In [1]:
```
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:
```
data = pd.read_csv("covid_vaccine_statewise.csv")
```

In [3]:
```
data.head()
```

Out[3]:

	Updated On	State	Total Doses Administered	Sessions	Sites	First Dose Administered	Second Dose Administered	Male (Doses Administered)	Female (Doses Administered)	Trans Admini
0	16/01/2021	India	48276.0	3455.0	2957.0	48276.0	0.0	NaN	NaN	
1	17/01/2021	India	58604.0	8532.0	4954.0	58604.0	0.0	NaN	NaN	
2	18/01/2021	India	99449.0	13611.0	6583.0	99449.0	0.0	NaN	NaN	
3	19/01/2021	India	195525.0	17855.0	7951.0	195525.0	0.0	NaN	NaN	
4	20/01/2021	India	251280.0	25472.0	10504.0	251280.0	0.0	NaN	NaN	

5 rows × 24 columns

In [4]:
```
print("The shape is: ",data.shape)
```

```
The shape is:  (7845, 24)
```

In [5]:
```
print("The columns present in the dataset are: ")
data.columns
```

```
The columns present in the dataset are:
```

Out[5]:
```
Index(['Updated On', 'State', 'Total Doses Administered', 'Sessions',
       ' Sites ', 'First Dose Administered', 'Second Dose Administered',
       'Male (Doses Administered)', 'Female (Doses Administered)',
       'Transgender (Doses Administered)', ' Covaxin (Doses Administered)',
       'CoviShield (Doses Administered)', 'Sputnik V (Doses Administered)',
       'AEFI', '18-44 Years (Doses Administered)',
       '45-60 Years (Doses Administered)', '60+ Years (Doses Administered)',
       '18-44 Years(Individuals Vaccinated)',
       '45-60 Years(Individuals Vaccinated)',
       '60+ Years(Individuals Vaccinated)', 'Male(Individuals Vaccinated)',
       'Female(Individuals Vaccinated)', 'Transgender(Individuals Vaccinated)',
       'Total Individuals Vaccinated'],
      dtype='object')
```

Describe the dataset

In [6]:
```
data.describe(include='object')
```

Out[6]:

	Updated On	State
count	7845	7845
unique	213	37
top	16/01/2021	Delhi
freq	37	213

In [7]:
```
data.describe()
```

Out[7]:

	Total Doses Administered	Sessions	Sites	First Dose Administered	Second Dose Administered	Male (Doses Administered)	Female (Doses Administered)	Transgend (Dose Administere
count	7.621000e+03	7.621000e+03	7621.000000	7.621000e+03	7.621000e+03	7.461000e+03	7.461000e+03	7461.00000
mean	9.188171e+06	4.792358e+05	2282.872064	7.414415e+06	1.773755e+06	3.620156e+06	3.168416e+06	1162.9780
std	3.746180e+07	1.911511e+06	7275.973730	2.995209e+07	7.570382e+06	1.737938e+07	1.515310e+07	5931.35399
min	7.000000e+00	0.000000e+00	0.000000	7.000000e+00	0.000000e+00	0.000000e+00	2.000000e+00	0.00000
25%	1.356570e+05	6.004000e+03	69.000000	1.166320e+05	1.283100e+04	5.655500e+04	5.210700e+04	8.00000
50%	8.182020e+05	4.547000e+04	597.000000	6.614590e+05	1.388180e+05	3.897850e+05	3.342380e+05	113.00000
75%	6.625243e+06	3.428690e+05	1708.000000	5.387805e+06	1.166434e+06	2.735777e+06	2.561513e+06	800.00000
max	5.132284e+08	3.501031e+07	73933.000000	4.001504e+08	1.130780e+08	2.701636e+08	2.395186e+08	98275.00000

8 rows × 22 columns

In [8]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 7845 entries, 0 to 7844
Data columns (total 24 columns):
 #   Column                                Non-Null Count  Dtype
---  ------                                --------------  -----
 0   Updated On                            7845 non-null   object
 1   State                                 7845 non-null   object
 2   Total Doses Administered              7621 non-null   float64
 3   Sessions                              7621 non-null   float64
 4    Sites                                7621 non-null   float64
 5   First Dose Administered               7621 non-null   float64
 6   Second Dose Administered              7621 non-null   float64
 7   Male (Doses Administered)             7461 non-null   float64
 8   Female (Doses Administered)           7461 non-null   float64
 9   Transgender (Doses Administered)      7461 non-null   float64
 10   Covaxin (Doses Administered)         7621 non-null   float64
 11  CoviShield (Doses Administered)       7621 non-null   float64
 12  Sputnik V (Doses Administered)        2995 non-null   float64
 13  AEFI                                  5438 non-null   float64
 14  18-44 Years (Doses Administered)      1702 non-null   float64
 15  45-60 Years (Doses Administered)      1702 non-null   float64
 16  60+ Years (Doses Administered)        1702 non-null   float64
 17  18-44 Years(Individuals Vaccinated)   3733 non-null   float64
 18  45-60 Years(Individuals Vaccinated)   3734 non-null   float64
 19  60+ Years(Individuals Vaccinated)     3734 non-null   float64
 20  Male(Individuals Vaccinated)          160 non-null    float64
 21  Female(Individuals Vaccinated)        160 non-null    float64
 22  Transgender(Individuals Vaccinated)   160 non-null    float64
 23  Total Individuals Vaccinated          5919 non-null   float64
dtypes: float64(22), object(2)
memory usage: 1.4+ MB
```

In [9]:
```
data.isnull().sum()
```

Out[9]:
```
Updated On                              0
State                                   0
Total Doses Administered              224
Sessions                              224
 Sites                                224
First Dose Administered               224
Second Dose Administered              224
Male (Doses Administered)             384
Female (Doses Administered)           384
Transgender (Doses Administered)      384
 Covaxin (Doses Administered)         224
CoviShield (Doses Administered)       224
Sputnik V (Doses Administered)       4850
AEFI                                 2407
18-44 Years (Doses Administered)     6143
45-60 Years (Doses Administered)     6143
60+ Years (Doses Administered)       6143
18-44 Years(Individuals Vaccinated)  4112
45-60 Years(Individuals Vaccinated)  4111
60+ Years(Individuals Vaccinated)    4111
Male(Individuals Vaccinated)         7685
Female(Individuals Vaccinated)       7685
Transgender(Individuals Vaccinated)  7685
Total Individuals Vaccinated         1926
dtype: int64
```

As there are many NULL values present in the given dataset. We need to replace those values by mean(in case of numerical data) or mode(in case of categorical data). Here, we need to work on "First Dose Administered" and "Second Dose Administered". Both of them are float, hence we will replace the Nan Values by mean(average).

For First Dose Administered

In [10]:
```
avg_firstdose = data["First Dose Administered"].astype("float").mean(axis = 0)
print("Average of First Dose:", avg_firstdose)
```

```
Average of First Dose: 7414415.300354284
```

In [11]:
```
data["First Dose Administered"].fillna(value = avg_firstdose, inplace=True)
data.head()
```

Out[11]:

	Updated On	State	Total Doses Administered	Sessions	Sites	First Dose Administered	Second Dose Administered	Male (Doses Administered)	Female (Doses Administered)	Trans Admini
0	16/01/2021	India	48276.0	3455.0	2957.0	48276.0	0.0	NaN	NaN	
1	17/01/2021	India	58604.0	8532.0	4954.0	58604.0	0.0	NaN	NaN	
2	18/01/2021	India	99449.0	13611.0	6583.0	99449.0	0.0	NaN	NaN	
3	19/01/2021	India	195525.0	17855.0	7951.0	195525.0	0.0	NaN	NaN	
4	20/01/2021	India	251280.0	25472.0	10504.0	251280.0	0.0	NaN	NaN	

5 rows × 24 columns

For Second Dose Administered

In [12]:
```
avg_seconddose = data["Second Dose Administered"].astype("float").mean(axis = 0)
print("Average of Second Dose:", avg_seconddose)
```

```
Average of Second Dose: 1773755.2436688098
```

In [13]:
```
data["Second Dose Administered"].fillna(value = avg_seconddose, inplace = True)
data.head()
```

Out[13]:

	Updated On	State	Total Doses Administered	Sessions	Sites	First Dose Administered	Second Dose Administered	Male (Doses Administered)	Female (Doses Administered)	Trans Admini
0	16/01/2021	India	48276.0	3455.0	2957.0	48276.0	0.0	NaN	NaN	
1	17/01/2021	India	58604.0	8532.0	4954.0	58604.0	0.0	NaN	NaN	
2	18/01/2021	India	99449.0	13611.0	6583.0	99449.0	0.0	NaN	NaN	
3	19/01/2021	India	195525.0	17855.0	7951.0	195525.0	0.0	NaN	NaN	
4	20/01/2021	India	251280.0	25472.0	10504.0	251280.0	0.0	NaN	NaN	

5 rows × 24 columns

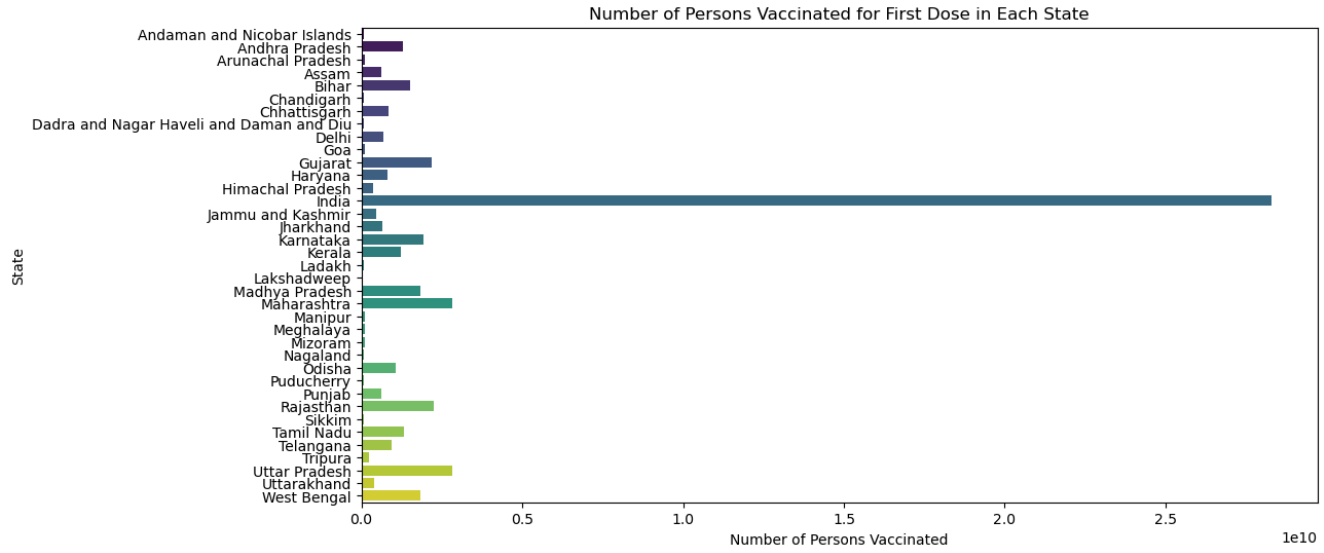
Number of persons state wise vaccinated for first dose in India

In [14]:
```
first_dose = data.groupby('State')[['First Dose Administered']].sum()
first_dose
```

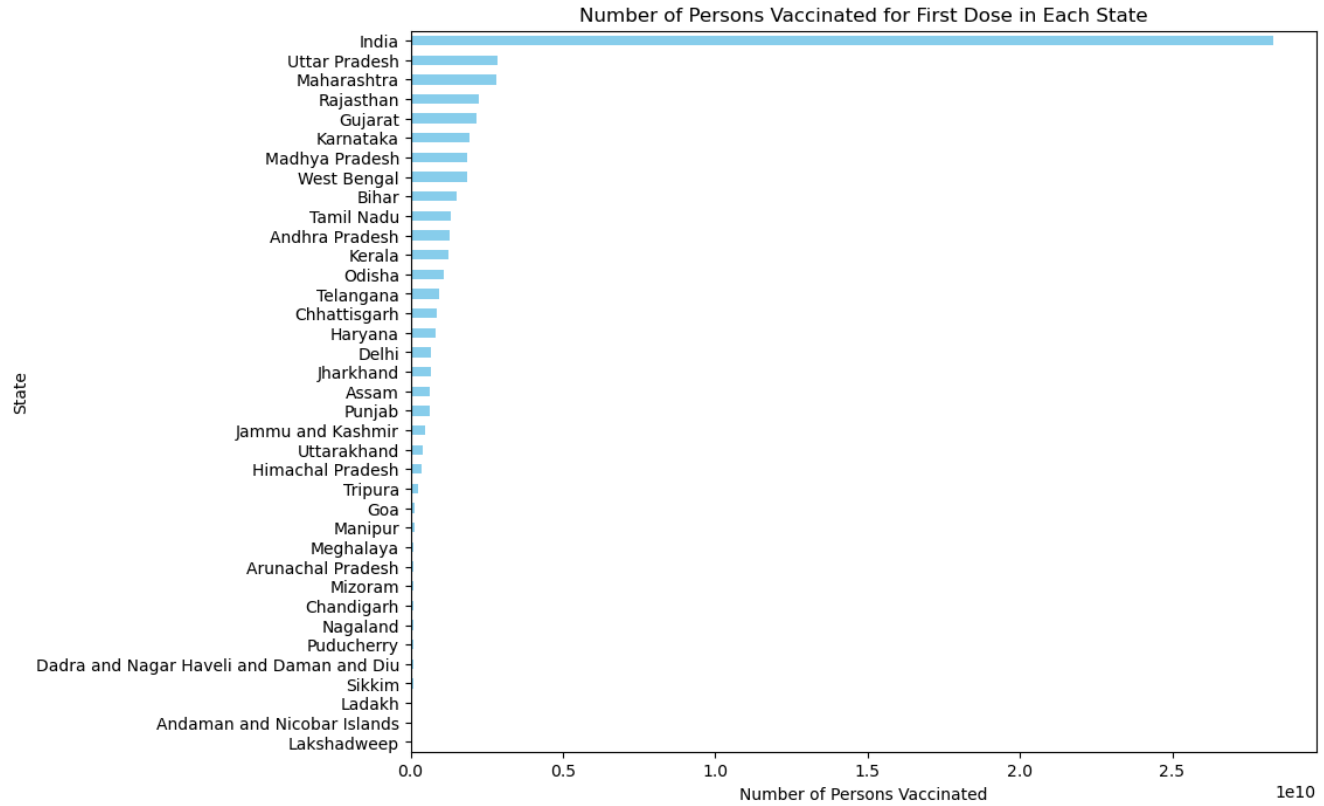
Out[14]:

State	First Dose Administered
Andaman and Nicobar Islands	6.091235e+07
Andhra Pradesh	1.277347e+09
Arunachal Pradesh	9.349147e+07
Assam	6.300867e+08
Bihar	1.514989e+09
Chandigarh	8.918960e+07
Chhattisgarh	8.404894e+08
Dadra and Nagar Haveli and Daman and Diu	8.549597e+07
Delhi	6.762404e+08
Goa	1.204779e+08

In [15]:
```
# b. Number of persons state-wise vaccinated for the first dose in India
first_dose_counts = data.groupby('State')['First Dose Administered'].sum()
# Visualizations
# b. Number of persons state-wise vaccinated for the first dose in India
plt.figure(figsize=(12, 6))
sns.barplot(x=first_dose_counts.values, y=first_dose_counts.index, palette="viridis")
plt.title('Number of Persons Vaccinated for First Dose in Each State')
plt.xlabel('Number of Persons Vaccinated')
plt.ylabel('State')
plt.show()
```



In [16]:
```
plt.figure(figsize=(10, 8))
first_dose_counts.sort_values().plot(kind='barh', color='skyblue')
plt.xlabel('Number of Persons Vaccinated')
plt.ylabel('State')
plt.title('Number of Persons Vaccinated for First Dose in Each State')
plt.show()
```



Number of persons state wise vaccinated for second dose in India

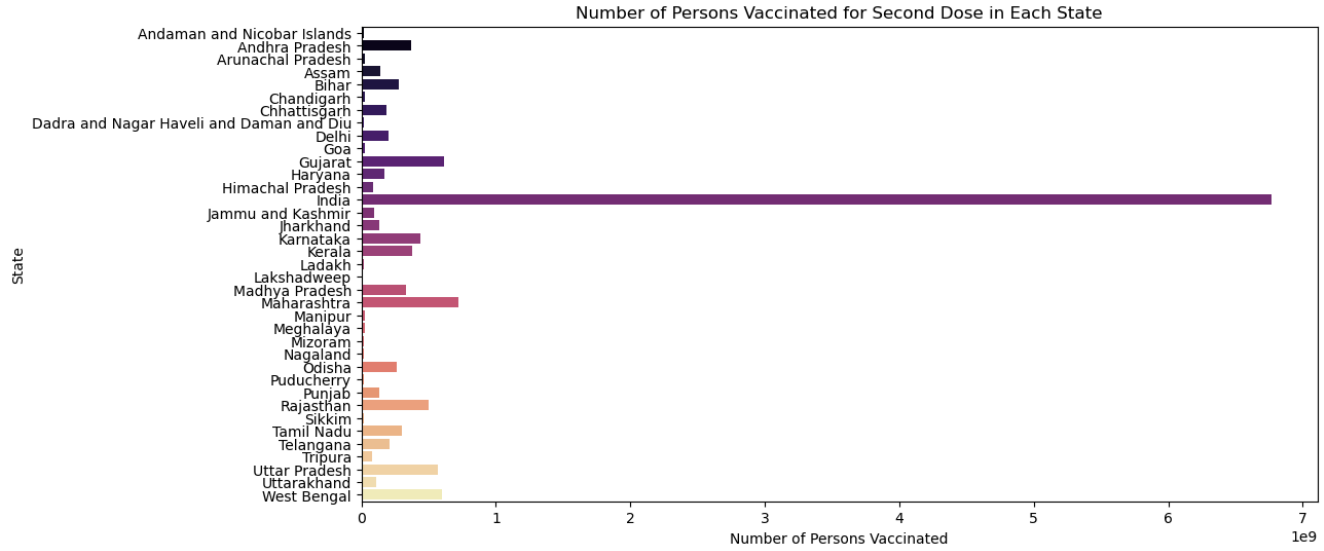
In [17]:
```
first_dose = data.groupby('State')[['Second Dose Administered']].sum()
first_dose
```

Out[17]:

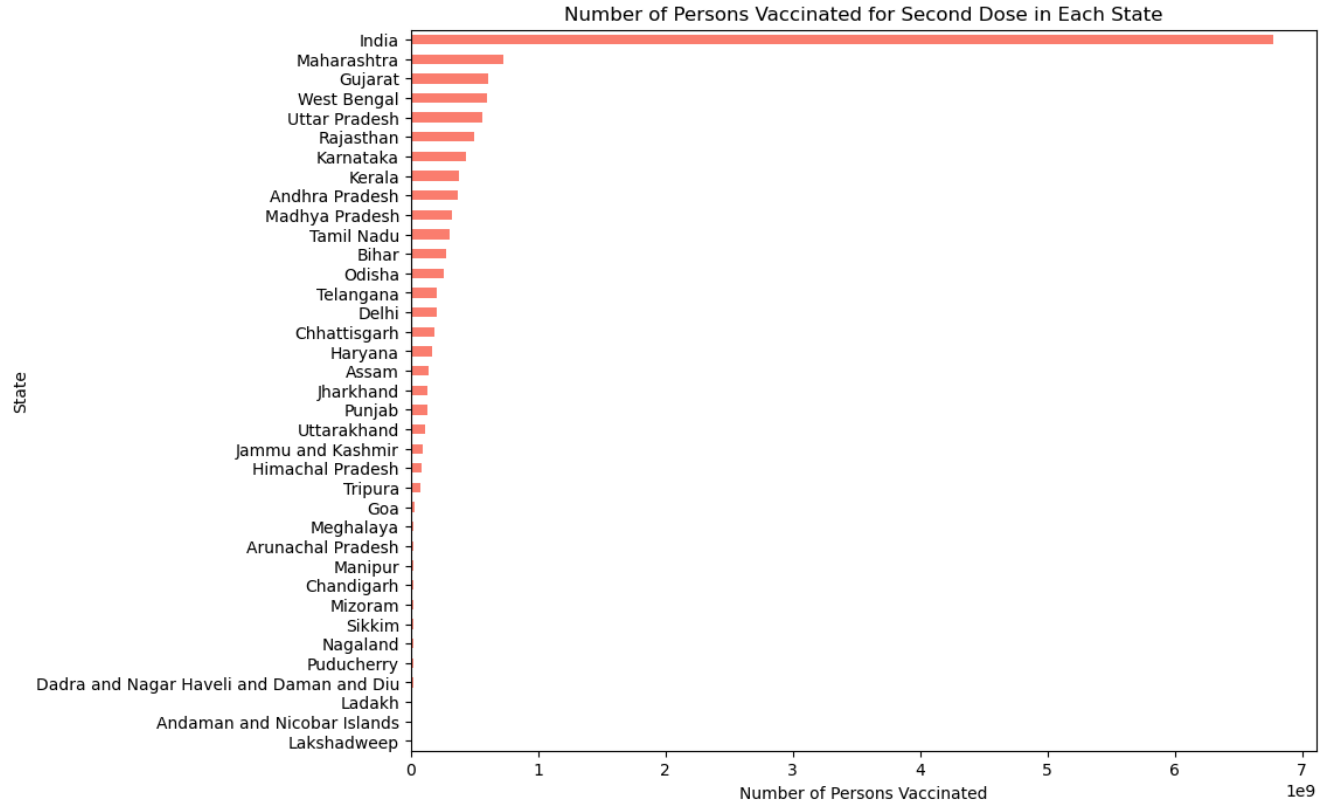
State	Second Dose Administered
Andaman and Nicobar Islands	1.476109e+07
Andhra Pradesh	3.694601e+08
Arunachal Pradesh	2.257485e+07
Assam	1.414313e+08
Bihar	2.814331e+08
Chandigarh	2.223627e+07
Chhattisgarh	1.827629e+08
Dadra and Nagar Haveli and Daman and Diu	1.701070e+07
Delhi	2.006352e+08
Goa	2.684071e+07
Gujarat	6.110609e+08
Haryana	1.692986e+08
Himachal Pradesh	8.448111e+07
India	6.770264e+09
Jammu and Kashmir	9.659418e+07
Jharkhand	1.327636e+08
Karnataka	4.378297e+08
Kerala	3.746913e+08
Ladakh	1.609629e+07
Lakshadweep	1.169898e+07
Madhya Pradesh	3.275755e+08
Maharashtra	7.235236e+08
Manipur	2.250068e+07
Meghalaya	2.280916e+07
Mizoram	2.064095e+07
Nagaland	1.984717e+07
Odisha	2.619453e+08
Puducherry	1.925139e+07
Punjab	1.317635e+08
Rajasthan	5.023455e+08
Sikkim	2.036617e+07
Tamil Nadu	3.013132e+08
Telangana	2.087955e+08
Tripura	7.591267e+07
Uttar Pradesh	5.650776e+08
Uttarakhand	1.107276e+08
West Bengal	5.967894e+08

In [18]:
```
# c. Number of persons state-wise vaccinated for the second dose in India
second_dose_counts = data.groupby('State')['Second Dose Administered'].sum()

# c. Number of persons state-wise vaccinated for the second dose in India
plt.figure(figsize=(12, 6))
sns.barplot(x=second_dose_counts.values, y=second_dose_counts.index, palette="magma")
plt.title('Number of Persons Vaccinated for Second Dose in Each State')
plt.xlabel('Number of Persons Vaccinated')
plt.ylabel('State')
plt.show()
```



In [19]:
```
plt.figure(figsize=(10, 8))
second_dose_counts.sort_values().plot(kind='barh', color='salmon')
plt.xlabel('Number of Persons Vaccinated')
plt.ylabel('State')
plt.title('Number of Persons Vaccinated for Second Dose in Each State')
plt.show()
```



Number of Males vaccinated

In [21]:
```
male = data["Male(Individuals Vaccinated)"].sum()
print("The total number of male individuals vaccinated are", int(male))
```

```
The total number of male individuals vaccinated are 7138698858
```

Number of females vaccinated

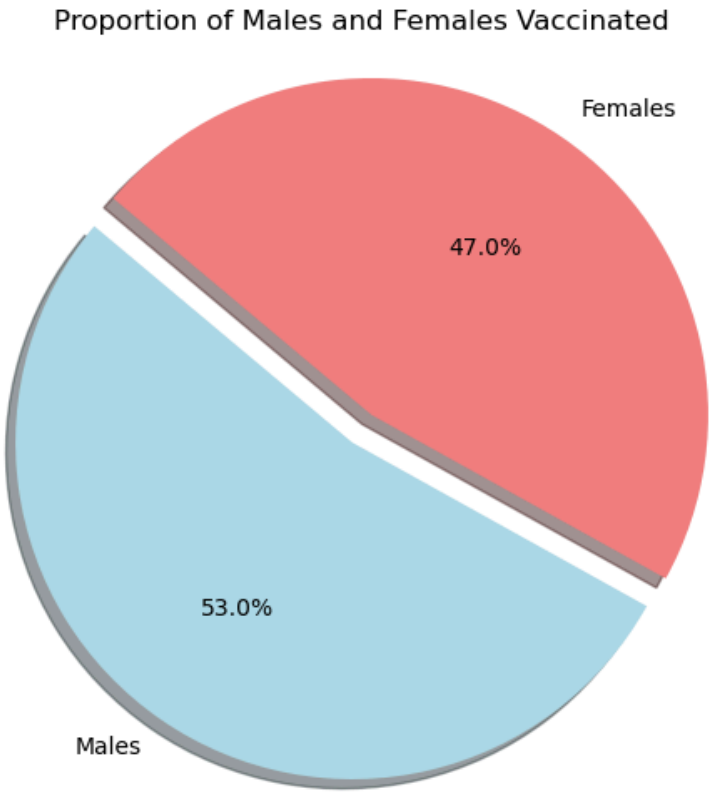
In [23]:
```
female = data["Female(Individuals Vaccinated)"].sum()
print("The total number of female individuals vaccinated are", int(female))
```

```
The total number of female individuals vaccinated are 6321628736
```

In [24]:
```
# Visualize Number of Males and Females vaccinated using a pie chart
plt.figure(figsize=(10, 6))

# Data for the pie chart
labels = ['Males', 'Females']
sizes = [male, female]
colors = ['lightblue', 'lightcoral']
explode = (0, 0.1)  # explode the 'Females' slice slightly

# Plotting the pie chart
plt.pie(sizes, explode=explode, labels=labels, colors=colors, autopct='%1.1f%%', shadow=True, star
plt.axis('equal')  # Equal aspect ratio ensures that pie is drawn as a circle.
plt.title('Proportion of Males and Females Vaccinated')
plt.show()
```



In [64]:
```
plt.figure(figsize=(10, 6))
plt.bar('Females', female, color='lightcoral')
plt.bar('Males', male, color='lightblue')
plt.ylabel('Number of Individuals Vaccinated')
plt.title('Number of Individuals Vaccinated by Gender')
plt.legend(['Females', 'Males'])
plt.show()
```