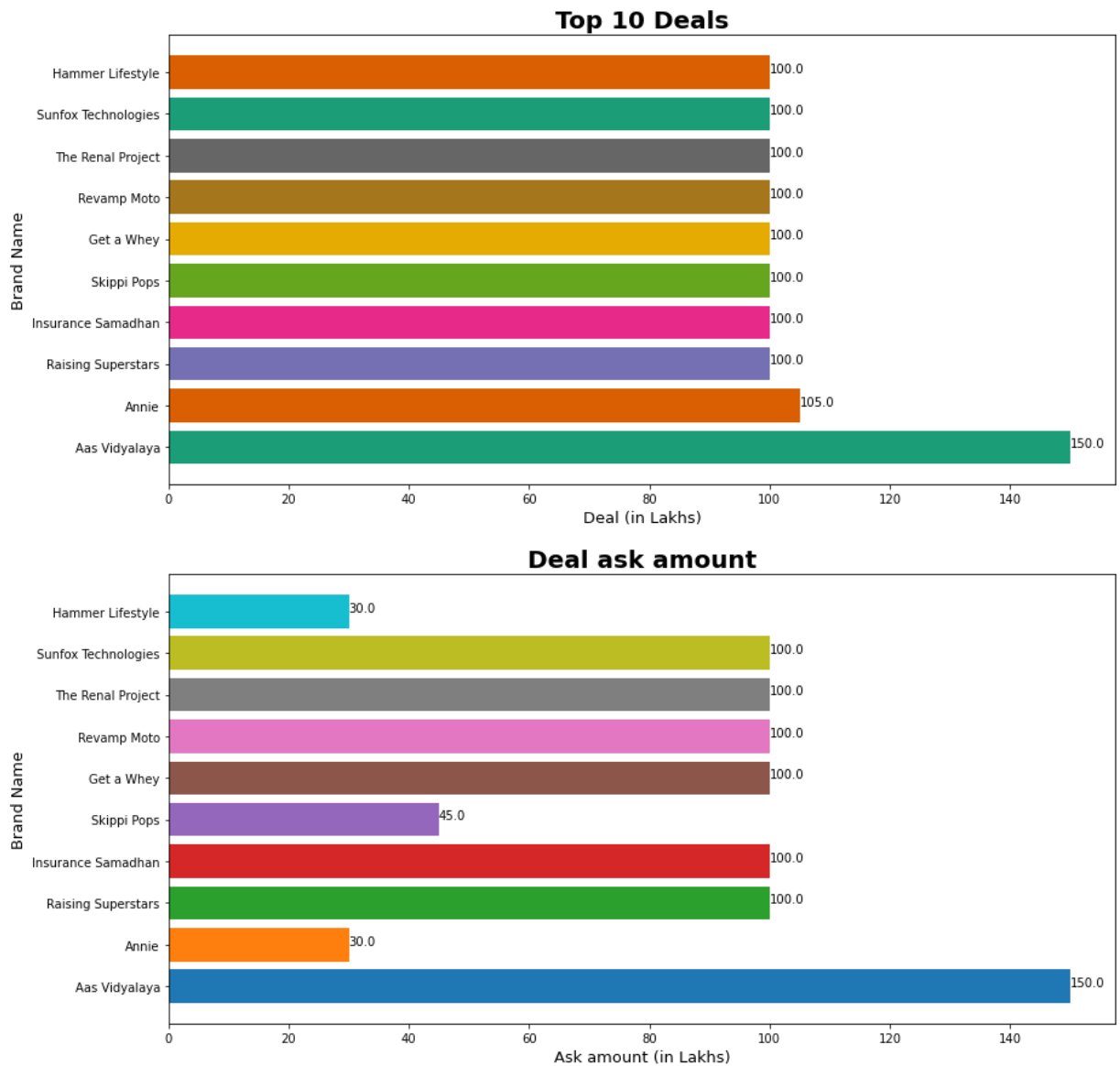|	episode_number	startup_number	brand_name	description	deal_offered	startup_ask_amount_la
50	17	51	Aas Vidyalaya	EdTech App	1	1
36	13	37	Annie	Braille Literary Device	1	
18	7	19	Raising Superstars	Child Development App	1	1
87	27	88	Insurance Samadhan	Insurance Solutions	1	1
15	6	16	Skippi Pops	Ice-Pops	1	
64	21	65	Get a Whey	Sugar-Free Icecream	1	1
12	5	13	Revamp Moto	E-Bike	1	1
39	14	40	The Renal Project	Home Dialysis Treatment	1	1
79	25	80	Sunfox Technologies	Portable ECG Device	1	1
42	15	43	Hammer Lifestyle	Smart Audio Products	1	

10 rows × 30 columns

8. Plotting the top 10 Deals offered by Sharks

In [34]:
```python
fig, axs = plt.subplots(2,1, figsize=(14, 15))

col_map = plt.get_cmap('Dark2')
axs[0].barh(top_deals['brand_name'], top_deals['deal_amount_lakhs'], color = col_map
axs[0].set_title('Top 10 Deals', fontdict={'fontsize':20, 'weight': 550})
axs[0].set_xlabel('Deal (in Lakhs)', fontdict={'fontsize':13})
axs[0].set_ylabel('Brand Name',fontdict={'fontsize':13})
for i in range(len(top_deals)):
    axs[0].text(top_deals['deal_amount_lakhs'].iloc[i], top_deals['brand_name'].iloc

col_map = plt.get_cmap('tab10')
axs[1].barh(top_deals['brand_name'], top_deals['startup_ask_amount_lakhs'], color =
axs[1].set_title('Deal ask amount', fontdict={'fontsize':20, 'weight': 550})
axs[1].set_xlabel('Ask amount (in Lakhs)', fontdict={'fontsize':13})
axs[1].set_ylabel('Brand Name',fontdict={'fontsize':13})
```

```
for i in range(len(top_deals)):
    axs[1].text(top_deals['startup_ask_amount_lakhs'].iloc[i], top_deals['brand_name
```

Top 10 Deals

Hammer Lifestyle	100.0
Sunfox Technologies	100.0
The Renal Project	100.0
Revamp Moto	100.0
Get a Whey	100.0
Skippi Pops	100.0
Insurance Samadhan	100.0
Raising Superstars	100.0
Annie	105.0
Aas Vidyalaya	150.0

Deal (in Lakhs)

Deal ask amount

Hammer Lifestyle	30.0
Sunfox Technologies	100.0
The Renal Project	100.0
Revamp Moto	100.0
Get a Whey	100.0
Skippi Pops	45.0
Insurance Samadhan	100.0
Raising Superstars	100.0
Annie	30.0
Aas Vidyalaya	150.0

Ask amount (in Lakhs)

- Aas Vidyalaya (EdTech App) was the top deal in the first season of Shark Tank with 150 Lakhs deal amount.
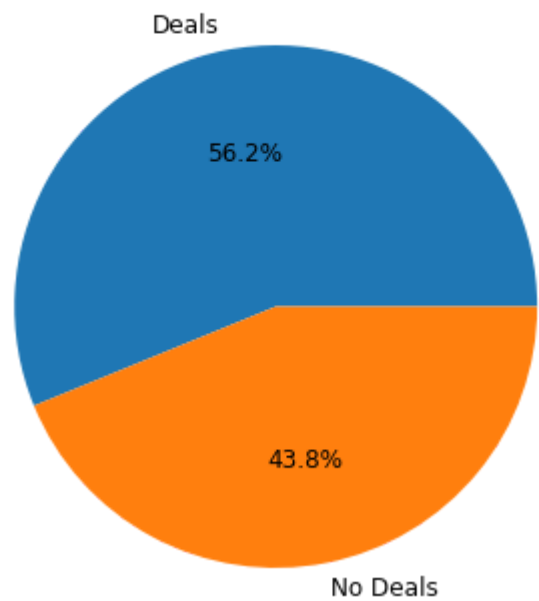
9. Total Deals taking place out of 121

In [35]:
```python
deals_taking_place = df[df['deal_offered'] == 1].shape[0]
```

In [36]:
```python
deals_not_taking_place = df[df['deal_offered'] == 0].shape[0]
```

In [37]:
```python
plt.pie([deals_taking_place, deals_not_taking_place], textprops={'fontsize':12}, lab
plt.title('Deals taking place', y=1.3, fontdict={'fontsize':20, 'weight': 550})
plt.show()
```

Deals taking place

Deals

56.2%

43.8%

No Deals

- 56.2% of 121 deals took place in the season 1 of shark tank.

In [ ]: