Welcome to Statistics Analysis of Russian Soldiers and Equipment lost in Russia Ukraine War

1. Reading and Importing Datasets and Libraries

In [1]:

```
import pandas as pd
import numpy as np
import seaborn as sb
import plotly.express as px
import matplotlib.pyplot as plt
from scipy import stats
```

1.1 Reading "russia_losses_equipment.csv" dataset

In [2]:

```
war=pd.read_csv('russia_losses_equipment.csv')
```

In [3]:

```
war.head()
```

Out[3]:

	date	day	aircraft	helicopter	tank	APC	field artillery	MRL	military auto	fuel tank	drone	naval ship	a airc warl
0	2022-02-25	2	10	7	80	516	49	4	100	60	0	2	
1	2022-02-26	3	27	26	146	706	49	4	130	60	2	2	
2	2022-02-27	4	27	26	150	706	50	4	130	60	2	2	
3	2022-02-28	5	29	29	150	816	74	21	291	60	3	2	
4	2022-03-01	6	29	29	198	846	77	24	305	60	3	2	

In [4]:

```
war.shape
```

Out[4]:

```
(39, 15)
```

In [5]:

```
war.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 39 entries, 0 to 38
Data columns (total 15 columns):
 #   Column                  Non-Null Count  Dtype  
---  ------                  --------------  -----  
 0   date                    39 non-null     object 
 1   day                     39 non-null     int64  
 2   aircraft                39 non-null     int64  
 3   helicopter              39 non-null     int64  
 4   tank                    39 non-null     int64  
 5   APC                     39 non-null     int64  
 6   field artillery         39 non-null     int64  
 7   MRL                     39 non-null     int64  
 8   military auto           39 non-null     int64  
 9   fuel tank               39 non-null     int64  
 10  drone                   39 non-null     int64  
 11  naval ship              39 non-null     int64  
 12  anti-aircraft warfare   39 non-null     int64  
 13  special equipment       20 non-null     float64
 14  mobile SRBM system      10 non-null     float64
dtypes: float64(2), int64(12), object(1)
memory usage: 4.7+ KB
```

1.2 Reading "russia_losses_personnel.csv" Dataset

In [6]:

```
war_person=pd.read_csv('russia_losses_personnel.csv')
```

In [7]:

```
war_person.head()
```

Out[7]:

	date	day	personnel	personnel*	POW
0	2022-02-25	2	2800	about	0
1	2022-02-26	3	4300	about	0
2	2022-02-27	4	4500	about	0
3	2022-02-28	5	5300	about	0
4	2022-03-01	6	5710	about	200

In [8]:

```
war_person.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 39 entries, 0 to 38
Data columns (total 5 columns):
 #   Column      Non-Null Count  Dtype 
---  ------      --------------  ----- 
 0   date        39 non-null     object
 1   day         39 non-null     int64 
 2   personnel   39 non-null     int64 
 3   personnel*  39 non-null     object
 4   POW         39 non-null     int64 
dtypes: int64(3), object(2)
memory usage: 1.6+ KB
```

In [9]:

```
war_person.shape
```

Out[9]:

```
(39, 5)
```

2. Data Cleaning

2.1.1 Analysing Null Values from the "russia_losses_equipment.csv" DataSets

In [10]:

```
war.isnull().sum()
```

Out[10]:

```
date                        0
day                         0
aircraft                    0
helicopter                  0
tank                        0
APC                         0
field artillery             0
MRL                         0
military auto               0
fuel tank                   0
drone                       0
naval ship                  0
anti-aircraft warfare       0
special equipment          19
mobile SRBM system         29
dtype: int64
```

Insights:

From the above query we conclude the following ::

1.There are no null values except "special equipment" and "mobile SRBM system".

2.In "special equipment" column, there are 19 null values in first 19 days which refers that initially the team were unable to get the data. It can be due to war or else Russia don't want to share their special equipments.

3.In "mobile SRBM system" column, there are 29 null values in the first 29 days war which refers the same situation as of the previous one.

2.1.2 Analysing Null Values from the "russia_losses_personnel.csv" DataSets

In [11]:

```
war_person.isnull().sum()
```

Out[11]:

```
date          0
day           0
personnel     0
personnel*    0
POW           0
dtype: int64
```

Insights:

From the above stats we can conclude that this dataset doesn't have any null values which can affect the analysis.

2.2.1 Cleaning the "special equipment" columns of "russia_losses_equipment.csv" DataSets

In [12]:

```
war['special equipment'].unique()
```

Out[12]:

```
array([nan, 10., 11., 12., 13., 15., 16., 18., 19., 21., 22., 24., 25.])
```

In [13]:

```
speci_equip_null=war['special equipment'][0]
speci_equip_null
```

Out[13]:

```
nan
```

In [14]:

```
war['special equipment'].replace(speci_equip_null,0.0,inplace=True)
```

In [15]:

```
war['special equipment'].unique()
```

Out[15]:

```
array([ 0., 10., 11., 12., 13., 15., 16., 18., 19., 21., 22., 24., 25.])
```

2.3.1 Cleaning the "mobile SRBM system" columns of "russia_losses_equipment.csv" DataSets

In [16]:

```
war['mobile SRBM system'].unique()
```

Out[16]:

```
array([nan,  2.,  4.])
```

In [17]:

```
mobile_srbm_null=war['mobile SRBM system'][0]
mobile_srbm_null
```

Out[17]:

```
nan
```

In [18]:

```
war['mobile SRBM system'].replace(mobile_srbm_null,0.0,inplace=True)
```

In [19]:

```
war['mobile SRBM system'].unique()
```

Out[19]:

```
array([0., 2., 4.])
```

3. Data Manipulation Section

Objective:

--> As we can see both the datasets are based on a similar agenda then we can merge them on the basis of their "DATE" and "DAY"columns.

In [20]:

```
varcol=['day','date']
ww3=pd.merge(war_person,war,on=varcol)
```

In [21]:

```
ww3.head()
```

Out[21]:

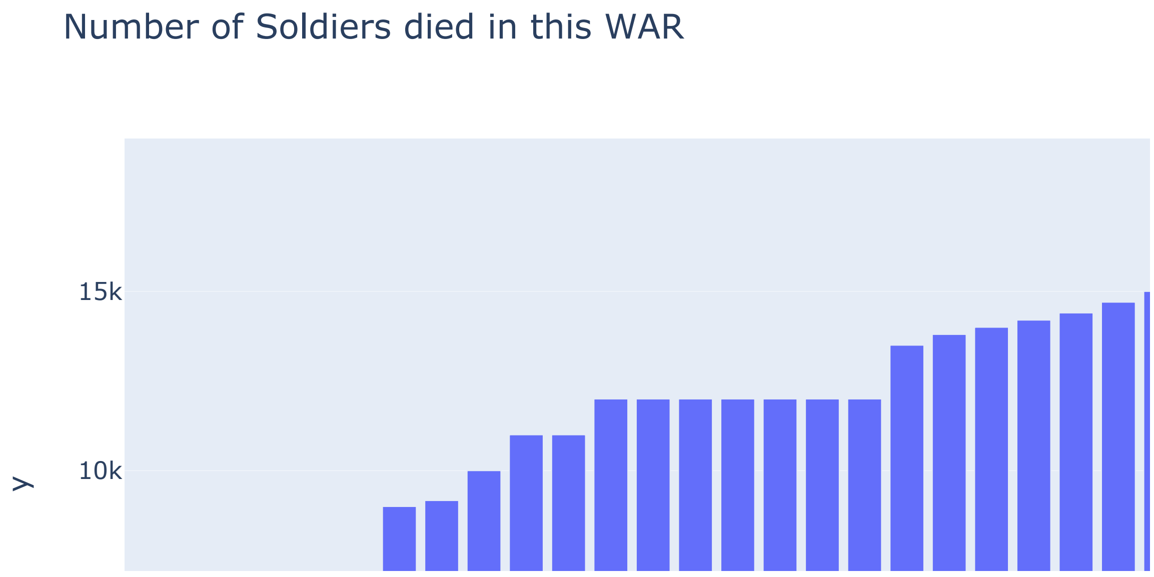
	date	day	personnel	personnel*	POW	aircraft	helicopter	tank	APC	field artillery	MRL	mil
0	2022-02-25	2	2800	about	0	10	7	80	516	49	4	
1	2022-02-26	3	4300	about	0	27	26	146	706	49	4	
2	2022-02-27	4	4500	about	0	27	26	150	706	50	4	
3	2022-02-28	5	5300	about	0	29	29	150	816	74	21	
4	2022-03-01	6	5710	about	200	29	29	198	846	77	24	

4. Data Visualisation

4.1 Taking visuals from 'personnel' column:

In [22]:

```
fig1=px.bar(x=ww3.date,y=ww3.personnel,title='Number of Soldiers died in this WAR')
fig1.show()
```



5. Analytics Section :

In [23]:

```
war.head()
```

Out[23]:

	date	day	aircraft	helicopter	tank	APC	field artillery	MRL	military auto	fuel tank	drone	naval ship	a airc warl
0	2022-02-25	2	10	7	80	516	49	4	100	60	0	2	
1	2022-02-26	3	27	26	146	706	49	4	130	60	2	2	
2	2022-02-27	4	27	26	150	706	50	4	130	60	2	2	
3	2022-02-28	5	29	29	150	816	74	21	291	60	3	2	
4	2022-03-01	6	29	29	198	846	77	24	305	60	3	2	

In [24]:

```
war_person.head()
```

Out[24]:

	date	day	personnel	personnel*	POW
0	2022-02-25	2	2800	about	0
1	2022-02-26	3	4300	about	0
2	2022-02-27	4	4500	about	0
3	2022-02-28	5	5300	about	0
4	2022-03-01	6	5710	about	200

In [25]:

```
ww3.head()
```

Out[25]:

	date	day	personnel	personnel*	POW	aircraft	helicopter	tank	APC	field artillery	MRL	mil
0	2022-02-25	2	2800	about	0	10	7	80	516	49	4	
1	2022-02-26	3	4300	about	0	27	26	146	706	49	4	
2	2022-02-27	4	4500	about	0	27	26	150	706	50	4	
3	2022-02-28	5	5300	about	0	29	29	150	816	74	21	
4	2022-03-01	6	5710	about	200	29	29	198	846	77	24	

5.1 Total Destructions made by Ukraine to Russian Troops:

In [26]:

```
ww3.tail(1)
```

Out[26]:

	date	day	personnel	personnel*	POW	aircraft	helicopter	tank	APC	field artillery	MRL	m
38	2022-04-04	40	18300	about	460	147	134	647	1844	330	107	

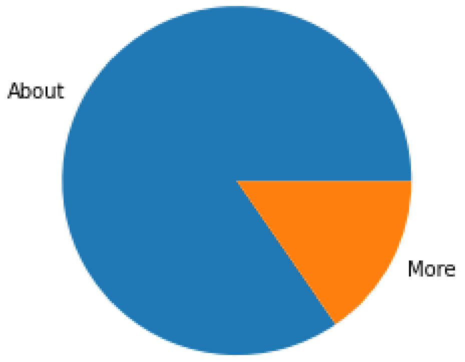
From the above stats,

We can conclude that Russia has lost 18300 personnel, 460 POW, 147 Aircrafts, 647 Tanks, 1844 APC, 330 Field artilery, 107 MRL, 1273 Military Auto, 76 Fuel Tank, 91 Drones, 7 Naval Ships, 54 anti-aircraft warfare, 25.0 special equipment , 4.0 mobile SRBM system

5.2 Analysing the number of Soldiers died

In [27]:

```
# Pie chart which describes the expectation vs assume range
fig2=plt.pie(ww3['personnel*'].value_counts(), labels=['About', 'More'], rotatelabels=False
```



In [28]:

```
dead_person=ww3['personnel'].values
dead_person
```

Out[28]:

```
array([ 2800,  4300,  4500,  5300,  5710,  5840,  9000,  9166, 10000,
       11000, 11000, 12000, 12000, 12000, 12000, 12000, 12000, 12000,
       13500, 13800, 14000, 14200, 14400, 14700, 15000, 15300, 15600,
       15800, 16100, 16400, 16600, 17000, 17200, 17300, 17500, 17700,
       17700, 18000, 18300], dtype=int64)
```

In [29]:

```
summer=[]
summer.append(dead_person[0])
n=len(dead_person)-1
for x in range(0,n-1):
    summer.append(dead_person[x+1]-dead_person[x])
summer.append(0)
len(summer)
```

Out[29]:

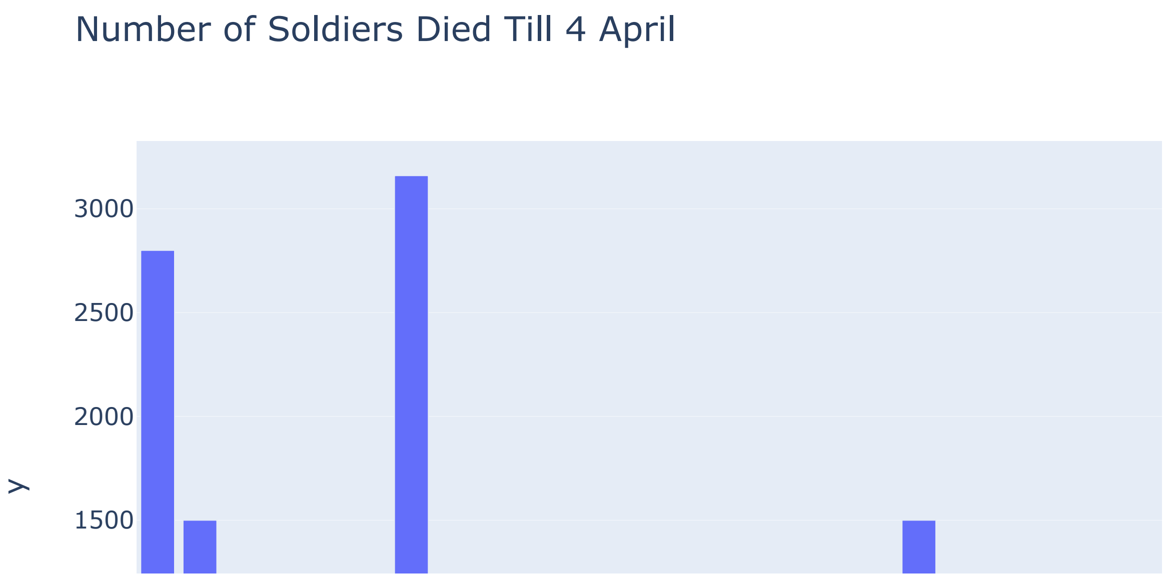
```
39
```

In [30]:

```
ww3_date=ww3['date'].values
```

In [31]:

```
fig3=px.bar(x=ww3_date,y=summer, title="Number of Soldiers Died Till 4 April")
fig3.show()
```



In [32]:

```
sum=0
for x in summer:
    sum=sum+x
excepted_dead_per_day=sum/40
```

5.2.1 Predicting the number of soldier will die if the WAR continues

In [33]:

```
print('This Model can predict the average number of soldiers going to die after 4 April')
a=int(input("Enter the Number of Days to which you want to know :"))
print("The number of soldiers will die is")
print(a*excepted_dead_per_day)
```

```
This Model can predict the average number of soldiers going to die after 4 A
pril
Enter the Number of Days to which you want to know :40
The number of soldiers will die is
18000.0
```

5.3 Analysing the tanks ruined using Machine Learning-Linear Regression

In [34]:

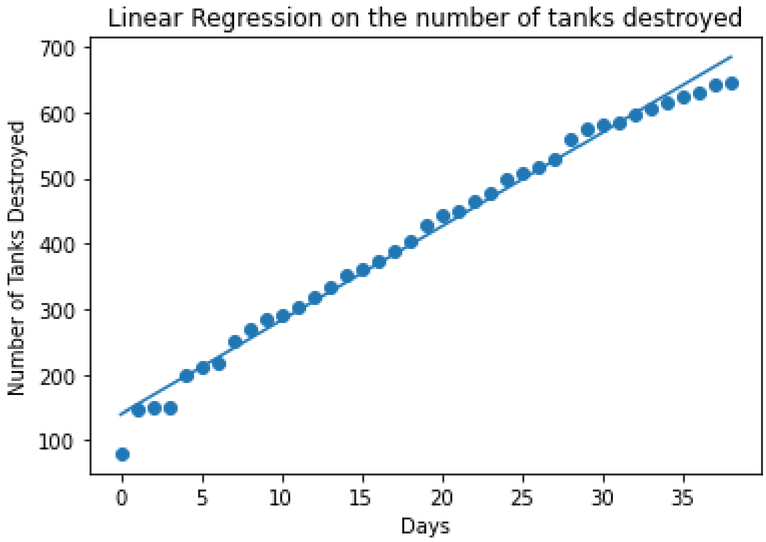
```python
tank_x=ww3['tank'].index.values
tank_y=ww3['tank'].values

slope, intercept, r, p, std_err = stats.linregress(tank_x, tank_y)

def tank_func(x):
  return slope * x + intercept

mymodel = list(map(tank_func, tank_x))

plt.scatter(tank_x, tank_y)
plt.plot(tank_x, mymodel)
plt.xlabel('Days')
plt.ylabel('Number of Tanks Destroyed')
plt.title('Linear Regression on the number of tanks destroyed')
plt.show()
print('The slope of this scatter plot is '+ str(slope))
```



```
The slope of this scatter plot is 14.350404858299596
```

In [35]:

```python
print('Enter the DAY number to which you want to know what number of tanks destroyed')
round(tank_func(int(input('Enter the DAY : '))))
```

```
Enter the DAY number to which you want to know what number of tanks destroye
d
Enter the DAY : 40
```

Out[35]:

```
713
```

Conclusion:

1. As per the stats, if this WAR will continue then, Russia will rapidly losses their soldiers along with Defensive Modules increasingly.

2.The above analysis defines that about 450 soldiers are losses their life in a day in this WAR.

3.The above linear regression represents that there is uplift in the number of tanks destroyed.

WAR CANNOT DEFINES ANYTHING , SO LETS MAKE ONE VOICE TO STOP THE WAR !!!!