Kaggle Survey 2022

In [1]:
```
# Importing required libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
import sklearn as sns
import warnings
warnings.filterwarnings('ignore')
```

In [2]:
```
# Import the dataset (kaggle)
df=pd.read_csv("kaggle_survey_2022_responses.csv")
```

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

Out[3]:

	Duration (in seconds)	Q2	Q3	Q4	Q5	Q6_1	Q6_2	Q6_3	Q6_4	Q6_5	...	Q44_3	Q44_4	
0	Duration (in seconds)	What is your age (# years)?	What is your gender? - Selected Choice	In which country do you currently reside?	Are you currently a student? (high school, uni...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	...	Who/what are your favorite media sources that ...	Who/what are your favorite media sources that ...	W m
1	121	30-34	Man	India	No	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	
2	462	30-34	Man	Algeria	No	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	
3	293	18-21	Man	Egypt	Yes	Coursera	edX	NaN	DataCamp	NaN	...	NaN	Kaggle (notebooks, forums, etc)	
4	851	55-59	Man	France	No	Coursera	NaN	Kaggle Learn Courses	NaN	NaN	...	NaN	Kaggle (notebooks, forums, etc)	(fo
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
23993	331	22-24	Man	United States of America	Yes	NaN	NaN	NaN	NaN	NaN	...	NaN	Kaggle (notebooks, forums, etc)	
23994	330	60-69	Man	United States of America	Yes	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	

	Duration (in seconds)	Q2	Q3	Q4	Q5	Q6_1	Q6_2	Q6_3	Q6_4	Q6_5	...	Q44_3	Q44_4
23995	860	25-29	Man	Turkey	No	NaN	NaN	NaN	DataCamp	NaN	...	NaN	Kaggle (notebooks, forums, etc)
23996	597	35-39	Woman	Israel	No	NaN	NaN	Kaggle Learn Courses	NaN	NaN	...	NaN	NaN
23997	303	18-21	Man	India	Yes	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN

23998 rows × 296 columns

In [4]:
```
# All Column Names
df.columns
```

Out[4]:
```
Index(['Duration (in seconds)', 'Q2', 'Q3', 'Q4', 'Q5', 'Q6_1', 'Q6_2', 'Q6_3',
       'Q6_4', 'Q6_5',
       ...
       'Q44_3', 'Q44_4', 'Q44_5', 'Q44_6', 'Q44_7', 'Q44_8', 'Q44_9', 'Q44_10',
       'Q44_11', 'Q44_12'],
      dtype='object', length=296)
```

In [5]:
```
#Basic commands, here we get no. of rows & columns.
df.shape
```

Out[5]:
```
(23998, 296)
```

In [6]:
```
#Check the null values presene in the dataset.
df.isnull().sum()
```

Out[6]:
```
Duration (in seconds)        0
Q2                           0
Q3                           0
Q4                           0
Q5                           0
                           ...
Q44_8                    16231
Q44_9                    20193
Q44_10                   22271
Q44_11                   22729
Q44_12                   23162
Length: 296, dtype: int64
```

In [7]:
```
# To extract unique values,total count, frequency & so on...
df.describe()
```

Out[7]:

	Duration (in seconds)	Q2	Q3	Q4	Q5	Q6_1	Q6_2	Q6_3	Q6_4	Q6_5	...	Q44_3	Q44_4	Q44_5
count	23998	23998	23998	23998	23998	9700	2475	6629	3719	945	...	2679	11182	4007
unique	4329	12	6	59	3	2	2	2	2	2	...	2	2	2
top	230	18-21	Man	India	No	Coursera	edX	Kaggle Learn Courses	DataCamp	Fast.ai	...	Reddit (r/machinelearning, etc)	Kaggle (notebooks, forums, etc)	Course Forums (forums.fast.ai, Coursera forums...
freq	59	4559	18266	8792	12036	9699	2474	6628	3718	944	...	2678	11181	4006

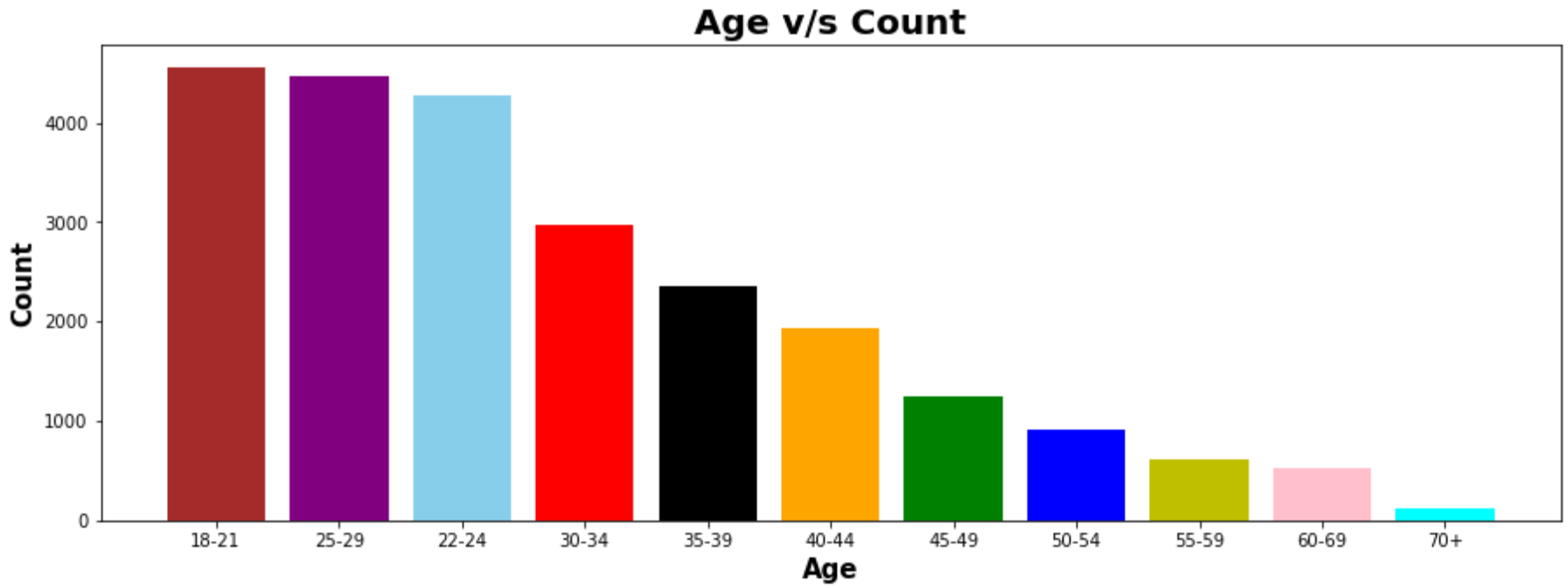
4 rows × 296 columns

What is your age (# years)?

In [8]:
```
# Here we find the total count for the range of age.
df['Q2'].value_counts()[:-1]
```

Out[8]:
```
18-21    4559
25-29    4472
22-24    4283
30-34    2972
35-39    2353
40-44    1927
45-49    1253
50-54     914
55-59     611
60-69     526
70+       127
Name: Q2, dtype: int64
```

In [9]:
```
# Plot a Bar graph
plt.figure(figsize=(15,5))
plt.bar(list(df['Q2'].value_counts().keys())[0:11],list(df['Q2'].value_counts())[0:11],color=['brown','purple','skyblu
plt.title('Age v/s Count',fontweight='bold',fontsize=20)
plt.xlabel('Age',fontweight='bold',fontsize=15)
plt.ylabel('Count',fontweight='bold',fontsize=15)
plt.show()
```



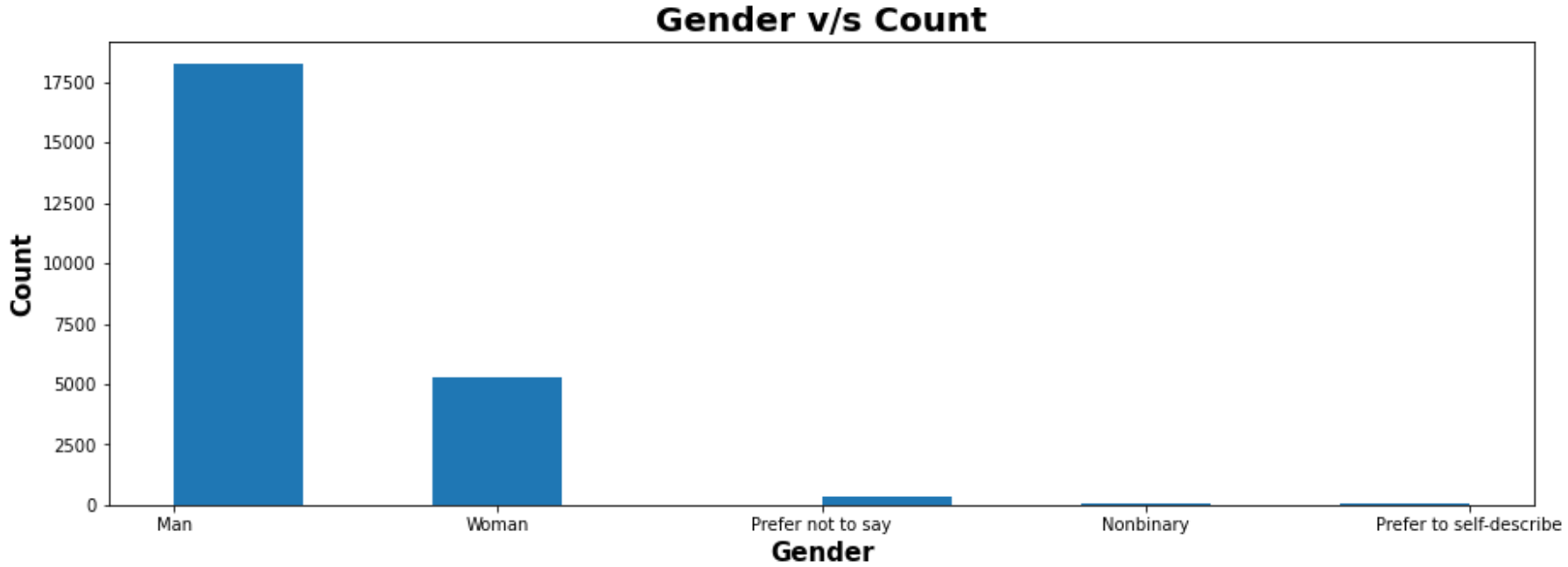
Age of 18 to 21 years are mostly participated in this survey with a count of 4559 nos.

What is your gender?

In [10]:
```
# Here we find total count of the participants considering their gender.
df['Q3'].value_counts()[:-1]
```

Out[10]:
```
Man                        18266
Woman                       5286
Prefer not to say            334
Nonbinary                     78
Prefer to self-describe       33
Name: Q3, dtype: int64
```

In [11]:
```
#Plot a histogram.
plt.figure(figsize=(15,5))
plt.hist(df['Q3'][1:])
plt.title('Gender v/s Count',fontweight='bold',fontsize=20)
plt.xlabel('Gender',fontweight='bold',fontsize=15)
plt.ylabel('Count',fontweight='bold',fontsize=15)
plt.show()
```



Total 18266 Men & 5286 Women participated in the survey.

In which country do you currently reside?

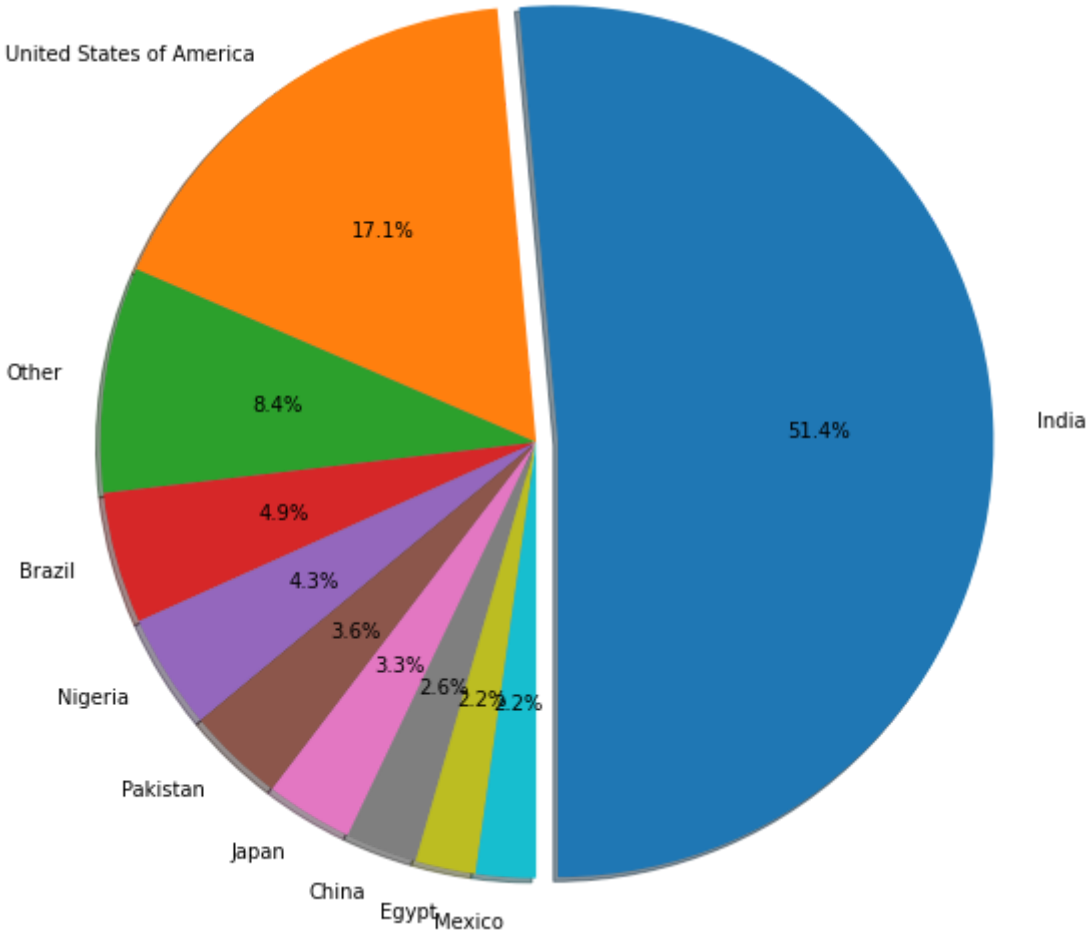
In [12]:
```
# Here we find country of the participants.
country = df['Q4'].value_counts()[:-1]
country
```

Out[12]:

India	8792
United States of America	2920
Other	1430
Brazil	833
Nigeria	731
Pakistan	620
Japan	556
China	453
Egypt	383
Mexico	380
Indonesia	376
Turkey	345
Russia	324
South Korea	317
France	262
United Kingdom of Great Britain and Northern Ireland	258
Canada	257
Spain	257
Colombia	256
Bangladesh	251
Taiwan	242
Viet Nam	212
Argentina	204
Kenya	201
Italy	182
Morocco	177
Australia	142
Thailand	132
Tunisia	125
Peru	121
Iran, Islamic Republic of...	120
Chile	115
Poland	113
South Africa	109
Philippines	108
Netherlands	108
Ghana	107
Israel	102
Germany	99
Ethiopia	98
United Arab Emirates	94

```
Portugal                                    87
Saudi Arabia                                84
Ukraine                                     79
Sri Lanka                                   77
Nepal                                       75
Malaysia                                    74
Singapore                                   68
Cameroon                                    68
Algeria                                     62
Hong Kong (S.A.R.)                          58
Zimbabwe                                    54
Ecuador                                     54
Ireland                                     53
Belgium                                     51
Romania                                     50
Czech Republic                              49
I do not wish to disclose my location       42
Name: Q4, dtype: int64
```

In [13]:
```
# Plot a Pie Chart
plt.figure(figsize=(5,5))
explode = (0.1, 0, 0, 0, 0, 0, 0, 0, 0, 0)
plt.pie(list(df['Q4'].value_counts()[:10]),
        labels=list(df['Q4'].value_counts().keys()[:10]),
        shadow=True, radius=2, autopct='%0.1f%%', explode=explode, startangle=270)
plt.show()
```



The presence of Indian participation is significantly higher with 51%.

On which platforms have you begun or completed data science courses?

In [14]:
```
# Here we extract columns from 'df' with index no 5 to 16.
course_platform = df.iloc[ : ,5:17]
course_platform.head()
```

Out[14]:

	Q6_1	Q6_2	Q6_3	Q6_4	Q6_5	Q6_6	Q6_7	Q6_8	Q6_9	Q6_10	Q6_11	
0	On which platforms have you begun or completed...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	On which platforms have you begun or completed...	Or pla ha be comp
1	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
2	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	University Courses (resulting in a university ...	NaN	
3	Coursera	edX	NaN	DataCamp	NaN	Udacity	Udemy	LinkedIn Learning	NaN	University Courses (resulting in a university ...	NaN	
4	Coursera	NaN	Kaggle Learn Courses	NaN	NaN	NaN	Udemy	NaN	NaN	NaN	NaN	

In [15]:
```
# Here we extract total count of course platforms where data science courses can be learnt.
cp1=course_platform['Q6_1'].value_counts()[:1]
cp1
```

Out[15]:
```
Coursera    9699
Name: Q6_1, dtype: int64
```

In [16]:
```
cp2=course_platform['Q6_2'].value_counts()[:1]
cp2
```

Out[16]:
```
edX    2474
Name: Q6_2, dtype: int64
```

In [17]:
```
cp3=course_platform['Q6_3'].value_counts()[:1]
cp3
```

Out[17]:
```
Kaggle Learn Courses    6628
Name: Q6_3, dtype: int64
```

In [18]:
```
cp4=course_platform['Q6_4'].value_counts()[:1]
cp4
```

Out[18]:
```
DataCamp    3718
Name: Q6_4, dtype: int64
```

In [19]:
```
cp5=course_platform['Q6_5'].value_counts()[:1]
cp5
```

Out[19]:
```
Fast.ai    944
Name: Q6_5, dtype: int64
```

In [20]:
```
cp6=course_platform['Q6_6'].value_counts()[:1]
cp6
```

Out[20]:
```
Udacity    2199
Name: Q6_6, dtype: int64
```

In [21]:
```
cp7=course_platform['Q6_7'].value_counts()[:1]
cp7
```

Out[21]:
```
Udemy    6116
Name: Q6_7, dtype: int64
```

In [22]:
```
cp8=course_platform['Q6_8'].value_counts()[:1]
cp8
```

Out[22]:
```
LinkedIn Learning    2766
Name: Q6_8, dtype: int64
```

In [23]:
```
cp9=course_platform['Q6_9'].value_counts()[:1]
cp9
```

Out[23]:
```
Cloud-certification programs (direct from AWS, Azure, GCP, or similar)    1821
Name: Q6_9, dtype: int64
```

In [24]:
```
cp10=course_platform['Q6_10'].value_counts()[:1]
cp10
```

Out[24]:
```
University Courses (resulting in a university degree)    6780
Name: Q6_10, dtype: int64
```

In [25]:
```
# create a new data frame with above information.Get the sum of elements of an iterable & then Sort them in desending
top_course_platform = pd.DataFrame([cp1,cp2,cp3,cp4,cp5,cp6,cp7,cp8,cp9,cp10]).sum().sort_values(ascending=False)
top_course_platform
```

Out[25]:
```
Coursera                                                          9699.0
University Courses (resulting in a university degree)             6780.0
Kaggle Learn Courses                                              6628.0
Udemy                                                             6116.0
DataCamp                                                          3718.0
LinkedIn Learning                                                 2766.0
edX                                                               2474.0
Udacity                                                           2199.0
Cloud-certification programs (direct from AWS, Azure, GCP, or similar)    1821.0
Fast.ai                                                            944.0
dtype: float64
```

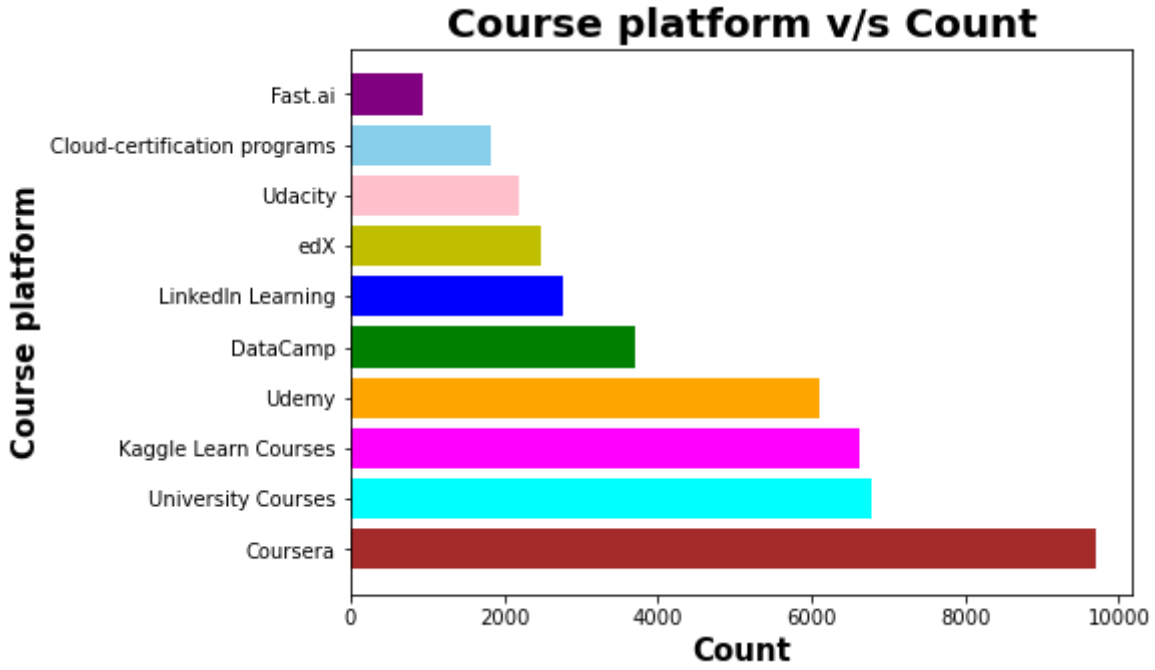
In [26]:
```
# Create a list of Categorical values.
x1 = ['Coursera','University Courses','Kaggle Learn Courses','Udemy','DataCamp','LinkedIn Learning','edX','Udacity','C
x1
```

Out[26]:
```
['Coursera',
 'University Courses',
 'Kaggle Learn Courses',
 'Udemy',
 'DataCamp',
 'LinkedIn Learning',
 'edX',
 'Udacity',
 'Cloud-certification programs',
 'Fast.ai']
```

In [27]:
```
# Create a list of Numeriacl values.
y1 =list(top_course_platform)
y1
```

Out[27]:
```
[9699.0, 6780.0, 6628.0, 6116.0, 3718.0, 2766.0, 2474.0, 2199.0, 1821.0, 944.0]
```

In [28]:
```
# Plot a Horizontal Bar Graph.
plt.figure(figsize=(7,5))
plt.barh(x1,y1,color=['brown','Cyan','magenta','Orange','g','b','y','pink','skyblue','purple'])
plt.title('Course platform v/s Count',fontweight='bold',fontsize=20)
plt.xlabel('Count',fontweight='bold',fontsize=15)
plt.ylabel('Course platform',fontweight='bold',fontsize=15)
plt.show()
```



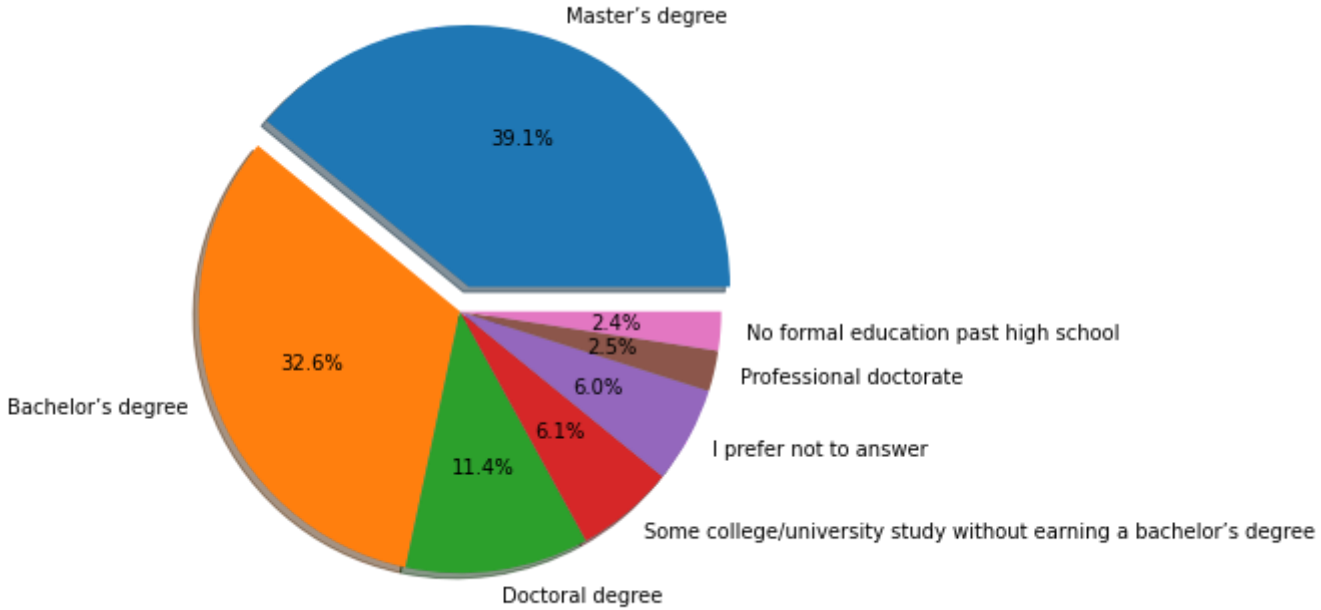
" Coursera " is the top platform where total 9699 people completed their course.

What is the highest level of formal education that you have attained or plan to attain within the next 2 years?

In [29]:
```
# Extract all the educational data with its count.
education =df['Q8'].value_counts()[:-1]
education
```

Out[29]:
```
Master’s degree                                                  9142
Bachelor’s degree                                                7625
Doctoral degree                                                  2657
Some college/university study without earning a bachelor’s degree    1431
I prefer not to answer                                           1394
Professional doctorate                                            585
No formal education past high school                              564
Name: Q8, dtype: int64
```

In [30]:
```
# PLot a Pie Chart
plt.figure(figsize=(6,6))
explode = (0.1, 0, 0, 0, 0, 0, 0)
plt.pie(list(df['Q8'].value_counts()[:7]),labels=list(df['Q8'].value_counts().keys()[:7]), shadow=True, autopct ='%0.1
plt.show()
```



39% people completed their Masters Degree and 32.6% people completed Bachelor's Degree.

What programming languages do you use on a regular basis?

In [31]:
```
# Here we extract total count of Programming Languages mostly used.
pl1 =df['Q12_1'].value_counts()[:1]
pl1
```

Out[31]:
```
Python    18653
Name: Q12_1, dtype: int64
```

In [32]:
```
pl2 =df['Q12_2'].value_counts()[:1]
pl2
```

Out[32]:
```
R    4571
Name: Q12_2, dtype: int64
```

In [33]:
```
pl3 =df['Q12_3'].value_counts()[:1]
pl3
```

Out[33]:
```
SQL    9620
Name: Q12_3, dtype: int64
```

In [34]:
```
pl4 =df['Q12_4'].value_counts()[:1]
pl4
```

Out[34]:
```
C    3801
Name: Q12_4, dtype: int64
```

In [35]:
```
pl5 =df['Q12_5'].value_counts()[:1]
pl5
```

Out[35]:
```
C#    1473
Name: Q12_5, dtype: int64
```

In [36]:
```
pl6 =df['Q12_6'].value_counts()[:1]
pl6
```

Out[36]:
```
C++    4549
Name: Q12_6, dtype: int64
```

In [37]:
```
pl7 =df['Q12_7'].value_counts()[:1]
pl7
```

Out[37]:
```
Java    3862
Name: Q12_7, dtype: int64
```

In [38]:
```
pl8 =df['Q12_8'].value_counts()[:1]
pl8
```

Out[38]:
```
Javascript    3489
Name: Q12_8, dtype: int64
```

In [39]:
```
pl9 =df['Q12_9'].value_counts()[:1]
pl9
```

Out[39]:
```
Bash    1674
Name: Q12_9, dtype: int64
```

In [40]:
```
pl10 =df['Q12_10'].value_counts()[:1]
pl10
```

Out[40]:
```
PHP    1443
Name: Q12_10, dtype: int64
```

In [41]:
```
pl11 =df['Q12_11'].value_counts()[:1]
pl11
```

Out[41]:
```
MATLAB    2441
Name: Q12_11, dtype: int64
```

In [42]:
```
pl12 =df['Q12_12'].value_counts()[:1]
pl12
```

Out[42]:
```
Julia    296
Name: Q12_12, dtype: int64
```

In [43]:
```
pl12 =df['Q12_12'].value_counts()[:1]
pl12
```

Out[43]:
```
Julia    296
Name: Q12_12, dtype: int64
```

In [44]:
```
# create a new data frame with above information.Get the sum of elements of an iterable & then Sort them in desending
programing_language= pd.DataFrame([pl1,pl2,pl3,pl4,pl5,pl6,pl7,pl8,pl9,pl10,pl11,pl12]).sum()[:12].sort_values(ascendi
programing_language
```

Out[44]:
```
Python        18653.0
SQL            9620.0
R              4571.0
C++            4549.0
Java           3862.0
C              3801.0
Javascript     3489.0
MATLAB         2441.0
Bash           1674.0
C#             1473.0
PHP            1443.0
Julia           296.0
dtype: float64
```

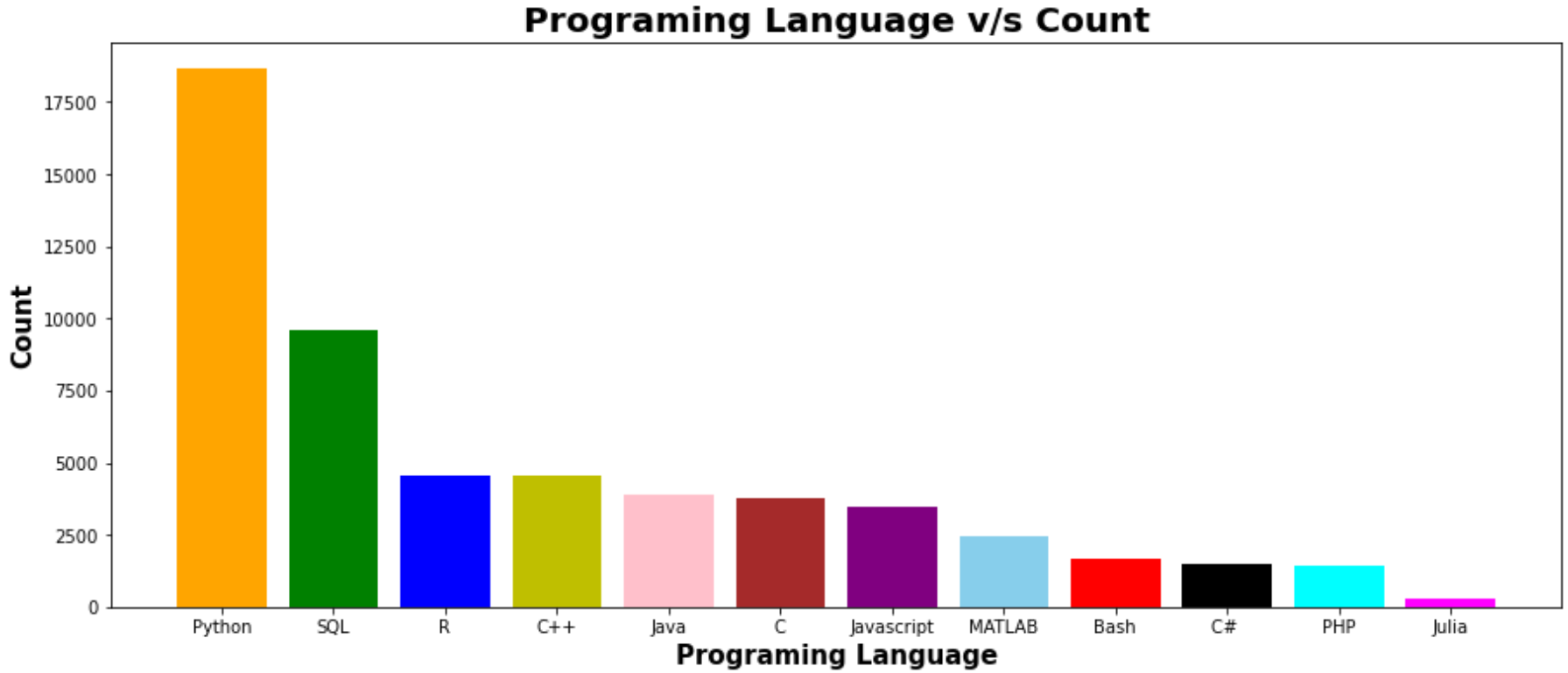
In [45]:
```
# Create a list of Categorical values.
x2=['Python','SQL','R','C++','Java','C','Javascript','MATLAB','Bash','C# ','PHP','Julia']
x2
```

Out[45]:
```
['Python',
 'SQL',
 'R',
 'C++',
 'Java',
 'C',
 'Javascript',
 'MATLAB',
 'Bash',
 'C# ',
 'PHP',
 'Julia']
```

In [46]:
```
# Create a list of Numeriacl values.
y2 = list(programing_language)
y2[:12]
```

Out[46]:
```
[18653.0,
 9620.0,
 4571.0,
 4549.0,
 3862.0,
 3801.0,
 3489.0,
 2441.0,
 1674.0,
 1473.0,
 1443.0,
 296.0]
```

In [47]:
```
# Plot a Bar graph.
plt.figure(figsize=(15,6))
plt.bar(x2,y2,color=['Orange','g','b','y','pink','brown','purple','skyblue','r','black','Cyan','magenta'])
plt.title('Programing Language v/s Count',fontweight='bold',fontsize=20)
plt.xlabel('Programing Language',fontweight='bold',fontsize=15)
plt.ylabel('Count',fontweight='bold',fontsize=15)
plt.show()
```



Python & SQL these two programming languages are widely used.

Which integrated development environments (IDE's) do you use on a regular basis?

In [48]:
```
# Here we extract total count of IDE's.
ide1 = df['Q13_1'].value_counts()[:1]
ide1
```

Out[48]:
```
JupyterLab     4887
Name: Q13_1, dtype: int64
```

In [49]:
```
ide2 = df['Q13_2'].value_counts()[:1]
ide2
```

Out[49]:
```
 RStudio     3824
Name: Q13_2, dtype: int64
```

In [50]:
```
ide3 = df['Q13_3'].value_counts()[:1]
ide3
```

Out[50]:
```
 Visual Studio     4416
Name: Q13_3, dtype: int64
```

In [51]:
```
ide4 = df['Q13_4'].value_counts()[:1]
ide4
```

Out[51]:
```
 Visual Studio Code (VSCode)     9976
Name: Q13_4, dtype: int64
```

In [52]:
```
ide5 = df['Q13_5'].value_counts()[:1]
ide5
```

Out[52]:
```
 PyCharm     6099
Name: Q13_5, dtype: int64
```

In [53]:
```
ide6 = df['Q13_6'].value_counts()[:1]
ide6
```

Out[53]:
```
  Spyder      2880
Name: Q13_6, dtype: int64
```

In [54]:
```
ide7 = df['Q13_7'].value_counts()[:1]
ide7
```

Out[54]:
```
  Notepad++      3891
Name: Q13_7, dtype: int64
```

In [55]:
```
ide8 = df['Q13_8'].value_counts()[:1]
ide8
```

Out[55]:
```
  Sublime Text      2218
Name: Q13_8, dtype: int64
```

In [56]:
```
ide9 = df['Q13_9'].value_counts()[:1]
ide9
```

Out[56]:
```
  Vim / Emacs      1448
Name: Q13_9, dtype: int64
```

In [57]:
```
ide10 = df['Q13_10'].value_counts()[:1]
ide10
```

Out[57]:
```
 MATLAB     2302
Name: Q13_10, dtype: int64
```

In [58]:
```
ide11 = df['Q13_11'].value_counts()[:1]
ide11
```

Out[58]:
```
 Jupyter Notebook    13684
Name: Q13_11, dtype: int64
```

In [59]:
```
# create a new data frame with above information.Get the unique values with count & then Sort them in desending order.
IDE = pd.DataFrame([ide1,ide2,ide3,ide4,ide5,ide6,ide7,
                    ide8,ide10,ide11]).sum().sort_values(ascending=False)[:10]
IDE
```

Out[59]:
```
 Jupyter Notebook               13684.0
 Visual Studio Code (VSCode)     9976.0
 PyCharm                         6099.0
JupyterLab                       4887.0
 Visual Studio                   4416.0
  Notepad++                      3891.0
 RStudio                         3824.0
  Spyder                         2880.0
 MATLAB                          2302.0
  Sublime Text                   2218.0
dtype: float64
```

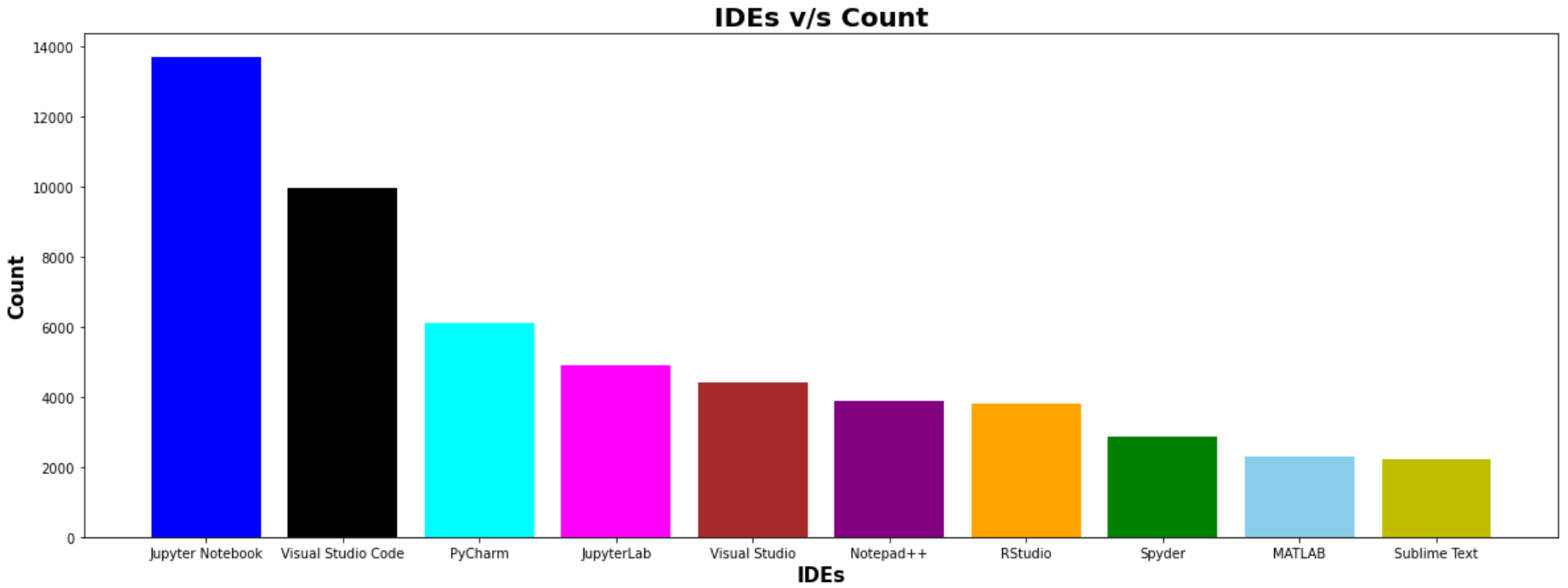
In [60]:
```
# Create a list of Categorical values.
x3 = ['Jupyter Notebook','Visual Studio Code','PyCharm','JupyterLab',
      'Visual Studio','Notepad++','RStudio','Spyder','MATLAB','Sublime Text']
x3
```

Out[60]:
```
['Jupyter Notebook',
 'Visual Studio Code',
 'PyCharm',
 'JupyterLab',
 'Visual Studio',
 'Notepad++',
 'RStudio',
 'Spyder',
 'MATLAB',
 'Sublime Text']
```

In [61]:
```
# Create a list of Numeriacl values.
y3 =list(IDE)
y3
```

Out[61]:
```
[13684.0,
 9976.0,
 6099.0,
 4887.0,
 4416.0,
 3891.0,
 3824.0,
 2880.0,
 2302.0,
 2218.0]
```

In [62]:
```
# Plot a Bar graph
plt.figure(figsize=(20,7))
plt.bar(x3, y3,color=['blue','black','Cyan','magenta','brown','purple','Orange','g','skyblue','y',])
plt.title('IDEs v/s Count',fontweight='bold',fontsize=20)
plt.xlabel('IDEs',fontweight='bold',fontsize=15)
plt.ylabel('Count',fontweight='bold',fontsize=15)
plt.show()
```



Jupyter Notebook & Visual Studio Code these 2 IDE's are widely used.

Which library do you use for data visualization on a regular basis ?

In [63]:
```
# Here we extract total count of different Libraries used for data Visualization.
lb1 =df['Q15_1'].value_counts()[:1]
lb1
```

Out[63]:
```
 Matplotlib     14010
Name: Q15_1, dtype: int64
```

In [64]:
```
lb2 =df['Q15_2'].value_counts()[:1]
lb2
```

Out[64]:
```
 Seaborn     10512
Name: Q15_2, dtype: int64
```

In [65]:
```
lb3 =df['Q15_3'].value_counts()[:1]
lb3
```

Out[65]:
```
 Plotly / Plotly Express     5078
Name: Q15_3, dtype: int64
```

In [66]:
```
lb4 =df['Q15_4'].value_counts()[:1]
lb4
```

Out[66]:
```
 Ggplot / ggplot2     4145
Name: Q15_4, dtype: int64
```

In [67]:
```
lb5 =df['Q15_5'].value_counts()[:1]
lb5
```

Out[67]:
```
 Shiny     1043
Name: Q15_5, dtype: int64
```

In [68]:
```
lb6 =df['Q15_6'].value_counts()[:1]
lb6
```

Out[68]:
```
 D3 js     734
Name: Q15_6, dtype: int64
```

In [69]:
```
lb7 =df['Q15_7'].value_counts()[:1]
lb7
```

Out[69]:
```
 Altair     300
Name: Q15_7, dtype: int64
```

In [70]:
```
lb8 =df['Q15_8'].value_counts()[:1]
lb8
```

Out[70]:
```
 Bokeh     771
Name: Q15_8, dtype: int64
```

In [71]:
```
lb9 =df['Q15_9'].value_counts()[:1]
lb9
```

Out[71]:
```
 Geoplotlib     1167
Name: Q15_9, dtype: int64
```

In [72]:
```
lb10 =df['Q15_10'].value_counts()[:1]
lb10
```

Out[72]:
```
 Leaflet / Folium     554
Name: Q15_10, dtype: int64
```

In [73]:
```
# create a new data frame with above information.Get the sum of elements of an iterable & then Sort them in desending
most_used_library= pd.DataFrame([lb1,lb2,lb3,lb4,lb5,lb6,lb7,lb8,lb9,lb10]).sum().sort_values(ascending=False)
most_used_library
```

Out[73]:
```
Matplotlib                 14010.0
Seaborn                    10512.0
Plotly / Plotly Express     5078.0
Ggplot / ggplot2           4145.0
Geoplotlib                 1167.0
Shiny                      1043.0
Bokeh                       771.0
D3 js                       734.0
Leaflet / Folium            554.0
Altair                      300.0
dtype: float64
```

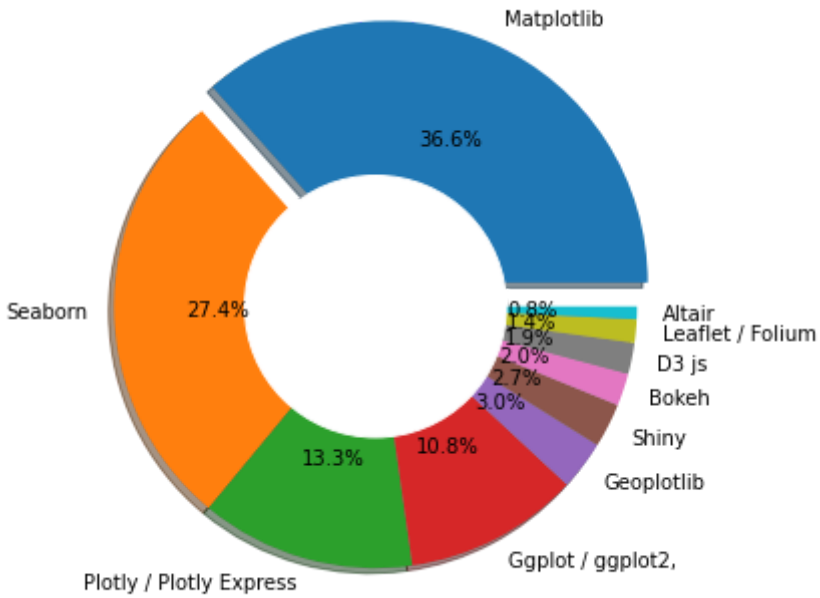
In [74]:
```
# Create a list of Numeriacl values.
x4 = list(most_used_library)
x4
```

Out[74]:
```
[14010.0, 10512.0, 5078.0, 4145.0, 1167.0, 1043.0, 771.0, 734.0, 554.0, 300.0]
```

In [75]:
```
# Create a list of Categorical values.
y4 = ['Matplotlib','Seaborn','Plotly / Plotly Express','Ggplot / ggplot2,','Geoplotlib','Shiny','Bokeh','D3 js','Leafl
y4
```

Out[75]:
```
['Matplotlib',
 'Seaborn',
 'Plotly / Plotly Express',
 'Ggplot / ggplot2,',
 'Geoplotlib',
 'Shiny',
 'Bokeh',
 'D3 js',
 'Leaflet / Folium',
 'Altair']
```

In [76]:
```
# Plot a Donut Chart.
plt.figure(figsize=(6,6))
explode = (0.1, 0, 0, 0, 0, 0, 0, 0, 0, 0)
plt.pie(x4,labels=y4, shadow=True, autopct ='%0.1f%%', explode=explode)
circle = plt.Circle( (0,0), 0.5, color='white')
p=plt.gcf()
p.gca().add_artist(circle)
plt.show()
```



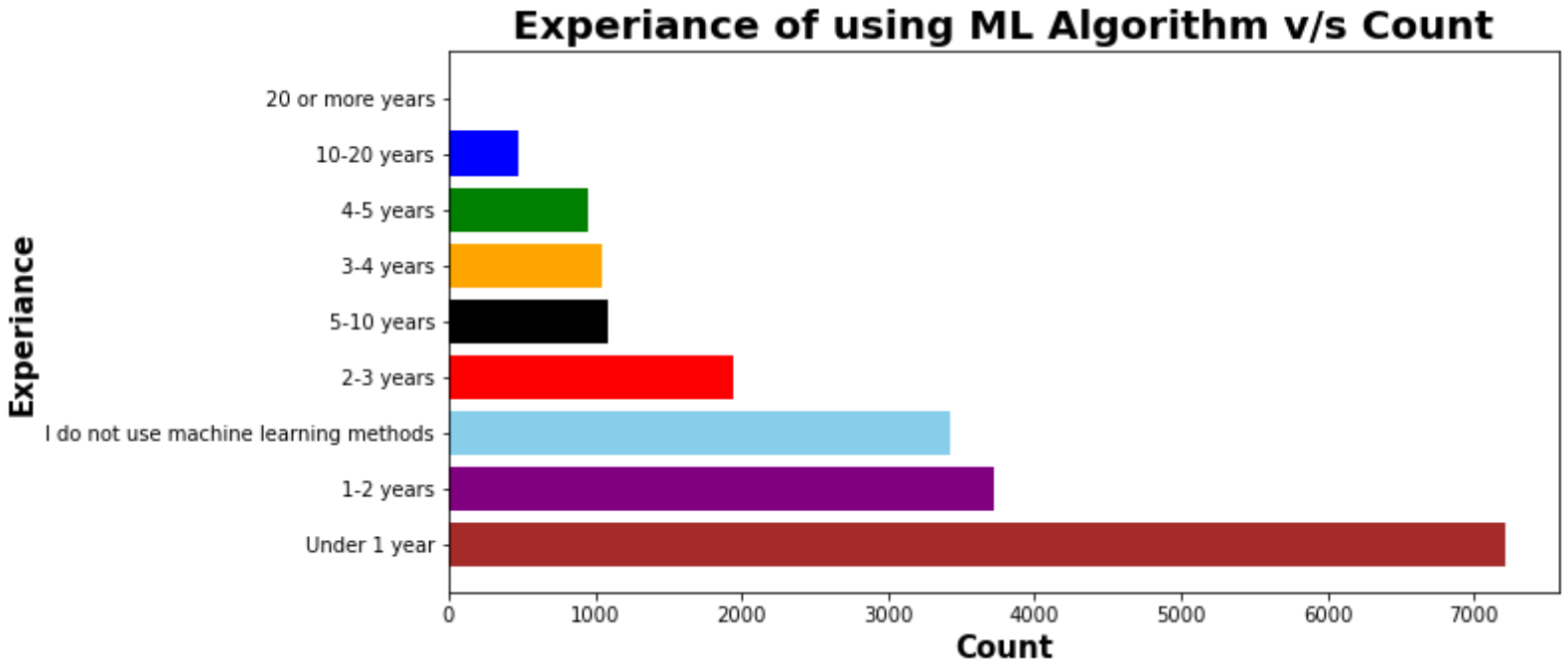
Matplotlib & Seaborn these two libraries are widely used for data visualization.

For how many years have you used machine learning methods?

In [77]:
```
# Here we extract total count of years candidates have used machine learning methods.
ml_experiance = df['Q16'].value_counts()[:-1]
ml_experiance
```

Out[77]:
```
Under 1 year                          7221
1-2 years                             3720
I do not use machine learning methods 3419
2-3 years                             1947
5-10 years                            1090
3-4 years                             1053
4-5 years                              950
10-20 years                            483
20 or more years                         3
Name: Q16, dtype: int64
```

In [78]:
```
# Plot a horizontal bar graph.
plt.figure(figsize=(10,5))
plt.barh(list(ml_experiance.keys()),list(ml_experiance),color=['brown','purple','skyblue','r','black','orange','g','b'
plt.title('Experiance of using ML Algorithm v/s Count',fontweight='bold',fontsize=20)
plt.xlabel('Count',fontweight='bold',fontsize=15)
plt.ylabel('Experiance',fontweight='bold',fontsize=15)
plt.show()
```



The candidates with less than 1 year experiance are more enthusiastic to use ML algorithm.

Which of the following machine learning frameworks do you use on a regular basis?

In [79]:
```
# Here we extract total count of machine learning frameworks used on a regular basis.
fr1 = df['Q17_1'].value_counts()[:1]
fr1
```

Out[79]:
```
  Scikit-learn     11403
Name: Q17_1, dtype: int64
```

In [80]:
```
fr2 = df['Q17_2'].value_counts()[:1]
fr2
```

Out[80]:
```
  TensorFlow     7953
Name: Q17_2, dtype: int64
```

In [81]:
```
fr3 = df['Q17_3'].value_counts()[:1]
fr3
```

Out[81]:
```
 Keras     6575
Name: Q17_3, dtype: int64
```

In [82]:
```
fr4 = df['Q17_4'].value_counts()[:1]
fr4
```

Out[82]:
```
 PyTorch     5191
Name: Q17_4, dtype: int64
```

In [83]:
```
fr6 = df['Q17_6'].value_counts()[:1]
fr6
```

Out[83]:
```
 Xgboost     4477
Name: Q17_6, dtype: int64
```

In [84]:
```
fr7 = df['Q17_7'].value_counts()[:1]
fr7
```

Out[84]:
```
 LightGBM     1940
Name: Q17_7, dtype: int64
```

In [85]:
```
fr8 = df['Q17_8'].value_counts()[:1]
fr8
```

Out[85]:
```
 CatBoost     1165
Name: Q17_8, dtype: int64
```

In [86]:
```
fr12 = df['Q17_12'].value_counts()[:1]
fr12
```

Out[86]:
```
 PyTorch Lightning     1013
Name: Q17_12, dtype: int64
```

In [87]:
```
fr13 = df['Q17_13'].value_counts()[:1]
fr13
```

Out[87]:
```
 Huggingface     1332
Name: Q17_13, dtype: int64
```

In [88]:
```
fr14 = df['Q17_14'].value_counts()[:1]
fr14
```

Out[88]:
```
None    1709
Name: Q17_14, dtype: int64
```

In [89]:
```
# create a new data frame with above information.Get the sum of elements of an iterable & then Sort them in desending
framework= pd.DataFrame([fr1,fr2,fr3,fr4,fr6,fr7,fr8,fr12,fr13,fr14]).sum().sort_values(ascending=False)
framework
```

Out[89]:
```
  Scikit-learn         11403.0
  TensorFlow            7953.0
 Keras                  6575.0
 PyTorch                5191.0
 Xgboost                4477.0
 LightGBM               1940.0
None                    1709.0
 Huggingface            1332.0
 CatBoost               1165.0
 PyTorch Lightning      1013.0
dtype: float64
```

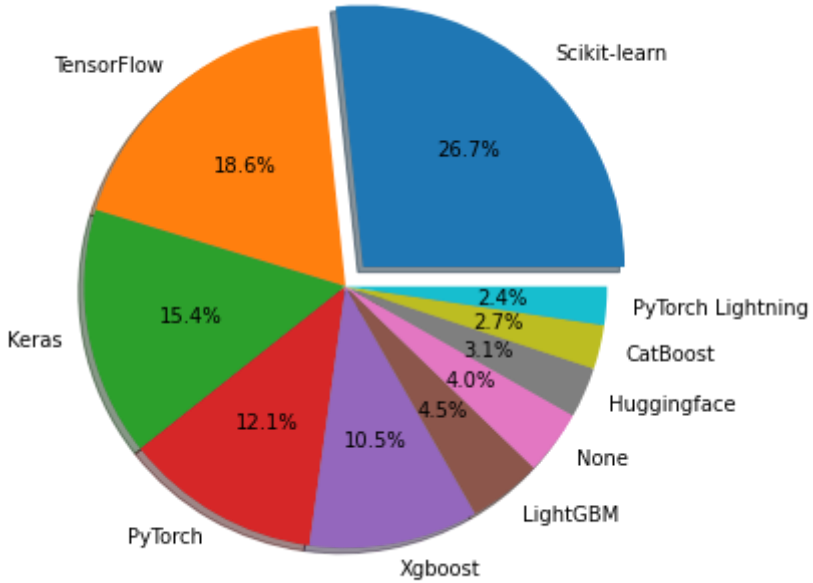
In [90]:
```
# Create a list of Numeriacl values.
x5 = list(framework)
x5
```

Out[90]:
```
[11403.0,
 7953.0,
 6575.0,
 5191.0,
 4477.0,
 1940.0,
 1709.0,
 1332.0,
 1165.0,
 1013.0]
```

In [91]:
```
# Create a list of Categorical values.
y5 = ['Scikit-learn','TensorFlow','Keras','PyTorch','Xgboost','LightGBM','None','Huggingface','CatBoost','PyTorch Ligh
y5
```

Out[91]:
```
['Scikit-learn',
 'TensorFlow',
 'Keras',
 'PyTorch',
 'Xgboost',
 'LightGBM',
 'None',
 'Huggingface',
 'CatBoost',
 'PyTorch Lightning']
```

In [92]:
```
# Plot a Pie Chart
plt.figure(figsize=(6,6))
explode = (0.1, 0, 0, 0, 0, 0, 0, 0, 0, 0)
plt.pie(x5,labels=y5, shadow=True, autopct ='%0.1f%%', explode=explode)
plt.show()
```



Scikit-learn is dominantly used by data scientists.

Which of the following ML algorithms do you use on a regular basis?

In [93]:
```
# Here we extract total count of ML algorithms regularly used .
algo1 = df['Q18_1'].value_counts()[:1]
algo1
```

Out[93]:
```
Linear or Logistic Regression    11338
Name: Q18_1, dtype: int64
```

In [94]:
```
algo2 = df['Q18_2'].value_counts()[:1]
algo2
```

Out[94]:
```
Decision Trees or Random Forests    9373
Name: Q18_2, dtype: int64
```

In [95]:
```
algo3 = df['Q18_3'].value_counts()[:1]
algo3
```

Out[95]:
```
Gradient Boosting Machines (xgboost, lightgbm, etc)    5506
Name: Q18_3, dtype: int64
```

In [96]:
```
algo4 = df['Q18_4'].value_counts()[:1]
algo4
```

Out[96]:
```
Bayesian Approaches    3661
Name: Q18_4, dtype: int64
```

In [97]:
```
algo5 = df['Q18_5'].value_counts()[:1]
algo5
```

Out[97]:
```
Evolutionary Approaches    823
Name: Q18_5, dtype: int64
```

In [98]:
```
algo6 = df['Q18_6'].value_counts()[:1]
algo6
```

Out[98]:
```
Dense Neural Networks (MLPs, etc)    3476
Name: Q18_6, dtype: int64
```

In [99]:
```
algo7 = df['Q18_7'].value_counts()[:1]
algo7
```

Out[99]:
```
Convolutional Neural Networks    6006
Name: Q18_7, dtype: int64
```

In [100]:
```
algo8 = df['Q18_8'].value_counts()[:1]
algo8
```

Out[100]:
```
Generative Adversarial Networks    1166
Name: Q18_8, dtype: int64
```

In [101]:
```
algo9 = df['Q18_9'].value_counts()[:1]
algo9
```

Out[101]:
```
Recurrent Neural Networks    3451
Name: Q18_9, dtype: int64
```

In [102]:
```
algo10 = df['Q18_10'].value_counts()[:1]
algo10
```

Out[102]:
```
Transformer Networks (BERT, gpt-3, etc)    2196
Name: Q18_10, dtype: int64
```

In [103]:
```
algo11 = df['Q18_11'].value_counts()[:1]
algo11
```

Out[103]:
```
Autoencoder Networks (DAE, VAE, etc)    1234
Name: Q18_11, dtype: int64
```

In [104]:
```
algo12 = df['Q18_12'].value_counts()[:1]
algo12
```

Out[104]:
```
Graph Neural Networks    1422
Name: Q18_12, dtype: int64
```

In [105]:
```
# create a new data frame with above information.Get the sum of elements of an iterable & then Sort them in desending
ML_algo = pd.DataFrame([algo1,algo2,algo3,algo4,algo5,algo6,algo7,algo8,algo9,algo10,algo11,algo12]).sum().sort_values
ML_algo
```

Out[105]:
```
Linear or Logistic Regression                        11338.0
Decision Trees or Random Forests                      9373.0
Convolutional Neural Networks                         6006.0
Gradient Boosting Machines (xgboost, lightgbm, etc)   5506.0
Bayesian Approaches                                   3661.0
Dense Neural Networks (MLPs, etc)                     3476.0
Recurrent Neural Networks                             3451.0
Transformer Networks (BERT, gpt-3, etc)               2196.0
Graph Neural Networks                                 1422.0
Autoencoder Networks (DAE, VAE, etc)                  1234.0
dtype: float64
```

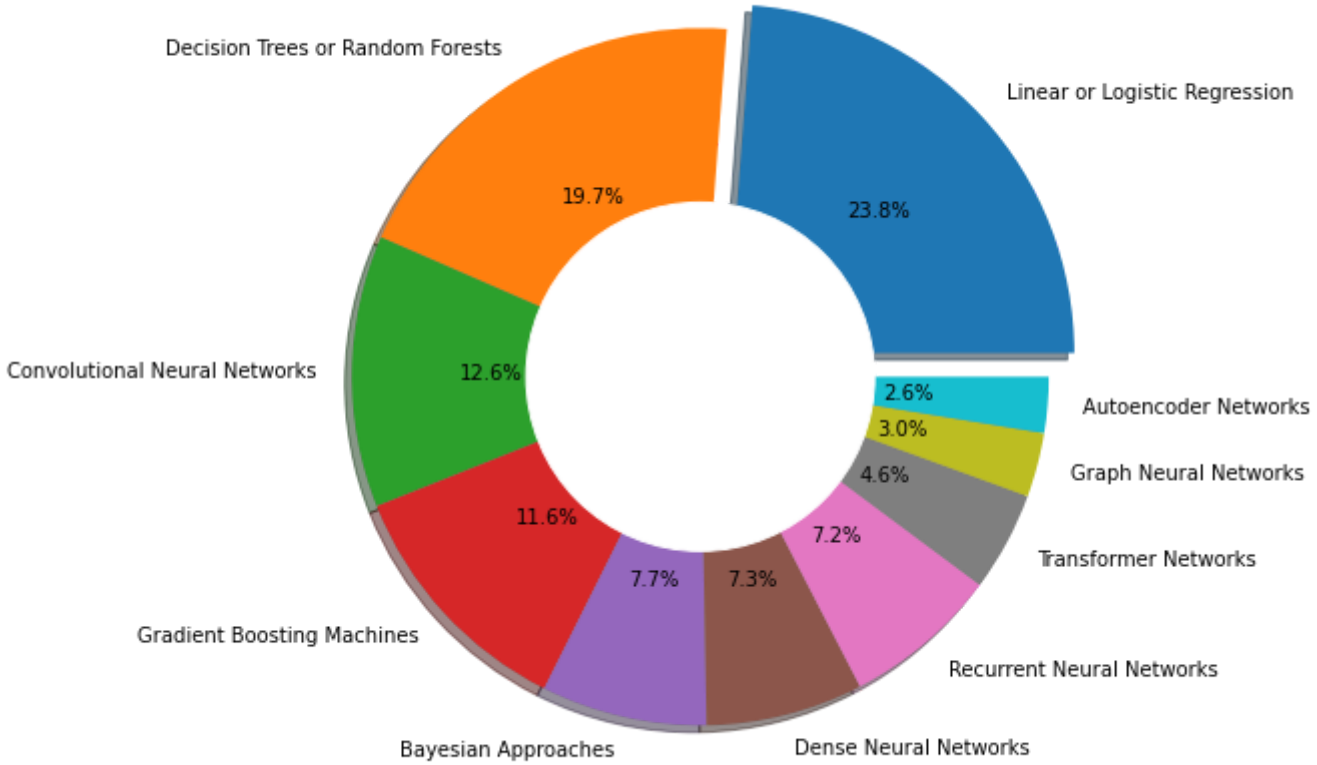
In [106]:
```
# Create a list of Numeriacl values.
x =list(ML_algo)
x
```

Out[106]:
```
[11338.0,
 9373.0,
 6006.0,
 5506.0,
 3661.0,
 3476.0,
 3451.0,
 2196.0,
 1422.0,
 1234.0]
```

In [107]:
```
# Create a list of Categorical values.
y = ['Linear or Logistic Regression','Decision Trees or Random Forests','Convolutional Neural Networks','Gradient Boos
     'Bayesian Approaches','Dense Neural Networks','Recurrent Neural Networks','Transformer Networks',
     'Graph Neural Networks','Autoencoder Networks']
y
```

Out[107]:
```
['Linear or Logistic Regression',
 'Decision Trees or Random Forests',
 'Convolutional Neural Networks',
 'Gradient Boosting Machines',
 'Bayesian Approaches',
 'Dense Neural Networks',
 'Recurrent Neural Networks',
 'Transformer Networks',
 'Graph Neural Networks',
 'Autoencoder Networks']
```

In [108]:
```
# Plot a Donut Chart.
plt.figure(figsize=(8,8))
explode = (0.1, 0, 0, 0, 0, 0, 0, 0, 0, 0)
plt.pie(x,labels=y, shadow=True, autopct ='%0.1f%%', explode=explode)
circle = plt.Circle( (0,0), 0.5, color='white')
p=plt.gcf()
p.gca().add_artist(circle)
plt.show()
```



Linear or Logistic Regression is widely used with almost 24% votes.

Which of the following ML model hubs/repositories do you use most often?

In [109]:
```
# Here we extract total count of ML model hubs/repositories most oftenly used.
ml_hub = df['Q22'].value_counts()[:-1]
ml_hub
```

Out[109]:
```
 Kaggle datasets                                1618
  TensorFlow Hub                                 799
 Huggingface Models                              502
 PyTorch Hub                                     412
 Timm                                            127
Other storage services (i.e. google drive)        96
 NVIDIA NGC models                                69
 ONNX models                                      46
 Jumpstart                                        24
Name: Q22, dtype: int64
```

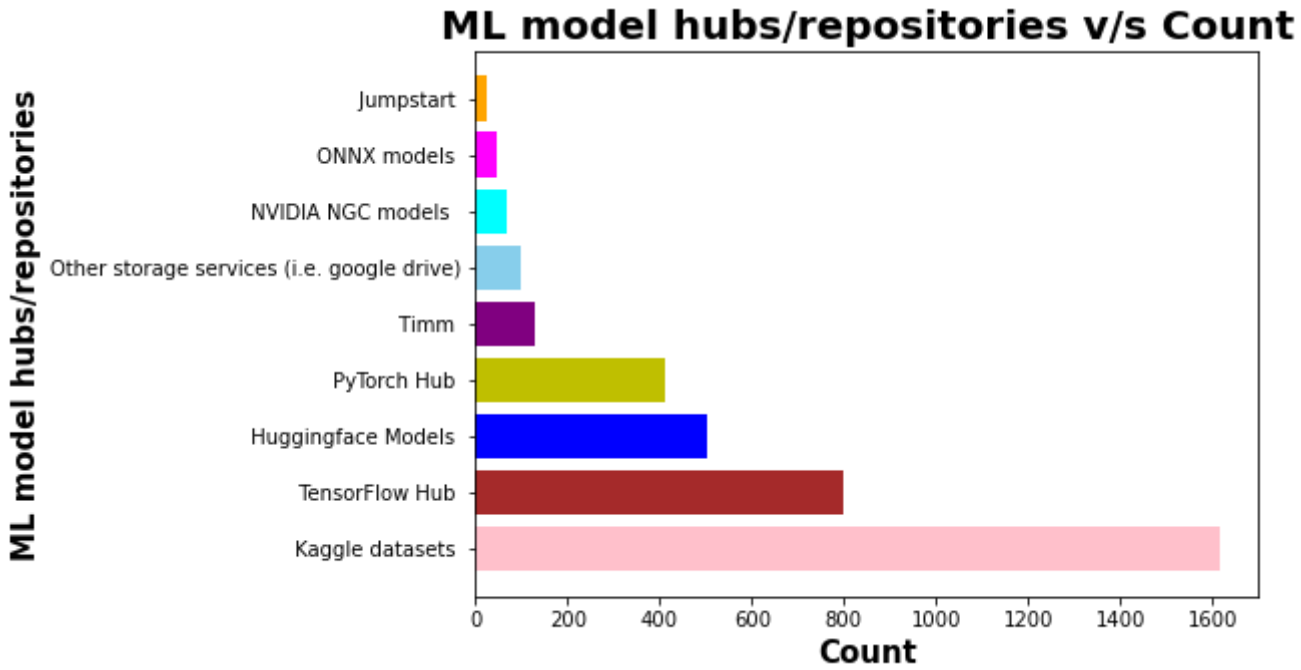
In [110]:
```
# Create a list of Categorical values.
x6 = ml_hub.keys()
x6
```

Out[110]:
```
Index([' Kaggle datasets ', '  TensorFlow Hub ', ' Huggingface Models ',
       ' PyTorch Hub ', ' Timm ', 'Other storage services (i.e. google drive)',
       ' NVIDIA NGC models  ', ' ONNX models ', ' Jumpstart '],
      dtype='object')
```

In [111]:
```
# Create a list of Numerical values.
y6 = list(ml_hub)
y6
```

Out[111]:
```
[1618, 799, 502, 412, 127, 96, 69, 46, 24]
```

In [112]:
```
# Plot a horizontal Bar graph
plt.figure(figsize=(7,5))
plt.barh(x6,y6,color=['pink','brown','b','y','purple','skyblue','Cyan','magenta','Orange','g'])
plt.title('ML model hubs/repositories v/s Count',fontweight='bold',fontsize=20)
plt.xlabel('Count',fontweight='bold',fontsize=15)
plt.ylabel('ML model hubs/repositories',fontweight='bold',fontsize=15)
plt.show()
```



Mostly Kaggle datasets are used for ML model hubs or repositories.

Select the title most similar to your current role?

In [113]:
```
# Here we extract total count of the title most similar to current role.
df['Q23'].value_counts()[:-1]
```

Out[113]:
```
Data Scientist                                                  1929
Data Analyst (Business, Marketing, Financial, Quantitative, etc)    1538
Currently not employed                                          1432
Software Engineer                                                980
Teacher / professor                                              833
Manager (Program, Project, Operations, Executive-level, etc)     832
Other                                                            754
Research Scientist                                               593
Machine Learning/ MLops Engineer                                 571
Engineer (non-software)                                          465
Data Engineer                                                    352
Statistician                                                     125
Data Architect                                                    95
Data Administrator                                                70
Developer Advocate                                                61
Name: Q23, dtype: int64
```

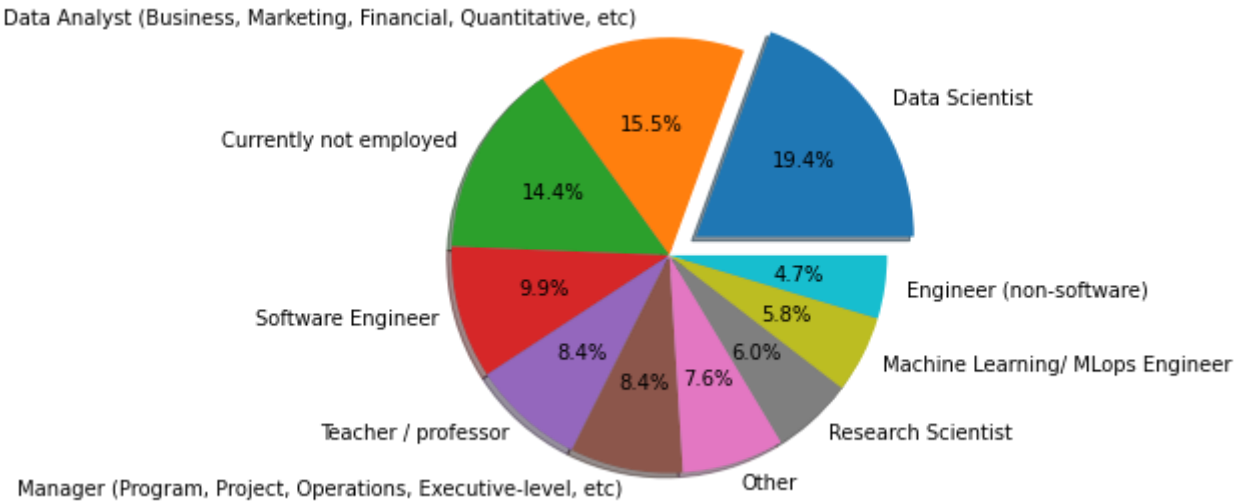
In [114]:
```
# Create a list of Numerical values.
x7 = list(df['Q23'].value_counts())[:10]
x7
```

Out[114]:
```
[1929, 1538, 1432, 980, 833, 832, 754, 593, 571, 465]
```

In [115]:
```
# Create a list of Categorical values.
y7 = list(df['Q23'].value_counts().keys())[:10]
y7
```

Out[115]:
```
['Data Scientist',
 'Data Analyst (Business, Marketing, Financial, Quantitative, etc)',
 'Currently not employed',
 'Software Engineer',
 'Teacher / professor',
 'Manager (Program, Project, Operations, Executive-level, etc)',
 'Other',
 'Research Scientist',
 'Machine Learning/ MLops Engineer',
 'Engineer (non-software)']
```

In [116]:
```
# Plot a Pie Chart
plt.figure(figsize=(5,5))
explode = (0.15, 0, 0, 0, 0, 0, 0, 0, 0, 0)
plt.pie(x7,labels=y7, shadow=True, autopct ='%0.1f%%', explode=explode)
plt.show()
```



' Data Scientist ' is the title most similar to current role for almost 20% people.

In what industry you are currently working?

In [117]:
```
# Here we extract total count of industry in which people are currently working.
df['Q24'].value_counts()[:15]
```

Out[117]:
```
Computers/Technology                          2321
Academics/Education                           1447
Accounting/Finance                             802
Other                                          750
Manufacturing/Fabrication                      561
Medical/Pharmaceutical                         509
Government/Public Service                      500
Online Service/Internet-based Services         461
Retail/Sales                                   398
Energy/Mining                                  320
Insurance/Risk Assessment                      256
Marketing/CRM                                  246
Non-profit/Service                             194
Broadcasting/Communications                    179
Shipping/Transportation                        150
Name: Q24, dtype: int64
```

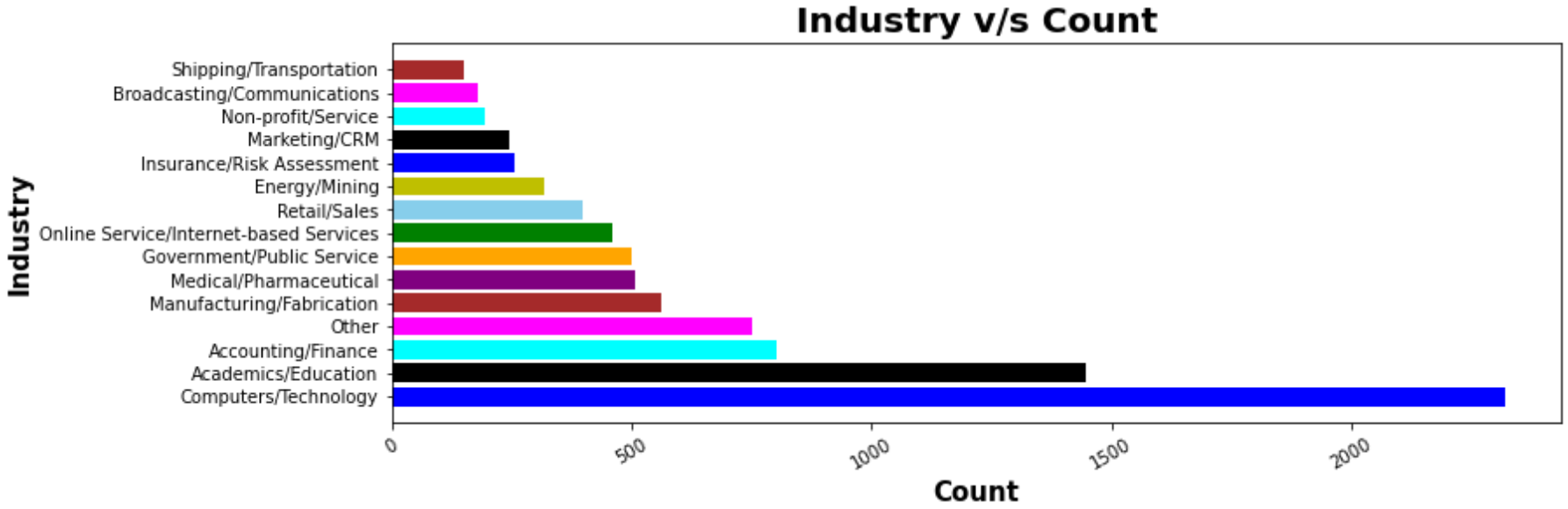
In [118]:
```
# Create a list of Numerical values.
y8 = list(df['Q24'].value_counts())[:15]
y8
```

Out[118]:
```
[2321, 1447, 802, 750, 561, 509, 500, 461, 398, 320, 256, 246, 194, 179, 150]
```

In [119]:
```
# Create a list of Categorical values.
x8 = list(df['Q24'].value_counts().keys())[:15]
x8
```

Out[119]:
```
['Computers/Technology',
 'Academics/Education',
 'Accounting/Finance',
 'Other',
 'Manufacturing/Fabrication',
 'Medical/Pharmaceutical',
 'Government/Public Service',
 'Online Service/Internet-based Services',
 'Retail/Sales',
 'Energy/Mining',
 'Insurance/Risk Assessment',
 'Marketing/CRM',
 'Non-profit/Service',
 'Broadcasting/Communications',
 'Shipping/Transportation']
```

In [120]:
```
# Plot a horizontal  Bar graph
plt.figure(figsize=(12,4))
plt.barh(x8, y8,color=['blue','black','Cyan','magenta','brown','purple','Orange','g','skyblue','y',])
plt.xticks(rotation=30, horizontalalignment="center")
plt.title('Industry v/s Count',fontweight='bold',fontsize=20)
plt.xlabel('Count',fontweight='bold',fontsize=15)
plt.ylabel('Industry',fontweight='bold',fontsize=15)
plt.show()
```



The people who are working in 'computers / Technology ' are mostly enthusiastic to work as Data Scientist.

Which of the following cloud computing platforms do you use?

In [121]:
```
# Here we extract total count of cloud computing platforms used.
cc1 = df['Q31_1'].value_counts()[:1]
cc1
```

Out[121]:
```
 Amazon Web Services (AWS)     2346
Name: Q31_1, dtype: int64
```

In [122]:
```
cc2 = df['Q31_2'].value_counts()[:1]
cc2
```

Out[122]:
```
 Microsoft Azure     1416
Name: Q31_2, dtype: int64
```

In [123]:
```
cc3 = df['Q31_3'].value_counts()[:1]
cc3
```

Out[123]:
```
 Google Cloud Platform (GCP)     2056
Name: Q31_3, dtype: int64
```

In [124]:
```
cc4 = df['Q31_4'].value_counts()[:1]
cc4
```

Out[124]:
```
 IBM Cloud / Red Hat     287
Name: Q31_4, dtype: int64
```

In [125]:
```
cc5 = df['Q31_5'].value_counts()[:1]
cc5
```

Out[125]:
```
 Oracle Cloud     230
Name: Q31_5, dtype: int64
```

In [126]:
```
cc6 = df['Q31_6'].value_counts()[:1]
cc6
```

Out[126]:
```
 SAP Cloud     107
Name: Q31_6, dtype: int64
```

In [127]:
```
cc7 = df['Q31_7'].value_counts()[:1]
cc7
```

Out[127]:
```
 VMware Cloud     155
Name: Q31_7, dtype: int64
```

In [128]:
```
cc8 = df['Q31_8'].value_counts()[:1]
cc8
```

Out[128]:
```
 Alibaba Cloud     76
Name: Q31_8, dtype: int64
```

In [129]:
```
cc9 = df['Q31_9'].value_counts()[:1]
cc9
```

Out[129]:
```
 Tencent Cloud     56
Name: Q31_9, dtype: int64
```

In [130]:
```
cc10 = df['Q31_10'].value_counts()[:1]
cc10
```

Out[130]:
```
 Huawei Cloud     47
Name: Q31_10, dtype: int64
```

In [131]:
```
# create a new data frame with above information.Get the sum of elements of an iterable & then Sort them in desending
cloud_computing_platform = pd.DataFrame([cc1,cc2,cc3,cc4,cc5,cc6,cc7,cc8,cc9,cc10]).sum().sort_values(ascending=False)
cloud_computing_platform
```

Out[131]:
```
 Amazon Web Services (AWS)       2346.0
 Google Cloud Platform (GCP)     2056.0
 Microsoft Azure                 1416.0
 IBM Cloud / Red Hat              287.0
 Oracle Cloud                     230.0
 VMware Cloud                     155.0
 SAP Cloud                        107.0
 Alibaba Cloud                     76.0
 Tencent Cloud                     56.0
 Huawei Cloud                      47.0
dtype: float64
```

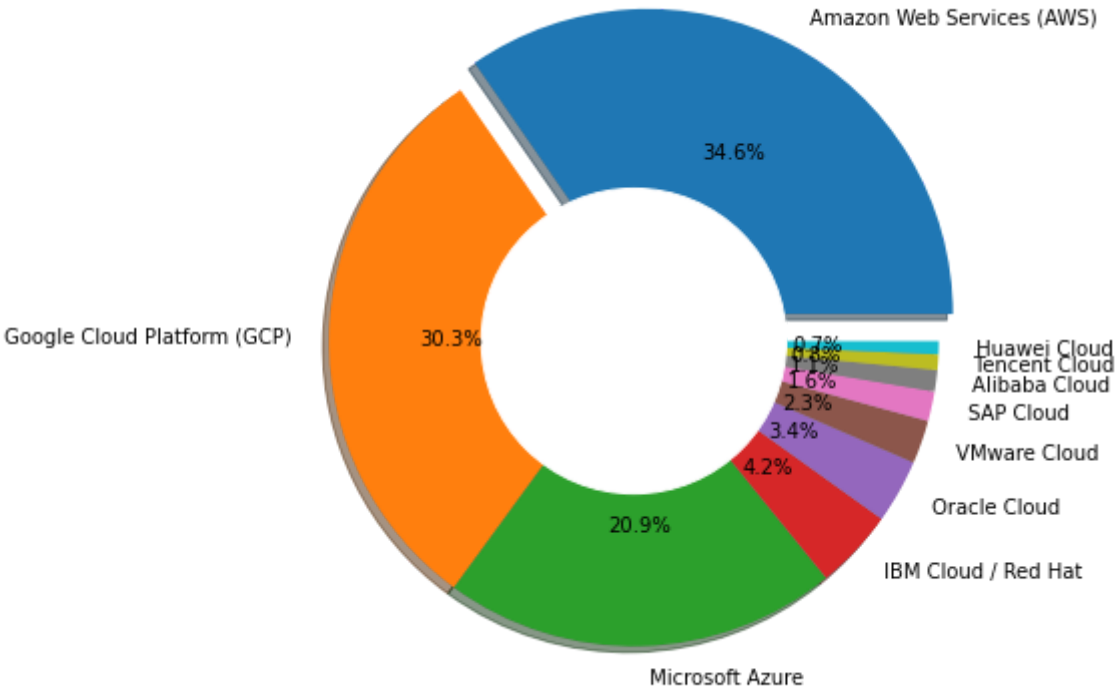
In [132]:
```
# Create a list of Numerical values.
x8 = list(cloud_computing_platform)
x8
```

Out[132]:
```
[2346.0, 2056.0, 1416.0, 287.0, 230.0, 155.0, 107.0, 76.0, 56.0, 47.0]
```

In [133]:
```
# Create a list of Categorical values.
y8 = [' Amazon Web Services (AWS) ',' Google Cloud Platform (GCP) ', ' Microsoft Azure ',
        ' IBM Cloud / Red Hat ',' Oracle Cloud ',' VMware Cloud ',' SAP Cloud ', ' Alibaba Cloud ',
      ' Tencent Cloud ', ' Huawei Cloud ']
y8
```

Out[133]:
```
[' Amazon Web Services (AWS) ',
 ' Google Cloud Platform (GCP) ',
 ' Microsoft Azure ',
 ' IBM Cloud / Red Hat ',
 ' Oracle Cloud ',
 ' VMware Cloud ',
 ' SAP Cloud ',
 ' Alibaba Cloud ',
 ' Tencent Cloud ',
 ' Huawei Cloud ']
```

In [134]:
```
# Plot a Pie Chart
plt.figure(figsize=(7,7))
explode = (0.1, 0, 0, 0, 0, 0, 0, 0, 0, 0)
plt.pie(x8,labels=y8, shadow=True, autopct ='%0.1f%%', explode=explode)
circle = plt.Circle( (0,0), 0.5, color='white')
p=plt.gcf()
p.gca().add_artist(circle)
plt.show()
```



' AWS ' & ' GCP' these cloud computing platforms are significantly used

Do you use any of the following data products (relational databases, data warehouses, data lakes, or similar)?

In [135]:
```
# Here we extract total count of data products like relational databases.
db1 = df['Q35_1'].value_counts()[:1]
db1
```

Out[135]:
```
MySQL     2233
Name: Q35_1, dtype: int64
```

In [136]:
```
db2 = df['Q35_2'].value_counts()[:1]
db2
```

Out[136]:
```
PostgreSQL     1516
Name: Q35_2, dtype: int64
```

In [137]:
```
db3 = df['Q35_3'].value_counts()[:1]
db3
```

Out[137]:
```
SQLite     1159
Name: Q35_3, dtype: int64
```

In [138]:
```
db4 = df['Q35_4'].value_counts()[:1]
db4
```

Out[138]:
```
Oracle Database     688
Name: Q35_4, dtype: int64
```

In [139]:
```
db5 = df['Q35_5'].value_counts()[:1]
db5
```

Out[139]:
```
MongoDB     1031
Name: Q35_5, dtype: int64
```

In [140]:
```
db6 = df['Q35_6'].value_counts()[:1]
db6
```

Out[140]:
```
Snowflake     399
Name: Q35_6, dtype: int64
```

In [141]:
```
db7 = df['Q35_7'].value_counts()[:1]
db7
```

Out[141]:
```
IBM Db2     192
Name: Q35_7, dtype: int64
```

In [142]:
```
db8 = df['Q35_8'].value_counts()[:1]
db8
```

Out[142]:
```
Microsoft SQL Server     1203
Name: Q35_8, dtype: int64
```

In [143]:
```
db9 = df['Q35_9'].value_counts()[:1]
db9
```

Out[143]:
```
Microsoft Azure SQL Database     520
Name: Q35_9, dtype: int64
```

In [144]:
```
db10 = df['Q35_10'].value_counts()[:1]
db10
```

Out[144]:
```
Amazon Redshift     380
Name: Q35_10, dtype: int64
```

In [145]:
```
db11 = df['Q35_11'].value_counts()[:1]
db11
```

Out[145]:
```
Amazon RDS     505
Name: Q35_11, dtype: int64
```

In [146]:
```
db12 = df['Q35_12'].value_counts()[:1]
db12
```

Out[146]:
```
Amazon DynamoDB     356
Name: Q35_12, dtype: int64
```

In [147]:
```
db13 = df['Q35_13'].value_counts()[:1]
db13
```

Out[147]:
```
Google Cloud BigQuery     690
Name: Q35_13, dtype: int64
```

In [148]:
```
db14 = df['Q35_14'].value_counts()[:1]
db14
```

Out[148]:
```
Google Cloud SQL     439
Name: Q35_14, dtype: int64
```

In [149]:
```
# create a new data frame with above information.Get the unique values with count & then Sort them in desending order.
data_base = pd.DataFrame([db1,db2,db3,db4,db5,db6,db7,db8,db9,db10,db11,db12,db13,db14]).sum().sort_values(ascending=F
data_base
```

Out[149]:
```
MySQL                          2233.0
PostgreSQL                     1516.0
Microsoft SQL Server           1203.0
SQLite                         1159.0
MongoDB                        1031.0
Google Cloud BigQuery           690.0
Oracle Database                 688.0
Microsoft Azure SQL Database    520.0
Amazon RDS                      505.0
Google Cloud SQL                439.0
dtype: float64
```

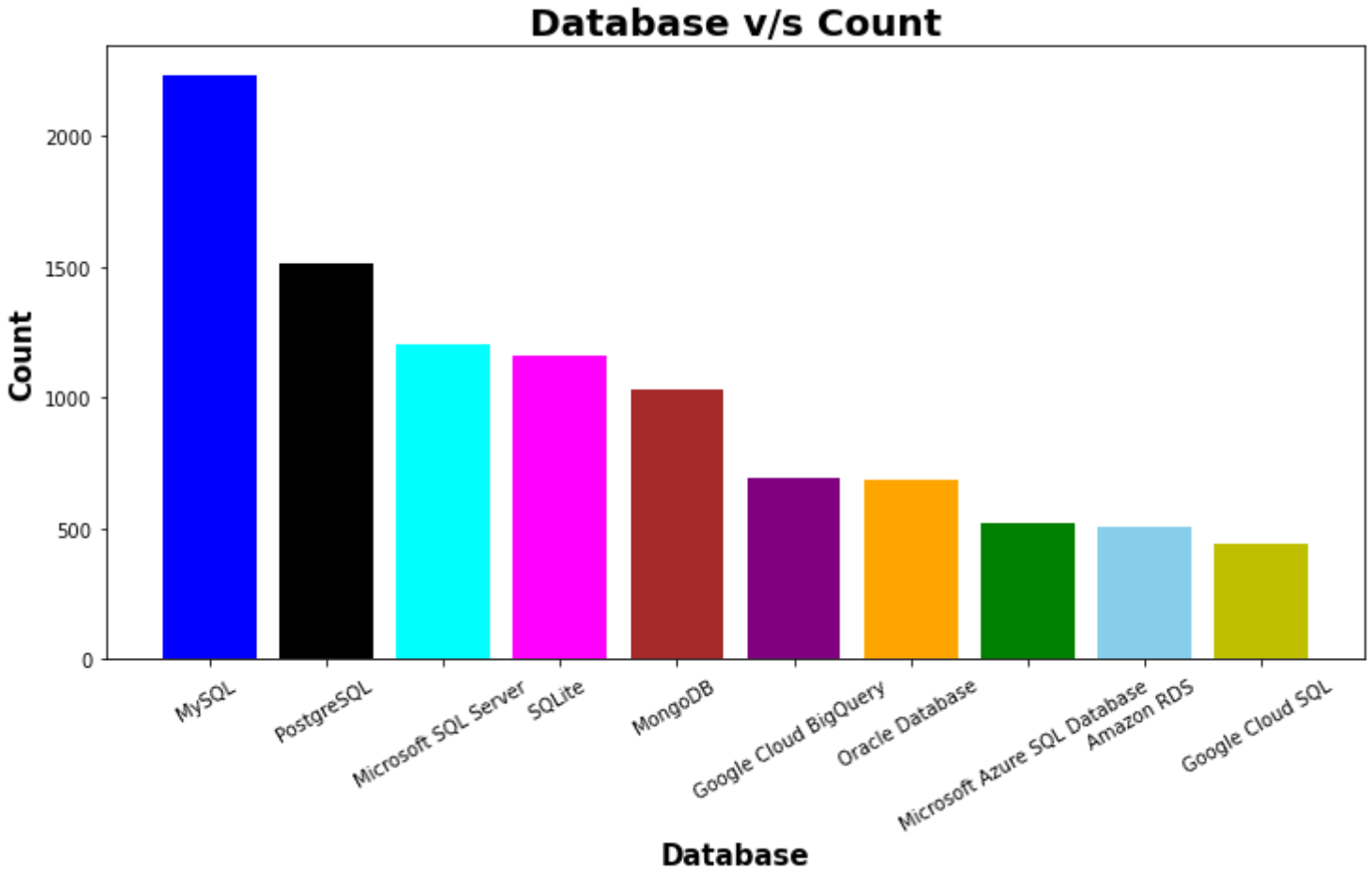
In [150]:
```
# Create a list of Categorical values.
x9 = ['MySQL ','PostgreSQL ','Microsoft SQL Server ','SQLite ','MongoDB ','Google Cloud BigQuery ',
      'Oracle Database ','Microsoft Azure SQL Database ','Amazon RDS ','Google Cloud SQL ']
x9
```

Out[150]:
```
['MySQL ',
 'PostgreSQL ',
 'Microsoft SQL Server ',
 'SQLite ',
 'MongoDB ',
 'Google Cloud BigQuery ',
 'Oracle Database ',
 'Microsoft Azure SQL Database ',
 'Amazon RDS ',
 'Google Cloud SQL ']
```

In [151]:
```
# Create a list of Numerical values.
y9 = list(data_base)
y9
```

Out[151]:
```
[2233.0, 1516.0, 1203.0, 1159.0, 1031.0, 690.0, 688.0, 520.0, 505.0, 439.0]
```

In [152]:
```
# Plot a Bar graph
plt.figure(figsize=(12,6))
plt.bar(x9, y9,color=['blue','black','Cyan','magenta','brown','purple','Orange','g','skyblue','y',])
plt.xticks(rotation=30, horizontalalignment="center")
plt.title('Database v/s Count',fontweight='bold',fontsize=20)
plt.xlabel('Database',fontweight='bold',fontsize=15)
plt.ylabel('Count',fontweight='bold',fontsize=15)
plt.show()
```



' MySQL ' is dominantly used as Database.

Do you use any of the following business intelligence tools?

In [153]:
```
# Here we extract total count of BI tools.
BI_Tool1 = df['Q36_1'].value_counts()[:1]
BI_Tool1
```

Out[153]:
```
Amazon QuickSight    224
Name: Q36_1, dtype: int64
```

In [154]:
```
BI_Tool2 = df['Q36_2'].value_counts()[:1]
BI_Tool2
```

Out[154]:
```
Microsoft Power BI    1658
Name: Q36_2, dtype: int64
```

In [155]:
```
BI_Tool3 = df['Q36_3'].value_counts()[:1]
BI_Tool3
```

Out[155]:
```
Google Data Studio    643
Name: Q36_3, dtype: int64
```

In [156]:
```
BI_Tool4 = df['Q36_4'].value_counts()[:1]
BI_Tool4
```

Out[156]:
```
Looker    166
Name: Q36_4, dtype: int64
```

In [157]:
```
BI_Tool5 = df['Q36_5'].value_counts()[:1]
BI_Tool5
```

Out[157]:
```
Tableau    1732
Name: Q36_5, dtype: int64
```

In [158]:
```
BI_Tool6 = df['Q36_6'].value_counts()[:1]
BI_Tool6
```

Out[158]:
```
Qlik Sense    207
Name: Q36_6, dtype: int64
```

In [159]:
```
BI_Tool7 = df['Q36_7'].value_counts()[:1]
BI_Tool7
```

Out[159]:
```
Domo    44
Name: Q36_7, dtype: int64
```

In [160]:
```
BI_Tool8 = df['Q36_8'].value_counts()[:1]
BI_Tool8
```

Out[160]:
```
TIBCO Spotfire    86
Name: Q36_8, dtype: int64
```

In [161]:
```
BI_Tool9 = df['Q36_9'].value_counts()[:1]
BI_Tool9
```

Out[161]:
```
Alteryx     132
Name: Q36_9, dtype: int64
```

In [162]:
```
BI_Tool10 = df['Q36_10'].value_counts()[:1]
BI_Tool10
```

Out[162]:
```
Sisense     38
Name: Q36_10, dtype: int64
```

In [163]:
```
BI_Tool11 = df['Q36_11'].value_counts()[:1]
BI_Tool11
```

Out[163]:
```
SAP Analytics Cloud     106
Name: Q36_11, dtype: int64
```

In [164]:
```
BI_Tool12 = df['Q36_12'].value_counts()[:1]
BI_Tool12
```

Out[164]:
```
Microsoft Azure Synapse     167
Name: Q36_12, dtype: int64
```

In [165]:
```
BI_Tool13 = df['Q36_13'].value_counts()[:1]
BI_Tool13
```

Out[165]:
```
Thoughtspot     22
Name: Q36_13, dtype: int64
```

In [166]:
```
# create a new data frame with above information.Get the sum of elements of an iterable & then Sort them in desending
BI_Tool = pd.DataFrame([BI_Tool1,BI_Tool2,BI_Tool3,BI_Tool4,BI_Tool5,BI_Tool6,BI_Tool7,BI_Tool8,
                        BI_Tool9,BI_Tool10,BI_Tool11,BI_Tool12,BI_Tool13]).sum().sort_values(ascending=False)[:5]
BI_Tool
```

Out[166]:
```
Tableau               1732.0
Microsoft Power BI    1658.0
Google Data Studio     643.0
Amazon QuickSight      224.0
Qlik Sense             207.0
dtype: float64
```

In [167]:
```
# Create a list of Numerical values.
x10 = list(BI_Tool)
x10
```

Out[167]:
```
[1732.0, 1658.0, 643.0, 224.0, 207.0]
```

In [168]:
```
# Create a list of Categorical values.
y10= ['Tableau','Microsoft Power BI','Google Data Studio','Amazon QuickSight','Qlik Sense']
y10
```

Out[168]:
```
['Tableau',
 'Microsoft Power BI',
 'Google Data Studio',
 'Amazon QuickSight',
 'Qlik Sense']
```

In [169]:
```
# Plot a Pie Chart
plt.figure(figsize=(8,8))
plt.pie(x10,labels=y10, shadow=True, autopct ='%0.1f%%')
plt.show()
```

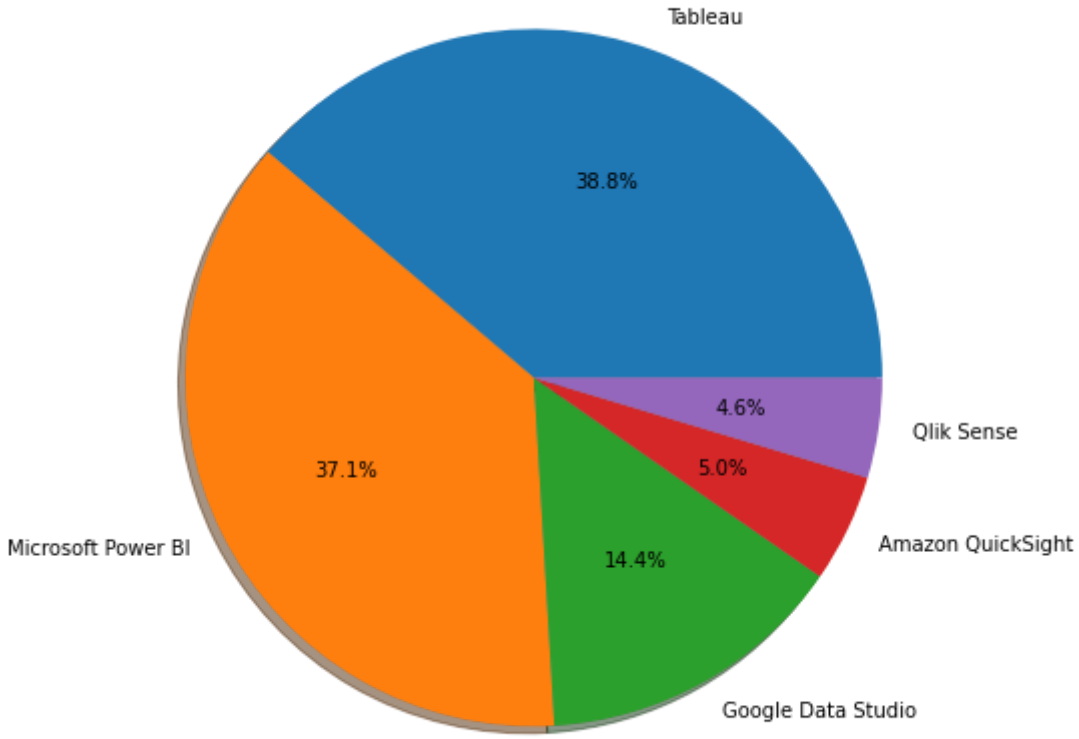


Tableau & Microsoft Power BI these two visualization tools captured almost 76% market .

Do you use any of the following types of specialized hardware when training machine learning models?

In [170]:
```
# Here we extract total count of hardware uesd while training machine learning models.
h1 = df['Q42_1'].value_counts()[:1]
h1
```

Out[170]:
```
 GPUs     2682
Name: Q42_1, dtype: int64
```

In [171]:
```
h2 = df['Q42_2'].value_counts()[:1]
h2
```

Out[171]:
```
 TPUs     653
Name: Q42_2, dtype: int64
```

In [172]:
```
h3 = df['Q42_3'].value_counts()[:1]
h3
```

Out[172]:
```
 IPUs     67
Name: Q42_3, dtype: int64
```

In [173]:
```
h4 = df['Q42_4'].value_counts()[:1]
h4
```

Out[173]:
```
 RDUs     58
Name: Q42_4, dtype: int64
```

In [174]:
```
h5 = df['Q42_5'].value_counts()[:1]
h5
```

Out[174]:
```
 WSEs     26
Name: Q42_5, dtype: int64
```

In [175]:
```
h6 = df['Q42_6'].value_counts()[:1]
h6
```

Out[175]:
```
 Trainium Chips     39
Name: Q42_6, dtype: int64
```

In [176]:
```
h7 = df['Q42_7'].value_counts()[:1]
h7
```

Out[176]:
```
 Inferentia Chips     58
Name: Q42_7, dtype: int64
```

In [177]:
```
h8 = df['Q42_8'].value_counts()[:1]
h8
```

Out[177]:
```
None    1772
Name: Q42_8, dtype: int64
```

In [178]:
```
# create a new data frame with above information.Get the sum of elements of an iterable & then Sort them in desending
hardware = pd.DataFrame([h1,h2,h3,h4,h5,h6,h7,h8]).sum().sort_values(ascending=False)
hardware
```

Out[178]:
```
 GPUs                2682.0
None                 1772.0
 TPUs                 653.0
 IPUs                  67.0
 RDUs                  58.0
 Inferentia Chips      58.0
 Trainium Chips        39.0
 WSEs                  26.0
dtype: float64
```

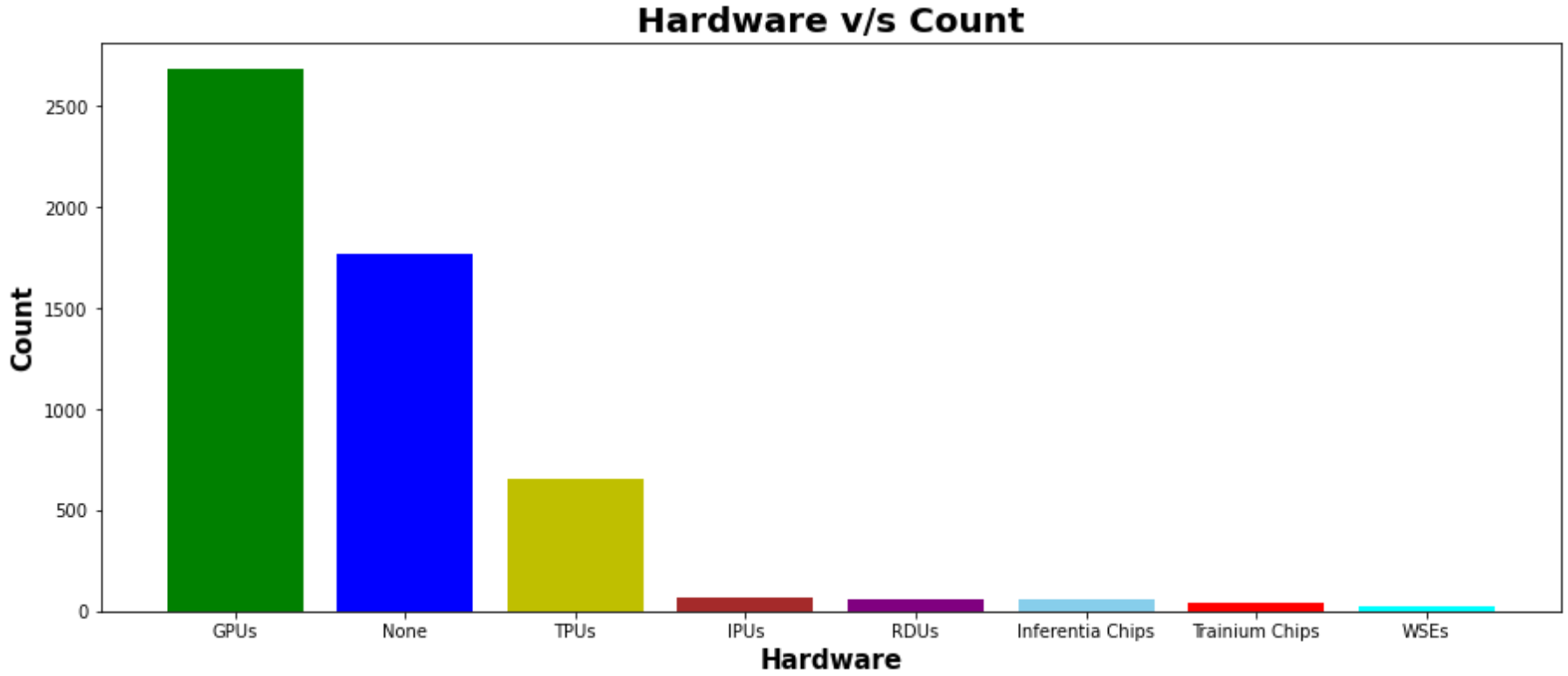
In [179]:
```
# Create a list of Numerical values.
y11 = list(hardware)
y11
```

Out[179]:
```
[2682.0, 1772.0, 653.0, 67.0, 58.0, 58.0, 39.0, 26.0]
```

In [180]:
```
# Create a list of Categorical values.
x11 = ['GPUs','None','TPUs','IPUs','RDUs','Inferentia Chips','Trainium Chips','WSEs']
x11
```

Out[180]:
```
['GPUs',
 'None',
 'TPUs',
 'IPUs',
 'RDUs',
 'Inferentia Chips',
 'Trainium Chips',
 'WSEs']
```

In [181]:
```
# Plot a bar graph
plt.figure(figsize=(15,6))
plt.bar(x11,y11,color=['g','b','y','brown','purple','skyblue','r','Cyan'])
plt.title('Hardware v/s Count',fontweight='bold',fontsize=20)
plt.xlabel('Hardware',fontweight='bold',fontsize=15)
plt.ylabel('Count',fontweight='bold',fontsize=15)
plt.show()
```



" GPUs " play an important role while training machine learning models.

what are your favorite media sources that report on data science topics?

In [182]:
```
# Here we extract total count of media sources that report on data science topics.
m1 = df['Q44_1'].value_counts()[:1]
m1
```

Out[182]:
```
Twitter (data science influencers)    3995
Name: Q44_1, dtype: int64
```

In [183]:
```
m2 = df['Q44_2'].value_counts()[:1]
m2
```

Out[183]:
```
Email newsletters (Data Elixir, O'Reilly Data & AI, etc)    3787
Name: Q44_2, dtype: int64
```

In [184]:
```
m3 = df['Q44_3'].value_counts()[:1]
m3
```

Out[184]:
```
Reddit (r/machinelearning, etc)    2678
Name: Q44_3, dtype: int64
```

In [185]:
```
m4 = df['Q44_4'].value_counts()[:1]
m4
```

Out[185]:
```
Kaggle (notebooks, forums, etc)    11181
Name: Q44_4, dtype: int64
```

In [186]:
```
m5 = df['Q44_5'].value_counts()[:1]
m5
```

Out[186]:
```
Course Forums (forums.fast.ai, Coursera forums, etc)    4006
Name: Q44_5, dtype: int64
```

In [187]:
```
m6 = df['Q44_6'].value_counts()[:1]
m6
```

Out[187]:
```
YouTube (Kaggle YouTube, Cloud AI Adventures, etc)    11957
Name: Q44_6, dtype: int64
```

In [188]:
```
m7 = df['Q44_7'].value_counts()[:1]
m7
```

Out[188]:
```
Podcasts (Chai Time Data Science, O’Reilly Data Show, etc)    2120
Name: Q44_7, dtype: int64
```

In [189]:
```
m8 = df['Q44_8'].value_counts()[:1]
m8
```

Out[189]:
```
Blogs (Towards Data Science, Analytics Vidhya, etc)    7766
Name: Q44_8, dtype: int64
```

In [190]:
```
m9 = df['Q44_9'].value_counts()[:1]
m9
```

Out[190]:
```
Journal Publications (peer-reviewed journals, conference proceedings, etc)    3804
Name: Q44_9, dtype: int64
```

In [191]:
```
m10 = df['Q44_10'].value_counts()[:1]
m10
```

Out[191]:
```
Slack Communities (ods.ai, kagglenoobs, etc)    1726
Name: Q44_10, dtype: int64
```

In [192]:
```
# create a new data frame with above information.Get the sum of elements of an iterable & then Sort them in desending
media = pd.DataFrame([m1,m2,m3,m4,m5,m6,m7,m8,m9,m10,]).sum().sort_values(ascending = False)
media
```

Out[192]:
```
YouTube (Kaggle YouTube, Cloud AI Adventures, etc)                          11957.0
Kaggle (notebooks, forums, etc)                                             11181.0
Blogs (Towards Data Science, Analytics Vidhya, etc)                          7766.0
Course Forums (forums.fast.ai, Coursera forums, etc)                         4006.0
Twitter (data science influencers)                                           3995.0
Journal Publications (peer-reviewed journals, conference proceedings, etc)   3804.0
Email newsletters (Data Elixir, O'Reilly Data & AI, etc)                     3787.0
Reddit (r/machinelearning, etc)                                              2678.0
Podcasts (Chai Time Data Science, O’Reilly Data Show, etc)                   2120.0
Slack Communities (ods.ai, kagglenoobs, etc)                                 1726.0
dtype: float64
```

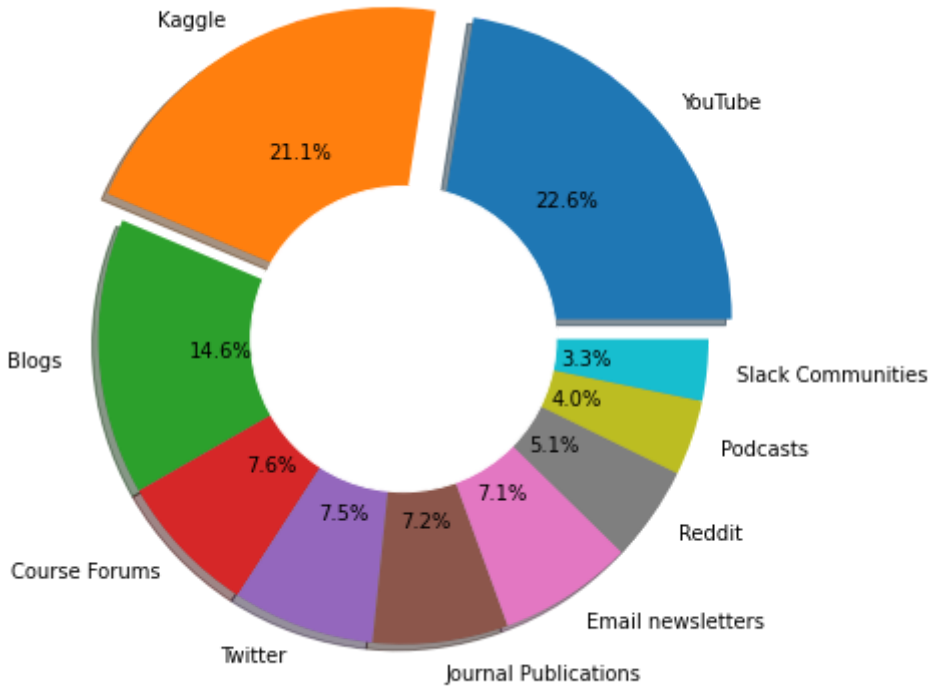
In [193]:
```
# Create a list of Numerical values.
x12 = list(media)
x12
```

Out[193]:
```
[11957.0,
 11181.0,
 7766.0,
 4006.0,
 3995.0,
 3804.0,
 3787.0,
 2678.0,
 2120.0,
 1726.0]
```

In [194]:
```
# Create a list of Categorical values.
y12 = ['YouTube ','Kaggle', 'Blogs ','Course Forums','Twitter ','Journal Publications','Email newsletters','Reddit ',
       'Podcasts','Slack Communities']
y12
```

Out[194]:
```
['YouTube ',
 'Kaggle',
 'Blogs ',
 'Course Forums',
 'Twitter ',
 'Journal Publications',
 'Email newsletters',
 'Reddit ',
 'Podcasts',
 'Slack Communities']
```

In [196]:
```
# Plot a  Donut Chart
plt.figure(figsize=(7,7))
explode = (0.1, 0.1, 0, 0, 0, 0, 0, 0, 0, 0)
plt.pie(x12,labels=y12, shadow=True, autopct ='%0.1f%%', explode=explode)
circle = plt.Circle( (0,0), 0.5, color='white')
p=plt.gcf()
p.gca().add_artist(circle)
plt.show()
```



Youtube & Kaggle these two media sourses helped data scientists to grow and learn.

Conclusion :

Maximum youngsters having age of 18-29 years and are participated in this survey. With 51.4 % votes india domonatly participated in the survey.Platformes like 'Coursers & Kaggle' palyed a vital role for learning Data Science course.39% people completed their Masters Degree and 32.6% people completed Bachelor’s Degree.To explore in the field of Data Science 'Python & SQL' these two programming languages are widely used.Matplotlib & Seaborn these two libraries are mostly used for data visualization.Linear Regression, Logistic Regression, Decision Trees, Random Forests these ML algorithms are regularly used by data scientists. Data base plays an important role in data science.MySQL is dominantly used for the same.Tableau & Microsoft Power BI these two visualization tools captured almost 76% market.'GPUs' play an important role while training machine learning models. Youtube & Kaggle these two media sourses helped data scientists to grow and learn.