Space Mission Launches- Exploratory Data Analysis

In [2]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
import datetime
import warnings
warnings.filterwarnings("ignore")
```

In [3]:
```
df = pd.read_csv("mission_launches.csv")
df.head()
```

Out[3]:

	Unnamed: 0.1	Unnamed: 0	Organisation	Location	Date	Detail	Rocket_Status
0	0	0	SpaceX	LC-39A, Kennedy Space Center, Florida, USA	Fri Aug 07, 2020 05:12 UTC	Falcon 9 Block 5 | Starlink V1 L9 & BlackSky	StatusActive
1	1	1	CASC	Site 9401 (SLS-2), Jiuquan Satellite Launch Ce...	Thu Aug 06, 2020 04:01 UTC	Long March 2D | Gaofen-9 04 & Q-SAT	StatusActive
2	2	2	SpaceX	Pad A, Boca Chica, Texas, USA	Tue Aug 04, 2020 23:57 UTC	Starship Prototype | 150 Meter Hop	StatusActive
3	3	3	Roscosmos	Site 200/39, Baikonur Cosmodrome, Kazakhstan	Thu Jul 30, 2020 21:25 UTC	Proton-M/Briz-M | Ekspress-80 & Ekspress-103	StatusActive
4	4	4	ULA	SLC-41, Cape Canaveral AFS, Florida, USA	Thu Jul 30, 2020 11:50 UTC	Atlas V 541 | Perseverance	StatusActive

In [4]:
```python
# Check null values
((df.isnull().sum())*100 / df.shape[0]).sort_values(ascending = False)
```

Out[4]:
```
Price             77.705828
Unnamed: 0.1       0.000000
Unnamed: 0         0.000000
Organisation       0.000000
Location           0.000000
Date               0.000000
Detail             0.000000
Rocket_Status      0.000000
Mission_Status     0.000000
dtype: float64
```

It has been recommended to remove the price column from the analysis as it contains 77% null values.

In [5]:
```python
# Remove unwanted columns
df.drop(['Unnamed: 0', 'Unnamed: 0.1', 'Price'], axis = 1, inplace = True)
```

In [6]:
```python
# Convert Date column to datetime data type
df['Date'] = pd.to_datetime(df['Date'])
```

In [7]:
```python
# Create year column
df['Year'] = ''
for i, datetime in enumerate(df['Date']):
    df['Year'][i] = datetime.year
```

In [8]:
```python
# Split location column to extract country and launch center from the text
for i, location in enumerate(df['Location']):
    df['Location'][i] = df['Location'][i].split(", ")
```

In [9]:
```python
df.head()
```

Out[9]:

	Organisation	Location	Date	Detail	Rocket_Status	Mission_Status
0	SpaceX	[LC-39A, Kennedy Space Center, Florida, USA]	2020-08-07 05:12:00+00:00	Falcon 9 Block 5 | Starlink V1 L9 & BlackSky	StatusActive	Success
1	CASC	[Site 9401 (SLS-2), Jiuquan Satellite Launch C...	2020-08-06 04:01:00+00:00	Long March 2D | Gaofen-9 04 & Q-SAT	StatusActive	Success
2	SpaceX	[Pad A, Boca Chica, Texas, USA]	2020-08-04 23:57:00+00:00	Starship Prototype | 150 Meter Hop	StatusActive	Success
3	Roscosmos	[Site 200/39, Baikonur Cosmodrome, Kazakhstan]	2020-07-30 21:25:00+00:00	Proton-M/Briz-M | Ekspress-80 & Ekspress-103	StatusActive	Success
4	ULA	[SLC-41, Cape Canaveral AFS, Florida, USA]	2020-07-30 11:50:00+00:00	Atlas V 541 | Perseverance	StatusActive	Success

In [10]:

```
# Create country and launch center columns
df['Country'] = ''
df['Launch_Center'] = ''
for i, location in enumerate(df['Location']):
    df['Country'][i] = location[-1]
    if len(location) > 3:
        df['Launch_Center'][i] = location[1] + ", " + location[2]
    else:
        df['Launch_Center'][i] = location[-2]
```

In [11]:

```
# Remove leading and trailing space from the columns
for i, location in enumerate(df['Country']):
    df['Country'][i] = df['Country'][i].strip()
for i, location in enumerate(df['Launch_Center']):
    df['Launch_Center'][i] = df['Launch_Center'][i].strip()
```

In [12]:

```
df.head()
```

Out[12]:

	Organisation	Location	Date	Detail	Rocket_Status	Mission_Status
0	SpaceX	[LC-39A, Kennedy Space Center, Florida, USA]	2020-08-07 05:12:00+00:00	Falcon 9 Block 5 | Starlink V1 L9 & BlackSky	StatusActive	Success
1	CASC	[Site 9401 (SLS-2), Jiuquan Satellite Launch C...	2020-08-06 04:01:00+00:00	Long March 2D | Gaofen-9 04 & Q-SAT	StatusActive	Success
2	SpaceX	[Pad A, Boca Chica, Texas, USA]	2020-08-04 23:57:00+00:00	Starship Prototype | 150 Meter Hop	StatusActive	Success
3	Roscosmos	[Site 200/39, Baikonur Cosmodrome, Kazakhstan]	2020-07-30 21:25:00+00:00	Proton-M/Briz-M | Ekspress-80 & Ekspress-103	StatusActive	Success
4	ULA	[SLC-41, Cape Canaveral AFS, Florida, USA]	2020-07-30 11:50:00+00:00	Atlas V 541 | Perseverance	StatusActive	Success

In [13]:

```
# Convert all column names into lowercase
df.columns = df.columns.str.lower()
```

In [14]:

```
#Remove unwanted character organisation columns
df['organisation'] = df['organisation'].str.replace('?', '')
```

In [15]:

```
# Copy data for the analysis
space_df = df[['organisation', 'country', 'launch_center', 'year', 'rocket_status',
```

In [16]:

```
space_df.head()
```

Out[16]:

	organisation	country	launch_center	year	rocket_status	mission_status
0	SpaceX	USA	Kennedy Space Center, Florida	2020	StatusActive	Success
1	CASC	China	Jiuquan Satellite Launch Center	2020	StatusActive	Success
2	SpaceX	USA	Boca Chica, Texas	2020	StatusActive	Success
3	Roscosmos	Kazakhstan	Baikonur Cosmodrome	2020	StatusActive	Success
4	ULA	USA	Cape Canaveral AFS, Florida	2020	StatusActive	Success

In [17]:
```
space_df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4324 entries, 0 to 4323
Data columns (total 6 columns):
 #   Column          Non-Null Count  Dtype 
---  ------          --------------  ----- 
 0   organisation    4324 non-null   object
 1   country         4324 non-null   object
 2   launch_center   4324 non-null   object
 3   year            4324 non-null   object
 4   rocket_status   4324 non-null   object
 5   mission_status  4324 non-null   object
dtypes: object(6)
memory usage: 202.8+ KB
```

In [18]:
```
# Check country names
space_df['country'].unique()
```

Out[18]:
```
array(['USA', 'China', 'Kazakhstan', 'Japan', 'Israel', 'New Zealand',
       'Russia', 'Shahrud Missile Test Site', 'France', 'Iran', 'India',
       'New Mexico', 'Yellow Sea', 'North Korea',
       'Pacific Missile Range Facility', 'Pacific Ocean', 'South Korea',
       'Barents Sea', 'Brazil', 'Gran Canaria', 'Kenya', 'Australia'],
      dtype=object)
```

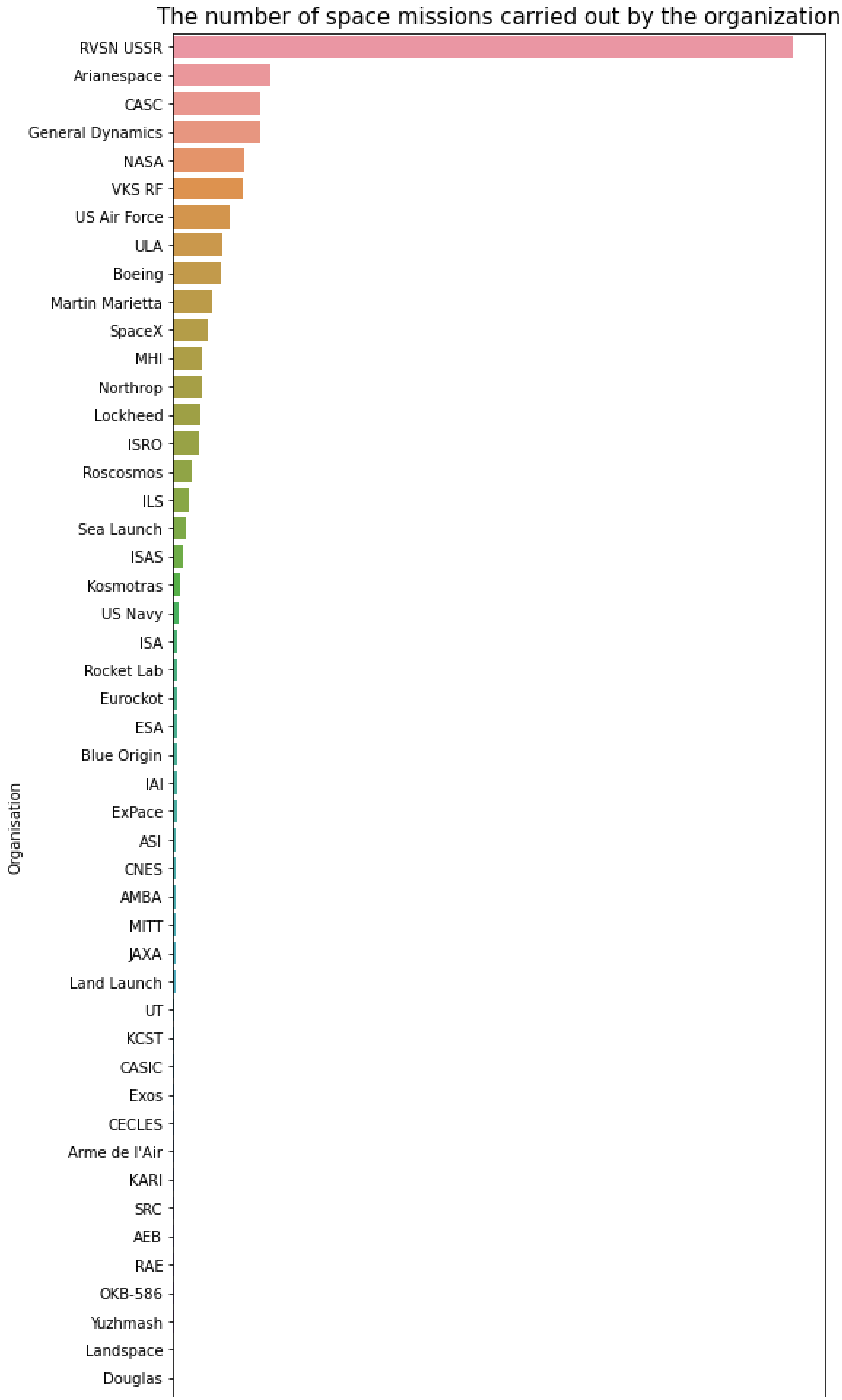
In [19]:
```
# Correct country names and launch center names
for i, country in enumerate(space_df['country']):
    if country == "Pacific Missile Range Facility":
        space_df['launch_center'][i] = space_df['country'][i]
        space_df['country'][i] = 'USA'
    if country == "Shahrud Missile Test Site":
        space_df['launch_center'][i] = space_df['country'][i]
        space_df['country'][i] = 'Iran'
    if country == "New Mexico":
        space_df['country'][i] = 'USA'
```

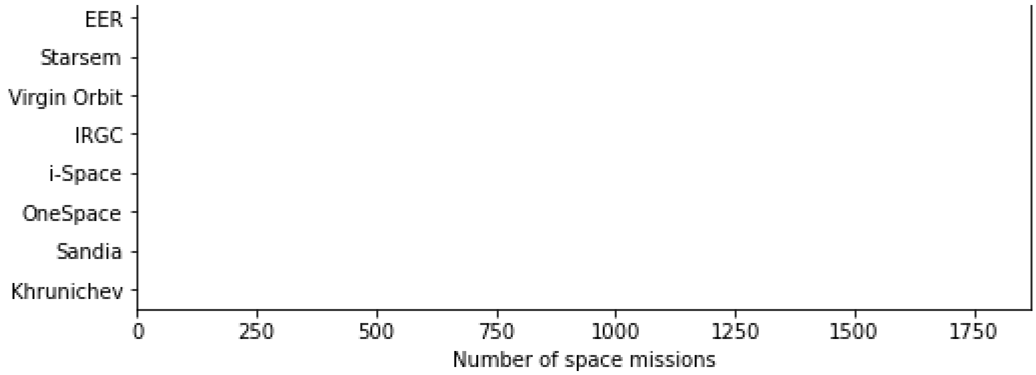
In [20]:
```
space_df['country'].unique()
```

Out[20]:
```
array(['USA', 'China', 'Kazakhstan', 'Japan', 'Israel', 'New Zealand',
       'Russia', 'Iran', 'France', 'India', 'Yellow Sea', 'North Korea',
       'Pacific Ocean', 'South Korea', 'Barents Sea', 'Brazil',
       'Gran Canaria', 'Kenya', 'Australia'], dtype=object)
```

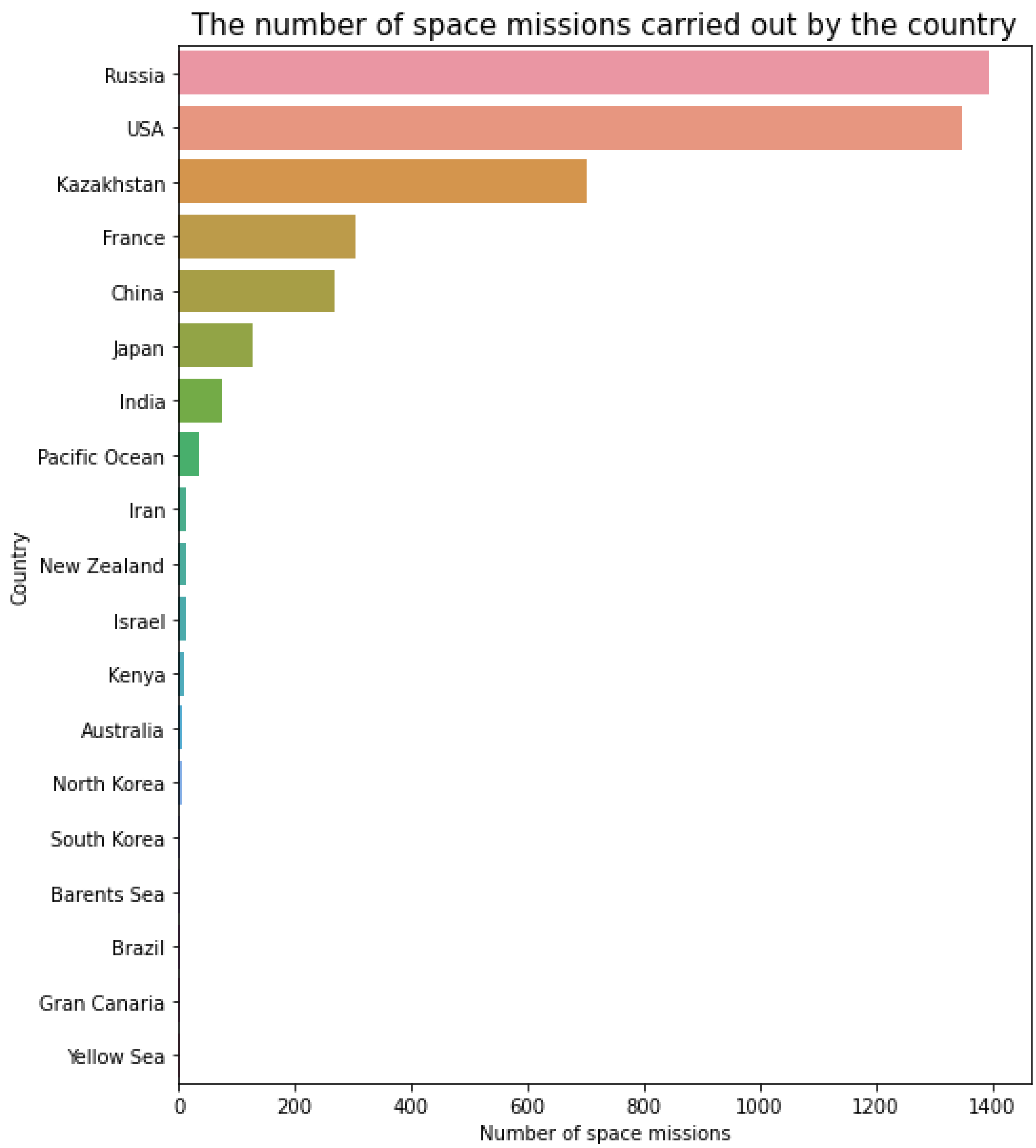
For the analysis, some of the space missions that were conducted in oceans or on islands and did not fall under a specific country were included in the country column.

In [21]:
```
# The number of space missions carried out by the organization
fig = plt.figure(figsize = (8, 20))
sns.countplot(space_df, y = 'organisation', order=space_df['organisation'].value_co
plt.title("The number of space missions carried out by the organization", size = 15
plt.xlabel("Number of space missions")
plt.ylabel("Organisation")
plt.show()
```

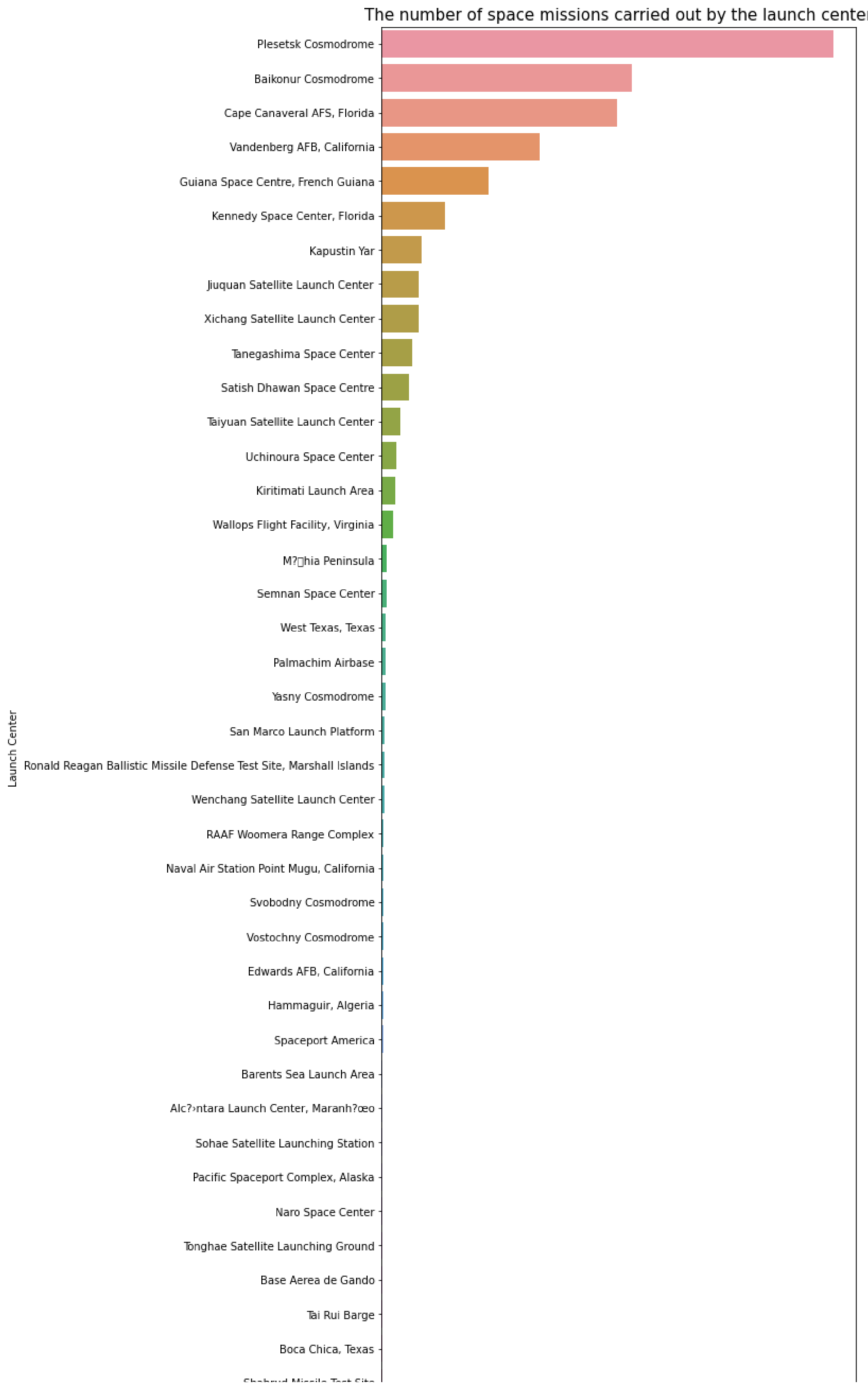

The number of space missions carried out by the organization
Organisation
RVSN USSR
Arianespace
CASC
General Dynamics
NASA
VKS RF
US Air Force
ULA
Boeing
Martin Marietta
SpaceX
MHI
Northrop
Lockheed
ISRO
Roscosmos
ILS
Sea Launch
ISAS
Kosmotras
US Navy
ISA
Rocket Lab
Eurockot
ESA
Blue Origin
IAI
ExPace
ASI
CNES
AMBA
MITT
JAXA
Land Launch
UT
KCST
CASIC
Exos
CECLES
Arme de l'Air
KARI
SRC
AEB
RAE
OKB-586
Yuzhmash
Landspace
Douglas



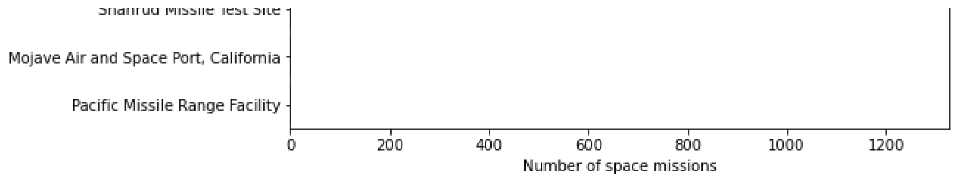
In [22]:
```
# The number of space missions carried out by the country
fig = plt.figure(figsize = (8, 10))
sns.countplot(space_df, y = 'country', order=space_df['country'].value_counts().ind
plt.title("The number of space missions carried out by the country", size = 15)
plt.xlabel("Number of space missions")
plt.ylabel("Country")
plt.show()
```



In [23]:
```
# The number of space missions carried out by the launch center
fig = plt.figure(figsize = (8, 25))
sns.countplot(space_df, y = 'launch_center', order=space_df['launch_center'].value_
plt.title("The number of space missions carried out by the launch center", size = 1
plt.xlabel("Number of space missions")
plt.ylabel("Launch Center")
plt.show()
```

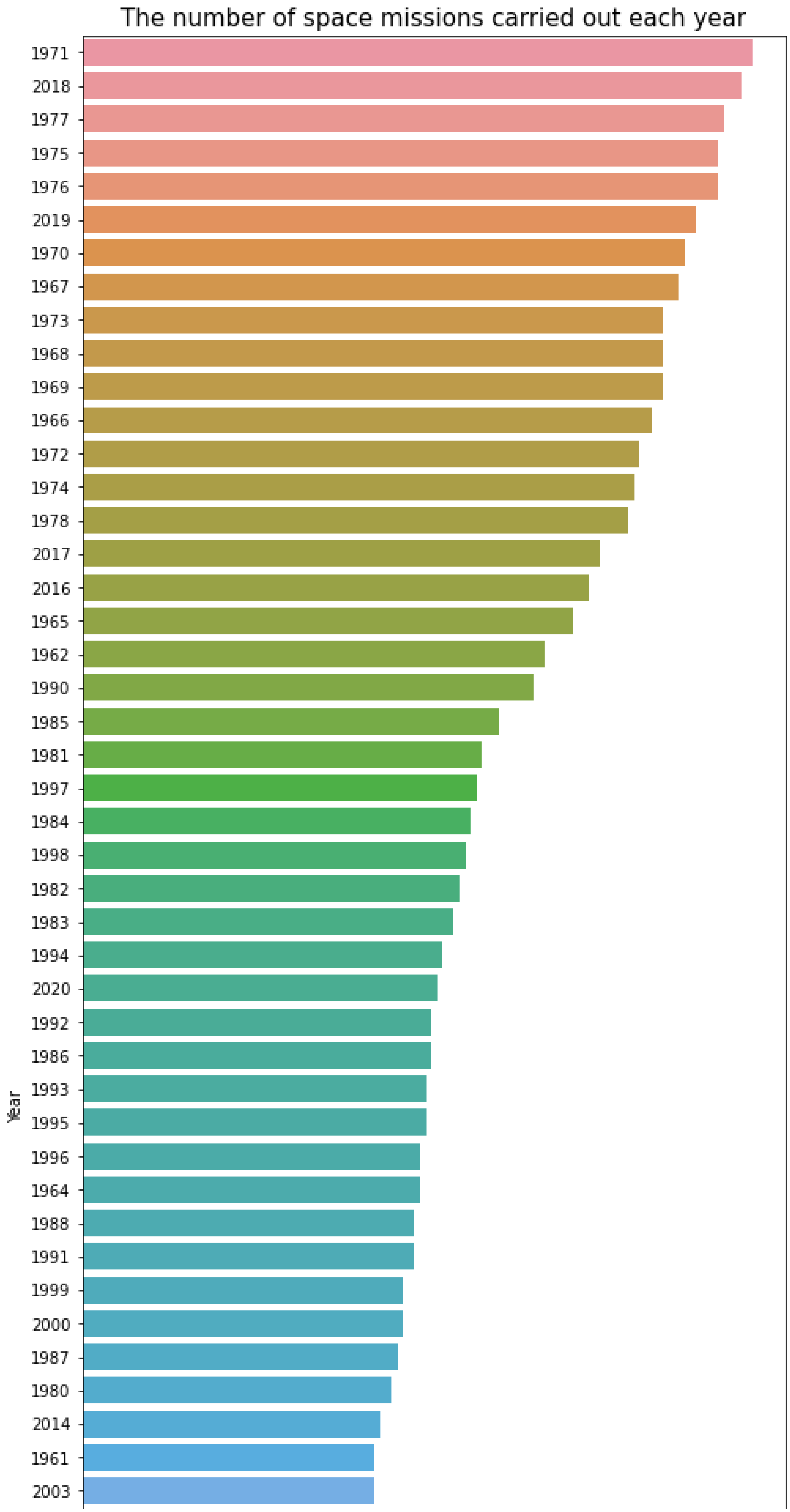


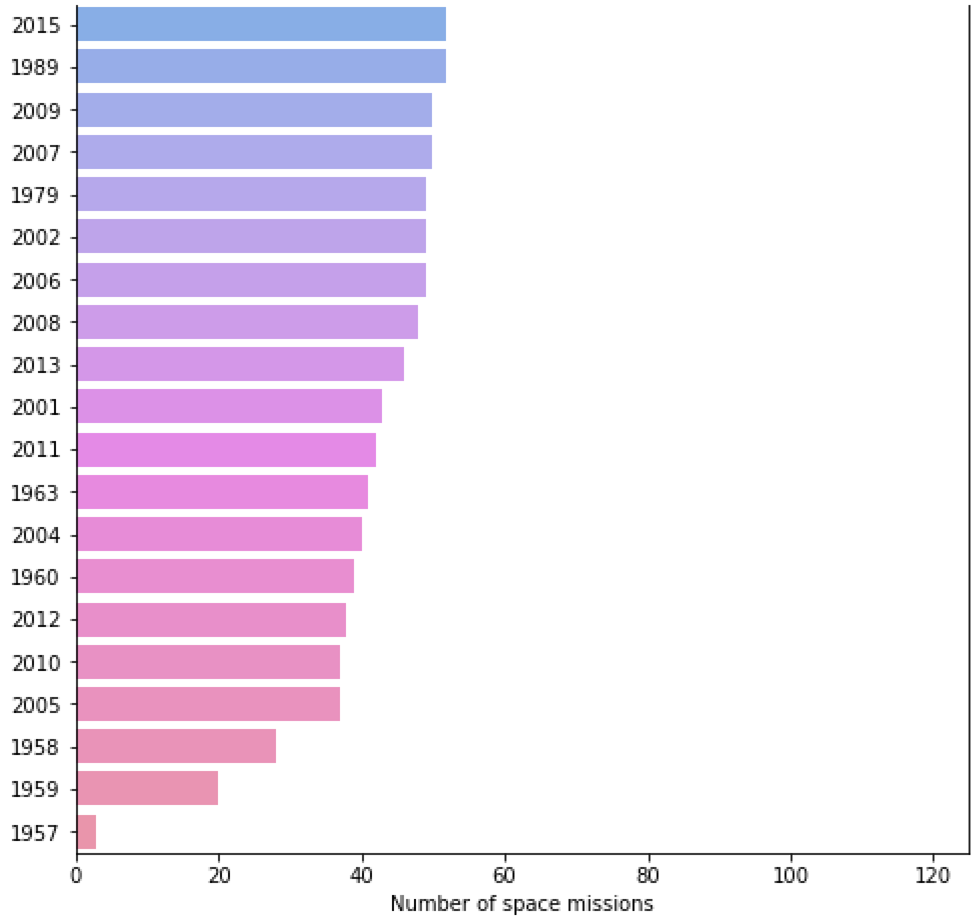
The number of space missions carried out by the launch center
Launch Center
Plesetsk Cosmodrome
Baikonur Cosmodrome
Cape Canaveral AFS, Florida
Vandenberg AFB, California
Guiana Space Centre, French Guiana
Kennedy Space Center, Florida
Kapustin Yar
Jiuquan Satellite Launch Center
Xichang Satellite Launch Center
Tanegashima Space Center
Satish Dhawan Space Centre
Taiyuan Satellite Launch Center
Uchinoura Space Center
Kiritimati Launch Area
Wallops Flight Facility, Virginia
M?hia Peninsula
Semnan Space Center
West Texas, Texas
Palmachim Airbase
Yasny Cosmodrome
San Marco Launch Platform
Ronald Reagan Ballistic Missile Defense Test Site, Marshall Islands
Wenchang Satellite Launch Center
RAAF Woomera Range Complex
Naval Air Station Point Mugu, California
Svobodny Cosmodrome
Vostochny Cosmodrome
Edwards AFB, California
Hammaguir, Algeria
Spaceport America
Barents Sea Launch Area
Alc?›ntara Launch Center, Maranh?œo
Sohae Satellite Launching Station
Pacific Spaceport Complex, Alaska
Naro Space Center
Tonghae Satellite Launching Ground
Base Aerea de Gando
Tai Rui Barge
Boca Chica, Texas



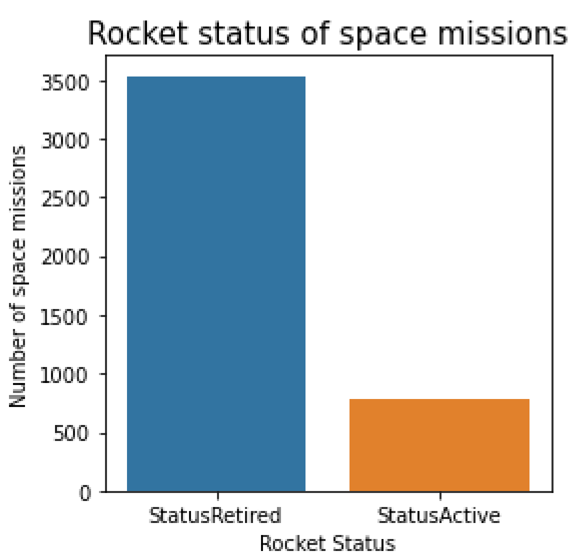
In [24]:

```
#The number of space missions carried out each year
fig = plt.figure(figsize = (8, 25))
sns.countplot(space_df, y = 'year', order=space_df['year'].value_counts().index)
plt.title("The number of space missions carried out each year", size = 15)
plt.xlabel("Number of space missions")
plt.ylabel("Year")
plt.show()
```


The number of space missions carried out each year
Year
1971
2018
1977
1975
1976
2019
1970
1967
1973
1968
1969
1966
1972
1974
1978
2017
2016
1965
1962
1990
1985
1981
1997
1984
1998
1982
1983
1994
2020
1992
1986
1993
1995
1996
1964
1988
1991
1999
2000
1987
1980
2014
1961
2003

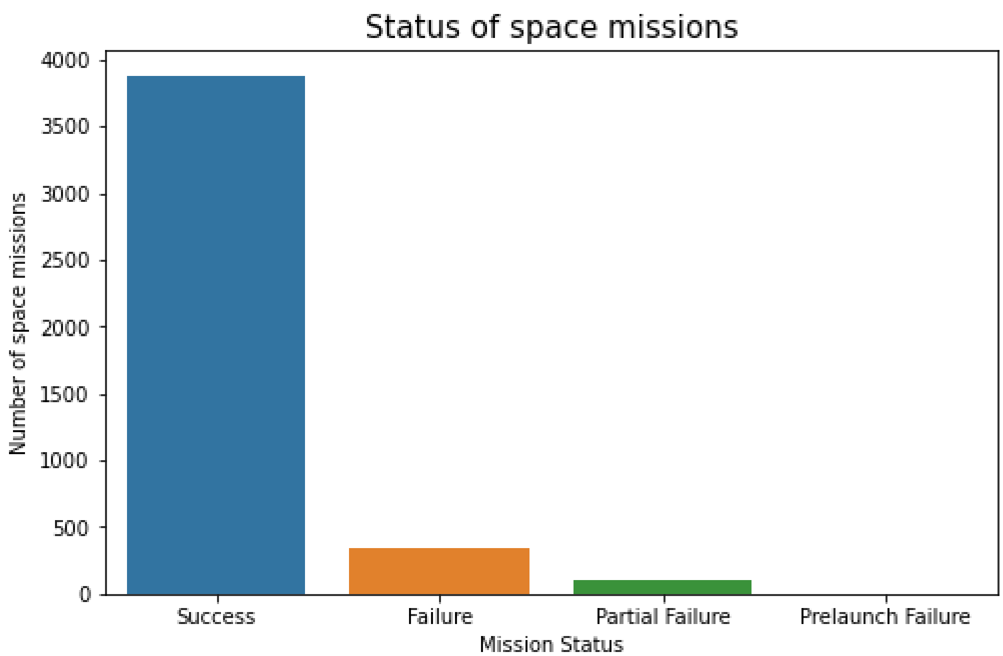


In [25]:
```
# Rocket status of space mission
fig = plt.figure(figsize = (4, 4))
sns.countplot(space_df, x = 'rocket_status', order=space_df['rocket_status'].value_
plt.title("Rocket status of space missions", size = 15)
plt.xlabel("Rocket Status")
plt.ylabel("Number of space missions")
plt.show()
```



In [26]:
```
# Staus of space missions
fig = plt.figure(figsize = (8, 5))
```

```
sns.countplot(space_df, x = 'mission_status', order=space_df['mission_status'].valu
plt.title("Status of space missions", size = 15)
plt.xlabel("Mission Status")
plt.ylabel("Number of space missions")
plt.show()
```



Countries having number of launch centers

In [27]:
```
country_coordinates = pd.read_csv("country_location.csv")
country_coordinates.head()
```

Out[27]:

	Location	Latitude	Longitude	ISO Alpha-3
0	USA	38.8835	-77.0320	USA
1	China	35.8617	104.1954	CHN
2	Kazakhstan	48.0196	66.9237	KAZ
3	Japan	36.2048	138.2529	JPN
4	Israel	31.0461	34.8516	ISR

In [28]:
```
merged_df = pd.merge(space_df, country_coordinates, left_on='country', right_on='Lo
merged_df.head()
```

Out[28]:

	organisation	country	launch_center	year	rocket_status	mission_status	Location	Lat
0	SpaceX	USA	Kennedy Space Center, Florida	2020	StatusActive	Success	USA	38
1	SpaceX	USA	Boca Chica, Texas	2020	StatusActive	Success	USA	38
2	ULA	USA	Cape Canaveral AFS, Florida	2020	StatusActive	Success	USA	38
3	SpaceX	USA	Cape Canaveral AFS, Florida	2020	StatusActive	Success	USA	38
4	Northrop	USA	Wallops Flight Facility, Virginia	2020	StatusActive	Success	USA	38

In [29]:

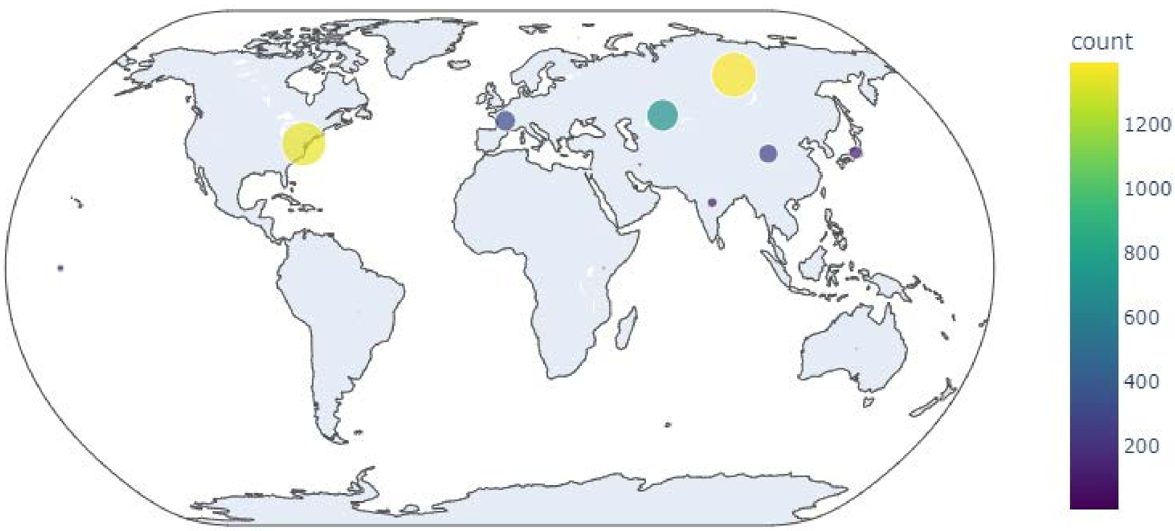
```
grouped_df = merged_df.groupby('country').agg({'Latitude': 'max', 'Longitude': 'max

# include count of countries
grouped_df['count'] = merged_df['country'].value_counts()

# display the resulting data frame
print(grouped_df.reset_index())
```

```
          country  Latitude  Longitude ISO Alpha-3  count
0       Australia  -25.2744   133.7751         AUS      6
1     Barents Sea   75.0000    40.0000         NaN      3
2          Brazil  -14.2350   -51.9253         BRA      3
3           China   35.8617   104.1954         CHN    268
4          France   46.2276     2.2137         FRA    303
5    Gran Canaria   28.1248   -15.4300         ESP      2
6           India   20.5937    78.9629         IND     76
7            Iran   32.4279    53.6880         IRN     14
8          Israel   31.0461    34.8516         ISR     11
9           Japan   36.2048   138.2529         JPN    126
10     Kazakhstan   48.0196    66.9237         KAZ    701
11          Kenya   -0.0236    37.9062         KEN      9
12    New Zealand  -40.9006   174.8860         NZL     13
13    North Korea   40.3399   127.5101         PRK      5
14  Pacific Ocean    0.0000  -160.0000         NaN     36
15         Russia   61.5240   105.3188         RUS   1395
16    South Korea   35.9078   127.7669         KOR      3
17            USA   38.8835   -77.0320         USA   1349
18     Yellow Sea   35.5986   123.9869         NaN      1
```

In [30]:

```
# The number of launch centers in each country|
fig = px.scatter_geo(grouped_df.reset_index(), lat = "Latitude", lon = "Longitude",
fig.show()
```

The number of launch centers in each country



Country-wise analysis

The space missions launched by different countries were analyzed, and their success rates were calculated. This analysis can assist in identifying the countries that are at the forefront of space exploration.

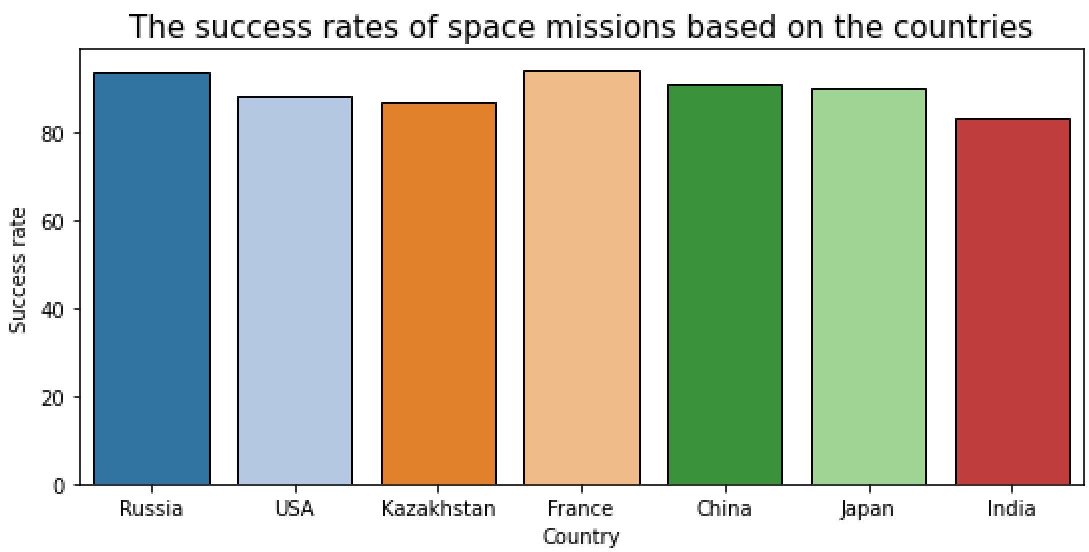
In [31]:
```
country_analysis = pd.pivot_table(space_df, values = ['mission_status'], index = 'c
country_analysis.rename(columns = {'mission_status':'total_missions'}, inplace = Tr
country_analysis['mission_success'] = space_df[space_df['mission_status'] == 'Succe
country_analysis['mission_success'] = country_analysis['mission_success'].fillna(0)
country_analysis['mission_success'] = country_analysis['mission_success'].astype('i
country_analysis['success_rate'] = round(country_analysis['mission_success'] * 100
country_analysis = country_analysis.sort_values('total_missions', ascending = False
country_analysis.style.background_gradient(cmap = 'cubehelix')
```

Out[31]:

country	total_missions	mission_success	success_rate
Russia	1395	1303	93.410000
USA	1349	1186	87.920000
Kazakhstan	701	608	86.730000
France	303	285	94.060000
China	268	243	90.670000
Japan	126	113	89.680000
India	76	63	82.890000
Pacific Ocean	36	33	91.670000
Iran	14	5	35.710000
New Zealand	13	11	84.620000
Israel	11	9	81.820000
Kenya	9	9	100.000000
Australia	6	3	50.000000
North Korea	5	2	40.000000
Barents Sea	3	2	66.670000
South Korea	3	1	33.330000
Brazil	3	0	0.000000
Gran Canaria	2	2	100.000000
Yellow Sea	1	1	100.000000

In [32]:

```
# The success rates of space missions based on the countries
fig = plt.figure(figsize = (9, 4))
sns.barplot(country_analysis.reset_index().iloc[:7], x = 'country', y = 'success_ra
plt.title("The success rates of space missions based on the countries", size = 15)
plt.xlabel("Country")
plt.ylabel("Success rate")
plt.show()
```



The success rates of space missions were analyzed based on the countries displayed in the column chart above, with respect to the number of total missions conducted by each country.

Launch center analysis

The space missions launched from different launch centers were analyzed, and their success rates were calculated. This analysis can assist in identifying the launch centers that are more efficient and successful.

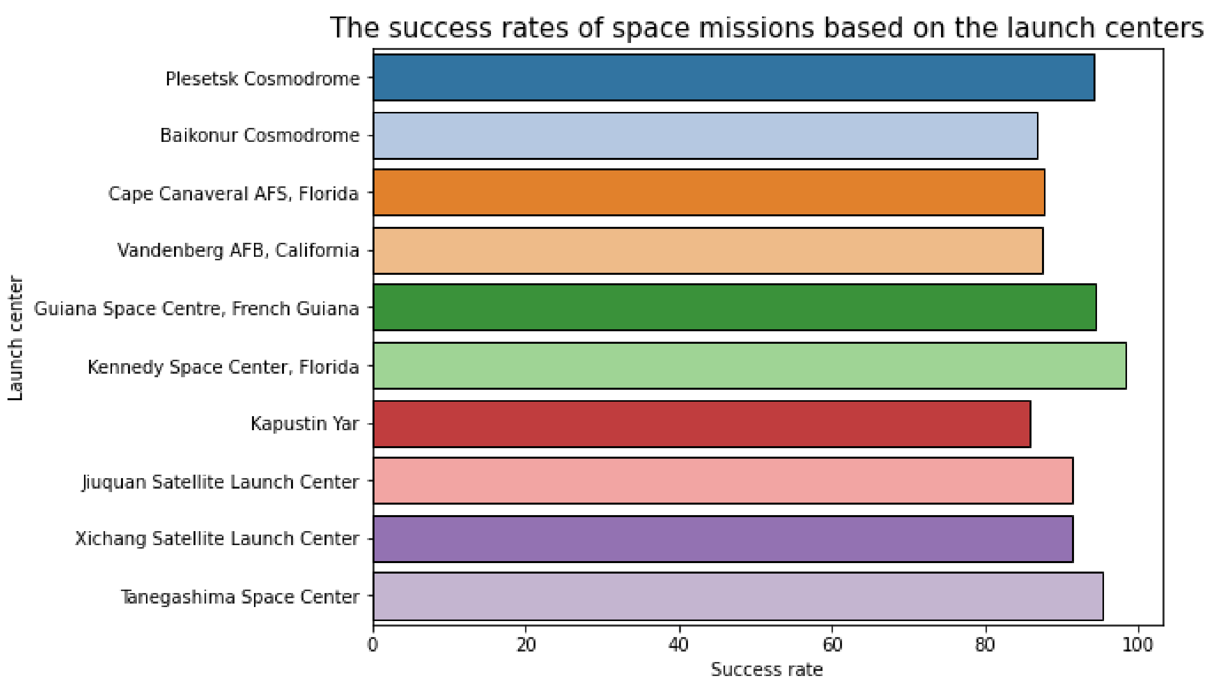
In [33]:
```
launch_center_analysis = pd.pivot_table(space_df, values = ['mission_status'], inde
launch_center_analysis.rename(columns = {'mission_status':'total_missions'}, inplac
launch_center_analysis['mission_success'] = space_df[space_df['mission_status'] ==
launch_center_analysis['mission_success'] = launch_center_analysis['mission_success
launch_center_analysis['mission_success'] = launch_center_analysis['mission_success
launch_center_analysis['success_rate'] = round(launch_center_analysis['mission_succ
launch_center_analysis =launch_center_analysis.sort_values('total_missions', ascend
launch_center_analysis.style.background_gradient(cmap = 'cubehelix')
```

Out[33]:

launch_center	total_missions	mission_success	success_rate
Plesetsk Cosmodrome	1263	1188	94.060000
Baikonur Cosmodrome	701	608	86.730000
Cape Canaveral AFS, Florida	658	577	87.690000
Vandenberg AFB, California	442	386	87.330000
Guiana Space Centre, French Guiana	299	282	94.310000
Kennedy Space Center, Florida	176	173	98.300000
Kapustin Yar	112	96	85.710000
Jiuquan Satellite Launch Center	104	95	91.350000
Xichang Satellite Launch Center	103	94	91.260000
Tanegashima Space Center	85	81	95.290000
Satish Dhawan Space Centre	76	63	82.890000
Taiyuan Satellite Launch Center	53	48	90.570000
Uchinoura Space Center	41	32	78.050000
Kiritimati Launch Area	36	33	91.670000
Wallops Flight Facility, Virginia	31	25	80.650000
M?□hia Peninsula	13	11	84.620000
Semnan Space Center	13	4	30.770000
West Texas, Texas	12	12	100.000000
Palmachim Airbase	11	9	81.820000
Yasny Cosmodrome	10	10	100.000000
San Marco Launch Platform	9	9	100.000000
Ronald Reagan Ballistic Missile Defense Test Site, Marshall Islands	9	6	66.670000
Wenchang Satellite Launch Center	8	6	75.000000
RAAF Woomera Range Complex	6	3	50.000000
Naval Air Station Point Mugu, California	6	0	0.000000
Edwards AFB, California	5	3	60.000000
Svobodny Cosmodrome	5	5	100.000000
Vostochny Cosmodrome	5	4	80.000000
Hammaguir, Algeria	4	3	75.000000
Spaceport America	4	0	0.000000

launch_center	total_missions	mission_success	success_rate
Pacific Spaceport Complex, Alaska	3	3	100.000000
Barents Sea Launch Area	3	2	66.670000
Alc?›ntara Launch Center, Maranh?œo	3	0	0.000000
Naro Space Center	3	1	33.330000
Sohae Satellite Launching Station	3	2	66.670000
Base Aerea de Gando	2	2	100.000000
Tonghae Satellite Launching Ground	2	0	0.000000
Pacific Missile Range Facility	1	0	0.000000
Mojave Air and Space Port, California	1	0	0.000000
Tai Rui Barge	1	1	100.000000
Boca Chica, Texas	1	1	100.000000
Shahrud Missile Test Site	1	1	100.000000

In [34]:
```
# The success rates of space missions based on the launch centers
fig = plt.figure(figsize = (8, 6))
sns.barplot(launch_center_analysis.reset_index().iloc[:10], y = 'launch_center', x
plt.title("The success rates of space missions based on the launch centers", size =
plt.xlabel("Success rate")
plt.ylabel("Launch center")
plt.show()
```



The success rates of space missions were analyzed based on the launch centers displayed in the *bar chart above, with respect to the number of total missions conducted in each launch center.*

Rocket status analysis

The success rates were analyzed based on the status of the rockets used, whether they were active or inactive. This analysis can help in identifying the impact of rocket status on the success of space missions.

In [35]:
```
rocket_status_analysis = space_df[space_df['mission_status'] == 'Success'].groupby(
rocket_status_analysis.rename(columns = {'mission_status':'Success'}, inplace = Tru
rocket_status_analysis['Failure'] = space_df[space_df['mission_status'] == 'Failure
rocket_status_analysis['Partial Failure'] = space_df[space_df['mission_status'] ==
rocket_status_analysis['Prelaunch Failure'] = space_df[space_df['mission_status'] =
rocket_status_analysis['Success Rate'] = rocket_status_analysis['Success'] * 100 /
rocket_status_analysis
```

Out[35]:

	Success	Failure	Partial Failure	Prelaunch Failure	Success Rate
rocket_status					
StatusActive	736	39	13	2	94.480103
StatusRetired	3143	300	89	2	91.180737

Time series analysis

A time series analysis can be performed on the dataset to identify any trends or patterns over time. The number of missions launched each year was analyzed to identify any trends in the frequency of space missions. Additionally, the success rate of space missions each year was also analyzed.

In [36]:
```
time_series_analysis = pd.pivot_table(space_df, values = ['mission_status'], index
time_series_analysis.rename(columns = {'mission_status':'total_missions'}, inplace
time_series_analysis['mission_success'] = space_df[space_df['mission_status'] == 'S
time_series_analysis['mission_success'] = time_series_analysis['mission_success'].f
time_series_analysis['mission_success'] = time_series_analysis['mission_success'].a
time_series_analysis['success_rate'] = round(time_series_analysis['mission_success'
time_series_analysis =time_series_analysis.sort_values('year')
time_series_analysis.style.background_gradient(cmap = 'cubehelix')
```

Out[36]:

year	total_missions	mission_success	success_rate
1957	3	2	66.670000
1958	28	6	21.430000
1959	20	8	40.000000
1960	39	19	48.720000
1961	52	32	61.540000
1962	82	65	79.270000
1963	41	29	70.730000
1964	60	48	80.000000
1965	87	74	85.060000
1966	101	81	80.200000
1967	106	87	82.080000
1968	103	92	89.320000
1969	103	85	82.520000
1970	107	93	86.920000
1971	119	105	88.240000
1972	99	89	89.900000
1973	103	96	93.200000
1974	98	90	91.840000
1975	113	107	94.690000
1976	113	108	95.580000
1977	114	110	96.490000
1978	97	94	96.910000
1979	49	46	93.880000
1980	55	49	89.090000
1981	71	65	91.550000
1982	67	62	92.540000
1983	66	65	98.480000
1984	69	66	95.650000
1985	74	68	91.890000
1986	62	56	90.320000
1987	56	53	94.640000

	total_missions	mission_success	success_rate
year			
1988	59	57	96.610000
1989	52	50	96.150000
1990	80	76	95.000000
1991	59	54	91.530000
1992	62	59	95.160000
1993	61	57	93.440000
1994	64	58	90.620000
1995	61	53	86.890000
1996	60	56	93.330000
1997	70	64	91.430000
1998	68	61	89.710000
1999	57	51	89.470000
2000	57	53	92.980000
2001	43	40	93.020000
2002	49	47	95.920000
2003	52	48	92.310000
2004	40	37	92.500000
2005	37	34	91.890000
2006	49	46	93.880000
2007	50	46	92.000000
2008	48	45	93.750000
2009	50	47	94.000000
2010	37	34	91.890000
2011	42	40	95.240000
2012	38	34	89.470000
2013	46	43	93.480000
2014	53	51	96.230000
2015	52	48	92.310000
2016	90	86	95.560000
2017	92	84	91.300000
2018	117	113	96.580000

	total_missions	mission_success	success_rate
year			
2019	109	100	91.740000
2020	63	57	90.480000

In [37]:
```
# The number of space missions carried out over time
fig, ax = plt.subplots(2, 1, figsize = (8, 8))
sns.lineplot(time_series_analysis.reset_index(), x = 'year', y = 'total_missions',
ax[0].set_xlabel("Year")
ax[0].set_ylabel("Number of space missions")
ax[0].set_title("The number of space missions carried out over time", size = 15)

# The success rate of space missions carried out over time
sns.lineplot(time_series_analysis.reset_index(), x = 'year', y = 'success_rate', ax
ax[1].set_xlabel("Year")
ax[1].set_ylabel("Success rate")
ax[1].set_title("The success rate of space missions carried out over time", size =
fig.tight_layout()
fig.show()
```