Define Problem

Adease is an ads and marketing company helping businesses elicit maximum clicks.

Our objective is to forecast or predict the views of different languages for the wikipedia articles

Import dataset, check structure & characteristics

```
import pandas as pd
import numpy as np
```

```
from google.colab import drive
drive.mount('/content/drive')
```

```
Drive already mounted at /content/drive; to attempt to forcibly remount, call drive.mount("/content/drive", force_remount=True).
```

```
train_1=pd.read_csv('/content/drive/MyDrive/Colab Notebooks/DS & ML/Projects/13. Adease/train_1.csv')
train_1
```

	Page	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	2015-07-07	2015-07-08	2015-07-09	...	2016-12-22	2016-12-23
0	2NE1_zh.wikipedia.org_all-access_spider	18.0	11.0	5.0	13.0	14.0	9.0	9.0	22.0	26.0	...	32.0	63.0
1	2PM_zh.wikipedia.org_all-access_spider	11.0	14.0	15.0	18.0	11.0	13.0	22.0	11.0	10.0	...	17.0	42.0
2	3C_zh.wikipedia.org_all-access_spider	1.0	0.0	1.0	1.0	0.0	4.0	0.0	3.0	4.0	...	3.0	1.0
3	4minute_zh.wikipedia.org_all-access_spider	35.0	13.0	10.0	94.0	4.0	26.0	14.0	9.0	11.0	...	32.0	10.0
4	52_Hz_I_Love_You_zh.wikipedia.org_all-access_s...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	48.0	9.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
145058	Underworld_(serie_de_películas)_es.wikipedia.o...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145059	Resident_Evil:_Capítulo_Final_es.wikipedia.org...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145060	Enamorándome_de_Ramón_es.wikipedia.org_all-acc...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145061	Hasta_el_último_hombre_es.wikipedia.org_all-ac...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145062	Francisco_el_matemático_(serie_de_televisión_d...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN

145063 rows × 551 columns

```
train_1.shape
```

```
(145063, 551)
```

The shape of train data is around 145063 rows and 551 columns

The data set contains data of about 145063 pages and the respective views of that page from date July 1 2015 to Dec 31 2016.

```
train_1.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 145063 entries, 0 to 145062
Columns: 551 entries, Page to 2016-12-31
dtypes: float64(550), object(1)
memory usage: 609.8+ MB
```

The values of the views are in objective type convert to float type

```
train_1.iloc[:,1:]=train_1.iloc[:,1:].astype('float64')
```

```
train_1
```

	Page	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	2015-07-07	2015-07-08	2015-07-09	...	2016-12-22	2016-12-23
0	2NE1_zh.wikipedia.org_all-access_spider	18.0	11.0	5.0	13.0	14.0	9.0	9.0	22.0	26.0	...	32.0	63.0
1	2PM_zh.wikipedia.org_all-access_spider	11.0	14.0	15.0	18.0	11.0	13.0	22.0	11.0	10.0	...	17.0	42.0
2	3C_zh.wikipedia.org_all-access_spider	1.0	0.0	1.0	1.0	0.0	4.0	0.0	3.0	4.0	...	3.0	1.0
3	4minute_zh.wikipedia.org_all-access_spider	35.0	13.0	10.0	94.0	4.0	26.0	14.0	9.0	11.0	...	32.0	10.0
4	52_Hz_I_Love_You_zh.wikipedia.org_all-access_s...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	48.0	9.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
145058	Underworld_(serie_de_películas)_es.wikipedia.o...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145059	Resident_Evil:_Capítulo_Final_es.wikipedia.org...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145060	Enamorándome_de_Ramón_es.wikipedia.org_all-acc...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145061	Hasta_el_último_hombre_es.wikipedia.org_all-ac...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145062	Francisco_el_matemático_(serie_de_televisión_d...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN

145063 rows × 551 columns

```
exog_1=pd.read_csv('/content/drive/MyDrive/Colab Notebooks/DS & ML/Projects/13. Adease/Exog_Campaign_eng')
exog_1
```

	Exog
0	0
1	0
2	0
3	0
4	0
...	...
545	1
546	1
547	1
548	0
549	0

550 rows × 1 columns

```
train_1.isna().sum(axis=1)
```

```
0           0
1           0
2           0
3           0
4         291
         ...
145058    544
145059    550
145060    550
145061    550
145062    550
Length: 145063, dtype: int64
```

no. of null values per each page and max is 550 for last 4 pages of dataset

```
train_1.isna().sum(axis=0),train_1.isna().sum(axis=0).argmax()
```

```
(Page              0
 2015-07-01    20740
 2015-07-02    20816
 2015-07-03    20544
 2015-07-04    20654
               ...
 2016-12-27     3701
 2016-12-28     3822
 2016-12-29    3826
 2016-12-30    3635
 2016-12-31    3465
 Length: 551, dtype: int64, 2)
```

no of null values per each date and max is having with date 2015-07-02

Impute the null values using linear interpolation

```
train_1.iloc[[4]]
```

	Page	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	2015-07-07	2015-07-08	2015-07-09	...	2016-12-22	2016-12-23	2016-12-24	2016-12-25
4	52_Hz_I_Love_You_zh.wikipedia.org_all-access_s...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	48.0	9.0	25.0	13.0

1 rows × 551 columns

Null values indicate either there might not be any views or page might not be in existence or created by then

So impute the values either with 0 or iterpolate

```
# if we eliminate rows even with a single value
```

```
train_1[~(train_1.isna().any(axis=1))]
```

	Page	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	2015-07-07	2015-07-08	2015-07-09	...	2016-12-22	201 12-
0	2NE1_zh.wikipedia.org_all-access_spider	18.0	11.0	5.0	13.0	14.0	9.0	9.0	22.0	26.0	...	32.0	63
1	2PM_zh.wikipedia.org_all-access_spider	11.0	14.0	15.0	18.0	11.0	13.0	22.0	11.0	10.0	...	17.0	42
2	3C_zh.wikipedia.org_all-access_spider	1.0	0.0	1.0	1.0	0.0	4.0	0.0	3.0	4.0	...	3.0	1
3	4minute_zh.wikipedia.org_all-access_spider	35.0	13.0	10.0	94.0	4.0	26.0	14.0	9.0	11.0	...	32.0	10
5	5566_zh.wikipedia.org_all-access_spider	12.0	7.0	4.0	5.0	20.0	8.0	5.0	17.0	24.0	...	16.0	27
...	...	...	...	...	...	...	...	...	...	...	...	...	
144944	Chichén_Itzá_es.wikipedia.org_all-access_spider	8.0	13.0	19.0	14.0	6.0	5.0	10.0	9.0	5.0	...	15.0	18
144945	Fecundación_es.wikipedia.org_all-access_spider	29.0	16.0	6.0	11.0	33.0	4.0	11.0	16.0	10.0	...	8.0	8
144946	Gran_Hermano_VIP_(España)_es.wikipedia.org_all...	4.0	25.0	7.0	11.0	6.0	6.0	16.0	11.0	23.0	...	12.0	299
144947	Modelo_atómico_de_Thomson_es.wikipedia.org_all...	0.0	2.0	6.0	6.0	7.0	5.0	4.0	6.0	7.0	...	13.0	1
144948	Copa_América_2019_es.wikipedia.org_all-access_...	3.0	10.0	41.0	17.0	16.0	14.0	8.0	12.0	4.0	...	8.0	8

117277 rows × 551 columns

```
117277/145063
```

```
0.808455636516548
```

```
117277+27786
```

```
145063
```

20% of data loss

```
import math
```

```
train_1.iloc[1,2]
```

```
14.0
```

```
null_rows=train_1[(train_1.isna().any(axis=1))]
null_rows
```

	Page	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	2015-07-07	2015-07-08	2015-07-09	...	2016-12-22	2016-12-23
4	52_Hz_I_Love_You_zh.wikipedia.org_all-access_s...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	48.0	9.0
6	91Days_zh.wikipedia.org_all-access_spider	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	2.0	7.0
10	ASTRO_zh.wikipedia.org_all-access_spider	NaN	NaN	NaN	NaN	NaN	1.0	1.0	NaN	NaN	...	11.0	38.0
13	AlphaGo_zh.wikipedia.org_all-access_spider	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	14.0	13.0
19	B-PROJECT_zh.wikipedia.org_all-access_spider	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	4.0	26.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
145058	Underworld_(serie_de_películas)_es.wikipedia.o...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145059	Resident_Evil:_Capítulo_Final_es.wikipedia.org...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145060	Enamorándome_de_Ramón_es.wikipedia.org_all-acc...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145061	Hasta_el_último_hombre_es.wikipedia.org_all-ac...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145062	Francisco_el_matemático_(serie_de_televisión_d...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN

27786 rows × 551 columns

Impute the values with interpolation

```
train_1.iloc[:,1:]
```

	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	2015-07-07	2015-07-08	2015-07-09	2015-07-10	...	2016-12-22	2016-12-23	2016-12-24	2016-12-25	2016-12-26	2016-12-27	2016-12-28	2 1
0	18.0	11.0	5.0	13.0	14.0	9.0	9.0	22.0	26.0	24.0	...	32.0	63.0	15.0	26.0	14.0	20.0	22.0	
1	11.0	14.0	15.0	18.0	11.0	13.0	22.0	11.0	10.0	4.0	...	17.0	42.0	28.0	15.0	9.0	30.0	52.0	
2	1.0	0.0	1.0	1.0	0.0	4.0	0.0	3.0	4.0	4.0	...	3.0	1.0	1.0	7.0	4.0	4.0	6.0	
3	35.0	13.0	10.0	94.0	4.0	26.0	14.0	9.0	11.0	16.0	...	32.0	10.0	26.0	27.0	16.0	11.0	17.0	
4	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	48.0	9.0	25.0	13.0	3.0	11.0	27.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
145058	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	13.0	12.0	13.0	
145059	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
145060	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
145061	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
145062	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	

145063 rows × 550 columns

```
##Linear interpolation
```

```
train_1_after_interpolation=train_1.iloc[:,1:].interpolate(method='linear',axis=1,limit_direction='both')
```

```
train_1_after_interpolation
```

	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	2015-07-07	2015-07-08	2015-07-09	2015-07-10	...	2016-12-22	2016-12-23	2016-12-24	2016-12-25	2016-12-26	2016-12-27	2016-12-28	2 1
0	18.0	11.0	5.0	13.0	14.0	9.0	9.0	22.0	26.0	24.0	...	32.0	63.0	15.0	26.0	14.0	20.0	22.0	
1	11.0	14.0	15.0	18.0	11.0	13.0	22.0	11.0	10.0	4.0	...	17.0	42.0	28.0	15.0	9.0	30.0	52.0	
2	1.0	0.0	1.0	1.0	0.0	4.0	0.0	3.0	4.0	4.0	...	3.0	1.0	1.0	7.0	4.0	4.0	6.0	
3	35.0	13.0	10.0	94.0	4.0	26.0	14.0	9.0	11.0	16.0	...	32.0	10.0	26.0	27.0	16.0	11.0	17.0	
4	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	...	48.0	9.0	25.0	13.0	3.0	11.0	27.0	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	
145058	13.0	13.0	13.0	13.0	13.0	13.0	13.0	13.0	13.0	13.0	...	13.0	13.0	13.0	13.0	13.0	12.0	13.0	
145059	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN		NaN	NaN	NaN	NaN	NaN	NaN	NaN	

```
train_1_after_interpolation=train_1[['Page']].join(train_1_after_interpolation)
```

```
train_1_after_interpolation
```

	Page	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	2015-07-07	2015-07-08	2015-07-09	...	2016-12-22	2016-12-23
0	2NE1_zh.wikipedia.org_all-access_spider	18.0	11.0	5.0	13.0	14.0	9.0	9.0	22.0	26.0	...	32.0	63.0
1	2PM_zh.wikipedia.org_all-access_spider	11.0	14.0	15.0	18.0	11.0	13.0	22.0	11.0	10.0	...	17.0	42.0
2	3C_zh.wikipedia.org_all-access_spider	1.0	0.0	1.0	1.0	0.0	4.0	0.0	3.0	4.0	...	3.0	1.0
3	4minute_zh.wikipedia.org_all-access_spider	35.0	13.0	10.0	94.0	4.0	26.0	14.0	9.0	11.0	...	32.0	10.0
4	52_Hz_I_Love_You_zh.wikipedia.org_all-access_s...	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	...	48.0	9.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...
145058	Underworld_(serie_de_películas)_es.wikipedia.o...	13.0	13.0	13.0	13.0	13.0	13.0	13.0	13.0	13.0	...	13.0	13.0
145059	Resident_Evil:_Capítulo_Final_es.wikipedia.org...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145060	Enamorándome_de_Ramón_es.wikipedia.org_all-acc...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145061	Hasta_el_último_hombre_es.wikipedia.org_all-ac...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN
145062	Francisco_el_matemático_(serie_de_televisión_d...	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	NaN	NaN

145063 rows × 551 columns

```
train_1_after_interpolation=train_1_after_interpolation[~(train_1_after_interpolation.isna().any(axis=1))]
```

```
train_1_after_interpolation
```

	Page	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	2015-07-07	2015-07-08	2015-07-09	...	2016-12-22	2016-1
0	2NE1_zh.wikipedia.org_all-access_spider	18.0	11.0	5.0	13.0	14.0	9.0	9.0	22.0	26.0	...	32.0	63.000
1	2PM_zh.wikipedia.org_all-access_spider	11.0	14.0	15.0	18.0	11.0	13.0	22.0	11.0	10.0	...	17.0	42.000
2	3C_zh.wikipedia.org_all-access_spider	1.0	0.0	1.0	1.0	0.0	4.0	0.0	3.0	4.0	...	3.0	1.000
3	4minute_zh.wikipedia.org_all-access_spider	35.0	13.0	10.0	94.0	4.0	26.0	14.0	9.0	11.0	...	32.0	10.000
4	52_Hz_I_Love_You_zh.wikipedia.org_all-access_s...	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	...	48.0	9.000
...	...	...	...	...	...	...	...	...	...	...	...	...	
145054	Skam_(serie_de_televisión)_es.wikipedia.org_al...	26.0	26.0	26.0	26.0	26.0	26.0	26.0	26.0	26.0	...	8.0	9.000
145055	Legión_(serie_de_televisión)_es.wikipedia.org_...	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	...	1.0	2.000
145056	Doble_tentación_es.wikipedia.org_all-access_sp...	11.0	11.0	11.0	11.0	11.0	11.0	11.0	11.0	11.0	...	21.0	24.333
145057	Mi_adorable_maldición_es.wikipedia.org_all-acc...	21.0	21.0	21.0	21.0	21.0	21.0	21.0	21.0	21.0	...	0.0	0.000
145058	Underworld_(serie_de_películas)_es.wikipedia.o...	13.0	13.0	13.0	13.0	13.0	13.0	13.0	13.0	13.0	...	13.0	13.000

144411 rows × 551 columns

```
train_1_after_interpolation=train_1_after_interpolation[~(train_1_after_interpolation.isna().any(axis=1))]
```

```
train_1_after_interpolation
```

	Page	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	2015-07-07	2015-07-08	2015-07-09	...	2016-12-22	2016-1
0	2NE1_zh.wikipedia.org_all-access_spider	18.0	11.0	5.0	13.0	14.0	9.0	9.0	22.0	26.0	...	32.0	63.000
1	2PM_zh.wikipedia.org_all-access_spider	11.0	14.0	15.0	18.0	11.0	13.0	22.0	11.0	10.0	...	17.0	42.000
2	3C_zh.wikipedia.org_all-access_spider	1.0	0.0	1.0	1.0	0.0	4.0	0.0	3.0	4.0	...	3.0	1.000
3	4minute_zh.wikipedia.org_all-access_spider	35.0	13.0	10.0	94.0	4.0	26.0	14.0	9.0	11.0	...	32.0	10.000
4	52_Hz_I_Love_You_zh.wikipedia.org_all-access_s...	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	...	48.0	9.000
...	...	...	...	...	...	...	...	...	...	...	...	...	
145054	Skam_(serie_de_televisión)_es.wikipedia.org_al...	26.0	26.0	26.0	26.0	26.0	26.0	26.0	26.0	26.0	...	8.0	9.000
145055	Legión_(serie_de_televisión)_es.wikipedia.org_...	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	...	1.0	2.000
145056	Doble_tentación_es.wikipedia.org_all-access_sp...	11.0	11.0	11.0	11.0	11.0	11.0	11.0	11.0	11.0	...	21.0	24.333
145057	Mi_adorable_maldición_es.wikipedia.org_all-acc...	21.0	21.0	21.0	21.0	21.0	21.0	21.0	21.0	21.0	...	0.0	0.000
145058	Underworld_(serie_de_películas)_es.wikipedia.o...	13.0	13.0	13.0	13.0	13.0	13.0	13.0	13.0	13.0	...	13.0	13.000

144411 rows × 551 columns

```
144411/145063
```

```
0.9955054011015904
```

As the data constitutes about 99.5% after dropping its ok to move forward

Data Visualization

```
trial_1=train_1_after_interpolation[train_1_after_interpolation['Page'].str.contains('wikipedia.org')]
```

```
def split_1(x):
  array_1=[]
  # y=a.split('wikipedia.org')
  # x=list(map(lambda z: z.strip('.'), a))

  array_1.append(x[0][:-3])
  array_1.append(x[0][-3:].strip('.'))
  list_1=x[1].split('_')
  array_1.append("_".join(list_1[:-1]))
  array_1.append(list_1[-1])
  return "$".join(array_1)
```

```
trial_1.loc[:,'split_1']=trial_1.loc[:,'Page'].apply(lambda x:x.split('wikipedia.org'))
```

```
/usr/local/lib/python3.8/dist-packages/pandas/core/indexing.py:1667: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus
  self.obj[key] = value
```

```
trial_1['array']=trial_1['split_1'].apply(lambda x: split_1(x))
```

```
<ipython-input-31-bbc2803ba9f3>:1: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead
```

```
See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus
  trial_1['array']=trial_1['split_1'].apply(lambda x: split_1(x))
```

```
trial_1['array']
```

```
0                             2NE1_$zh$_all-access$spider
1                              2PM_$zh$_all-access$spider
2                               3C_$zh$_all-access$spider
3                          4minute_$zh$_all-access$spider
4                 52_Hz_I_Love_You_$zh$_all-access$spider
                                ...                     
145054    Skam_(serie_de_televisión)_$es$_all-access$spider
145055    Legión_(serie_de_televisión)_$es$_all-access$s...
145056              Doble_tentación_$es$_all-access$spider
145057          Mi_adorable_maldición_$es$_all-access$spider
145058    Underworld_(serie_de_películas)_$es$_all-acces...
Name: array, Length: 126683, dtype: object
```

```
trial_1[['Title','Language','Access_type','Access_origin']]=trial_1['array'].str.split("$",expand=True)
```

```
/usr/local/lib/python3.8/dist-packages/pandas/core/frame.py:3641: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus
  self[k1] = value[k2]
```

```
trial_1
```

	Page	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	2015-07-07	2015-07-08	2015-07-09	...	2016-12-28	2016-1
0	2NE1_zh.wikipedia.org_all-access_spider	18.0	11.0	5.0	13.0	14.0	9.0	9.0	22.0	26.0	...	22.0	19.000(
1	2PM_zh.wikipedia.org_all-access_spider	11.0	14.0	15.0	18.0	11.0	13.0	22.0	11.0	10.0	...	52.0	45.000(
2	3C_zh.wikipedia.org_all-access_spider	1.0	0.0	1.0	1.0	0.0	4.0	0.0	3.0	4.0	...	6.0	3.000(
3	4minute_zh.wikipedia.org_all-access_spider	35.0	13.0	10.0	94.0	4.0	26.0	14.0	9.0	11.0	...	17.0	19.000(
4	52_Hz_I_Love_You_zh.wikipedia.org_all-access_s...	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	...	27.0	13.000(
...	...	...	...	...	...	...	...	...	...	...	...	...	
145054	Skam_(serie_de_televisión)_es.wikipedia.org_al...	26.0	26.0	26.0	26.0	26.0	26.0	26.0	26.0	26.0	...	13.0	12.000(
145055	Legión_(serie_de_televisión)_es.wikipedia.org_...	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	...	2.0	4.000(
145056	Doble_tentación_es.wikipedia.org_all-access_sp...	11.0	11.0	11.0	11.0	11.0	11.0	11.0	11.0	11.0	...	41.0	44.3333
145057	Mi_adorable_maldición_es.wikipedia.org_all-acc...	21.0	21.0	21.0	21.0	21.0	21.0	21.0	21.0	21.0	...	0.0	0.000(
145058	Underworld_(serie_de_películas)_es.wikipedia.o...	13.0	13.0	13.0	13.0	13.0	13.0	13.0	13.0	13.0	...	13.0	3.000(

126683 rows × 557 columns

```
trial_1.shape
```

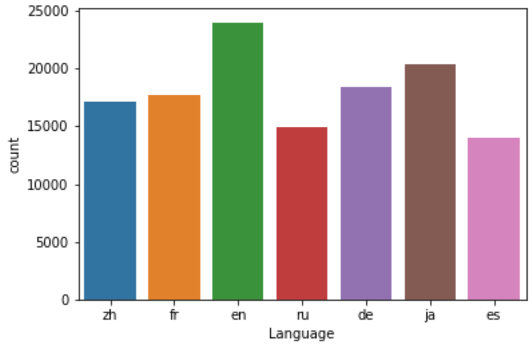
```
(126683, 557)
```

```
trial_1['Language'].value_counts()
```

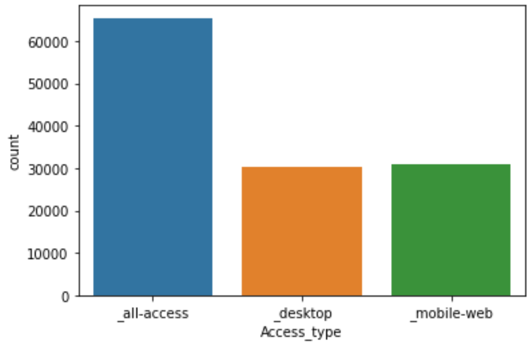
```
en    24010
ja    20340
de    18438
fr    17761
zh    17103
ru    14990
es    14041
Name: Language, dtype: int64
```

```
import seaborn as sns
import matplotlib.pyplot as plt
```

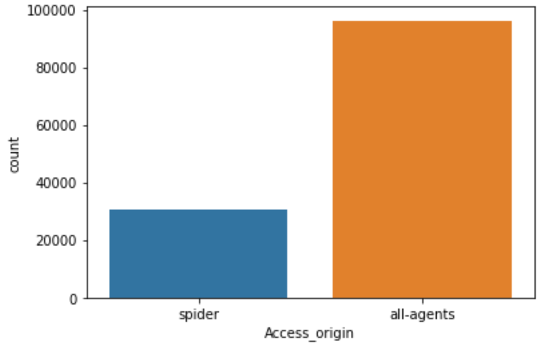
```
sns.countplot(x=trial_1['Language'])
plt.show()
```



```
sns.countplot(x=trial_1['Access_type'])
plt.show()
```



```
sns.countplot(x=trial_1['Access_origin'])
plt.show()
```



```
# top 10 days where the visits are more
trial_1.columns
```

```
Index(['Page', '2015-07-01', '2015-07-02', '2015-07-03', '2015-07-04',
       '2015-07-05', '2015-07-06', '2015-07-07', '2015-07-08', '2015-07-09',
       ...
       '2016-12-28', '2016-12-29', '2016-12-30', '2016-12-31', 'split_1',
       'array', 'Title', 'Language', 'Access_type', 'Access_origin'],
      dtype='object', length=557)
```

```
trial_1.iloc[:,1:551].sum(axis=0).nlargest(10)
```

```
2016-08-15    3.210200e+08
2016-07-26    3.122488e+08
2016-07-25    3.118267e+08
2016-08-10    3.084869e+08
2016-08-14    3.084316e+08
2016-08-01    3.083562e+08
2016-07-27    3.059005e+08
2016-08-12    3.045786e+08
2016-08-08    3.035776e+08
2016-11-09    3.033059e+08
dtype: float64
```

Those were the dates with highest number of visits

```
trial_2=trial_1.iloc[:,[0,551,552,553,554,555,556]].join(trial_1.iloc[:,1:551])
trial_2
```

	Page	split_1	array	
0	2NE1_zh.wikipedia.org_all-access_spider	[2NE1_zh., _all-access_spider]	2NE1_$zh$_all-access$spider	
1	2PM_zh.wikipedia.org_all-access_spider	[2PM_zh., _all-access_spider]	2PM_$zh$_all-access$spider	
2	3C_zh.wikipedia.org_all-access_spider	[3C_zh., _all-access_spider]	3C_$zh$_all-access$spider	
3	4minute_zh.wikipedia.org_all-access_spider	[4minute_zh., _all-access_spider]	4minute_$zh$_all-access$spider	
4	52_Hz_I_Love_You_zh.wikipedia.org_all-access_s...	[52_Hz_I_Love_You_zh., _all-access_spider]	52_Hz_I_Love_You_$zh$_all-access$spider	
...	...	...	...	
145054	Skam_(serie_de_televisión)_es.wikipedia.org_al...	[Skam_(serie_de_televisión)_es., _all-access_s...	Skam_(serie_de_televisión)_$es$_all-access$spider	Skam_
145055	Legión_(serie_de_televisión)_es.wikipedia.org_...	[Legión_(serie_de_televisión)_es., _all-access...	Legión_(serie_de_televisión)_$es$_all-access$s...	Legión_
145056	Doble_tentación_es.wikipedia.org_all-access_sp...	[Doble_tentación_es., _all-access_spider]	Doble_tentación_$es$_all-access$spider	
145057	Mi_adorable_maldición_es.wikipedia.org_all-acc...	[Mi_adorable_maldición_es., _all-access_spider]	Mi_adorable_maldición_$es$_all-access$spider	M
145058	Underworld_(serie_de_películas)_es.wikipedia.o...	[Underworld_(serie_de_películas)_es., _all-acc...	Underworld_(serie_de_películas)_$es$_all-acces...	Underworld

126683 rows × 557 columns

```
##language wise highest number of visits on which day
trial_3=trial_2.iloc[:,3:]
trial_3
```

	Title	Language	Access_type	Access_origin	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	...	2016-12-22	2(
0	2NE1_	zh	_all-access	spider	18.0	11.0	5.0	13.0	14.0	9.0	...	32.0	63
1	2PM_	zh	_all-access	spider	11.0	14.0	15.0	18.0	11.0	13.0	...	17.0	42
2	3C_	zh	_all-access	spider	1.0	0.0	1.0	1.0	0.0	4.0	...	3.0	1
3	4minute_	zh	_all-access	spider	35.0	13.0	10.0	94.0	4.0	26.0	...	32.0	10
4	52_Hz_I_Love_You_	zh	_all-access	spider	38.0	38.0	38.0	38.0	38.0	38.0	...	48.0	9
...	...	...	...	...	...	...	...	...	...	...	...	...	
145054	Skam_(serie_de_televisión)_	es	_all-access	spider	26.0	26.0	26.0	26.0	26.0	26.0	...	8.0	9
145055	Legión_(serie_de_televisión)_	es	_all-access	spider	50.0	50.0	50.0	50.0	50.0	50.0	...	1.0	2
145056	Doble_tentación_	es	_all-access	spider	11.0	11.0	11.0	11.0	11.0	11.0	...	21.0	24
145057	Mi_adorable_maldición_	es	_all-access	spider	21.0	21.0	21.0	21.0	21.0	21.0	...	0.0	0
145058	Underworld_(serie_de_películas)_	es	_all-access	spider	13.0	13.0	13.0	13.0	13.0	13.0	...	13.0	13

126683 rows × 554 columns

```
# ([trial_3.columns[1]])+(trial_3.columns[4:]).tolist()
```

```
trial_4_columns=([trial_3.columns[1]])+(trial_3.columns[4:]).tolist()
```

```
trial_4=trial_3[trial_4_columns]
trial_4
```

	Language	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	2015-07-07	2015-07-08	2015-07-09	...	2016-12-22	2016-12-23	2016-12-24	2016-12-25	2016-12-26
0	zh	18.0	11.0	5.0	13.0	14.0	9.0	9.0	22.0	26.0	...	32.0	63.000000	15.000000	26.0	14.000000
1	zh	11.0	14.0	15.0	18.0	11.0	13.0	22.0	11.0	10.0	...	17.0	42.000000	28.000000	15.0	9.000000
2	zh	1.0	0.0	1.0	1.0	0.0	4.0	0.0	3.0	4.0	...	3.0	1.000000	1.000000	7.0	4.000000
3	zh	35.0	13.0	10.0	94.0	4.0	26.0	14.0	9.0	11.0	...	32.0	10.000000	26.000000	27.0	16.000000
4	zh	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	38.0	...	48.0	9.000000	25.000000	13.0	3.000000
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
145054	es	26.0	26.0	26.0	26.0	26.0	26.0	26.0	26.0	26.0	...	8.0	9.000000	9.000000	19.0	17.000000
145055	es	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	50.0	...	1.0	2.000000	1.000000	1.0	3.000000
145056	es	11.0	11.0	11.0	11.0	11.0	11.0	11.0	11.0	11.0	...	21.0	24.333333	27.666667	31.0	34.333333
145057	es	21.0	21.0	21.0	21.0	21.0	21.0	21.0	21.0	21.0	...	0.0	0.000000	0.000000	0.0	0.000000

```
trial_5=trial_4.groupby(['Language']).agg(lambda x: x.sum()).T.reset_index().set_index('index')
```

```
trial_5
```

Language	de	en	es	fr	ja	ru	zh
index							
2015-07-01	1.507832e+07	9.415409e+07	1.618992e+07	9.232359e+06	1.549925e+07	1.170350e+07	5.159275e+06
2015-07-02	1.489782e+07	9.387991e+07	1.551257e+07	9.286675e+06	1.725639e+07	1.186729e+07	5.165336e+06
2015-07-03	1.437188e+07	8.960875e+07	1.433925e+07	8.959763e+06	1.594099e+07	1.116309e+07	5.137253e+06
2015-07-04	1.333821e+07	9.290429e+07	1.351702e+07	9.521535e+06	1.909167e+07	1.063287e+07	5.176900e+06
2015-07-05	1.521015e+07	9.563976e+07	1.462066e+07	9.362161e+06	1.846234e+07	1.117435e+07	5.454677e+06
...	...	...	...	...	...	...	...
2016-12-27	2.032261e+07	1.458706e+08	1.594582e+07	1.528222e+07	1.626896e+07	1.520168e+07	6.487991e+06
2016-12-28	1.934974e+07	1.415205e+08	1.657789e+07	1.378210e+07	1.629641e+07	1.416161e+07	6.522969e+06
2016-12-29	1.864423e+07	1.507996e+08	1.564768e+07	1.340043e+07	1.782839e+07	1.364024e+07	6.051296e+06
2016-12-30	1.780216e+07	1.256468e+08	1.156067e+07	1.247502e+07	1.959575e+07	1.222803e+07	6.117870e+06
2016-12-31	1.675861e+07	1.238632e+08	1.107802e+07	1.150493e+07	2.460054e+07	1.338433e+07	6.305259e+06

550 rows × 7 columns

```
trial_5.idxmax(axis=0),trial_5.max(axis=0)
```

```
(Language
 de    2015-12-07
 en    2016-07-26
 es    2016-11-09
 fr    2016-04-24
 ja    2016-01-11
 ru    2016-07-28
 zh    2016-01-16
 dtype: object, Language
 de    2.510500e+07
 en    2.050938e+08
 es    3.045691e+07
 fr    2.000300e+07
 ja    3.289058e+07
 ru    4.523846e+07
 zh    1.175250e+07
 dtype: float64)
```

The above are the dates for each language on which the views are highest respectively

```
trial_3
```

	Title	Language	Access_type	Access_origin	2015-07-01	2015-07-02	2015-07-03	2015-07-04	2015-07-05	2015-07-06	...	2016-12-22	20
0	2NE1_	zh	_all-access	spider	18.0	11.0	5.0	13.0	14.0	9.0	...	32.0	63
1	2PM_	zh	_all-access	spider	11.0	14.0	15.0	18.0	11.0	13.0	...	17.0	42
2	3C_	zh	_all-access	spider	1.0	0.0	1.0	1.0	0.0	4.0	...	3.0	1
3	4minute_	zh	_all-access	spider	35.0	13.0	10.0	94.0	4.0	26.0	...	32.0	10
4	52_Hz_I_Love_You_	zh	_all-access	spider	38.0	38.0	38.0	38.0	38.0	38.0	...	48.0	9
...	...	...	...	...	...	...	...	...	...	...	...	...	
145054	Skam_(serie_de_televisión)_	es	_all-access	spider	26.0	26.0	26.0	26.0	26.0	26.0	...	8.0	9
145055	Legión_(serie_de_televisión)_	es	_all-access	spider	50.0	50.0	50.0	50.0	50.0	50.0	...	1.0	2
145056	Doble_tentación_	es	_all-access	spider	11.0	11.0	11.0	11.0	11.0	11.0	...	21.0	24
[illegible]	Underworld_(serie_de_películas)_	es	_all-access	spider	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	...	[illegible]	[illegible]

```
trial_3.groupby(['Language','Access_type']).agg(lambda x: x.sum() if x.dtype=='float64' else x.head(1)).drop(columns=['Tit
```

```
Language  Access_type
de        _all-access    4.827087e+09
          _desktop       2.277197e+09
          _mobile-web    2.413740e+09
en        _all-access    3.133542e+10
          _desktop       1.833471e+10
          _mobile-web    1.238175e+10
es        _all-access    4.964843e+09
          _desktop       2.285070e+09
          _mobile-web    2.630231e+09
fr        _all-access    3.345472e+09
          _desktop       1.586711e+09
          _mobile-web    1.648578e+09
ja        _all-access    4.493121e+09
          _desktop       2.724364e+09
          _mobile-web    2.646978e+09
ru        _all-access    4.398165e+09
          _desktop       2.760635e+09
          _mobile-web    1.586912e+09
zh        _all-access    1.778901e+09
          _desktop       1.047478e+09
          _mobile-web    6.672629e+08
dtype: float64
```

The above are the views for every langauge and accesse type

```
trial_6=trial_3.groupby(['Language','Access_type']).agg(lambda x: x.sum() if x.dtype=='float64' else x.head(1)).drop(columns=['Title','A
```

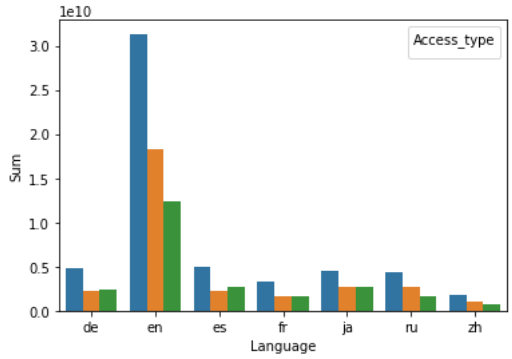
```
trial_6.columns=['Language','Access_type','Sum']
```

```
trial_6
```

	Language	Access_type	Sum
0	de	_all-access	4.827087e+09
1	de	_desktop	2.277197e+09
2	de	_mobile-web	2.413740e+09
3	en	_all-access	3.133542e+10
4	en	_desktop	1.833471e+10
5	en	mobile-web	1.238175e+10

```
plt.plot(figsize=(20,10))
sns.barplot(data=trial_6,x='Language',y='Sum',hue='Access_type')
plt.show()
```

```
WARNING:matplotlib.legend:No handles with labels found to put in legend.
```



English language is having major visits and as per the access type

checking stationarity

```
trial_5
```

Language	de	en	es	fr	ja	ru	zh
index							
2015-07-01	1.507832e+07	9.415409e+07	1.618992e+07	9.232359e+06	1.549925e+07	1.170350e+07	5.159275e+06
2015-07-02	1.489782e+07	9.387991e+07	1.551257e+07	9.286675e+06	1.725639e+07	1.186729e+07	5.165336e+06
2015-07-03	1.437188e+07	8.960875e+07	1.433925e+07	8.959763e+06	1.594099e+07	1.116309e+07	5.137253e+06
2015-07-04	1.333821e+07	9.290429e+07	1.351702e+07	9.521535e+06	1.909167e+07	1.063287e+07	5.176900e+06
2015-07-05	1.521015e+07	9.563976e+07	1.462066e+07	9.362161e+06	1.846234e+07	1.117435e+07	5.454677e+06
...	...	...	...	...	...	...	...
2016-12-27	2.032261e+07	1.458706e+08	1.594582e+07	1.528222e+07	1.626896e+07	1.520168e+07	6.487991e+06
2016-12-28	1.934974e+07	1.415205e+08	1.657789e+07	1.378210e+07	1.629641e+07	1.416161e+07	6.522969e+06
2016-12-29	1.864423e+07	1.507996e+08	1.564768e+07	1.340043e+07	1.782839e+07	1.364024e+07	6.051296e+06
2016-12-30	1.780216e+07	1.256468e+08	1.156067e+07	1.247502e+07	1.959575e+07	1.222803e+07	6.117870e+06
2016-12-31	1.675861e+07	1.238632e+08	1.107802e+07	1.150493e+07	2.460054e+07	1.338433e+07	6.305259e+06

550 rows × 7 columns

```
trial_5.columns=['de', 'en', 'es', 'fr', 'ja', 'ru', 'zh']
```

```
trial_5.reset_index(inplace=True)
```

```
trial_5.columns=['date','de', 'en', 'es', 'fr', 'ja', 'ru', 'zh']
```

```
trial_5.set_index('date',inplace=True)
```

```
trial_5
```

	de	en	es	fr	ja	ru	zh
date							
2015-07-01	1.507832e+07	9.415409e+07	1.618992e+07	9.232359e+06	1.549925e+07	1.170350e+07	5.159275e+06
2015-07-02	1.489782e+07	9.387991e+07	1.551257e+07	9.286675e+06	1.725639e+07	1.186729e+07	5.165336e+06
2015-07-03	1.437188e+07	8.960875e+07	1.433925e+07	8.959763e+06	1.594099e+07	1.116309e+07	5.137253e+06
2015-07-04	1.333821e+07	9.290429e+07	1.351702e+07	9.521535e+06	1.909167e+07	1.063287e+07	5.176900e+06
2015-07-05	1.521015e+07	9.563976e+07	1.462066e+07	9.362161e+06	1.846234e+07	1.117435e+07	5.454677e+06
...	...	...	...	...	...	...	...
2016-12-27	2.032261e+07	1.458706e+08	1.594582e+07	1.528222e+07	1.626896e+07	1.520168e+07	6.487991e+06
2016-12-28	1.934974e+07	1.415205e+08	1.657789e+07	1.378210e+07	1.629641e+07	1.416161e+07	6.522969e+06
2016-12-29	1.864423e+07	1.507996e+08	1.564768e+07	1.340043e+07	1.782839e+07	1.364024e+07	6.051296e+06
2016-12-30	1.780216e+07	1.256468e+08	1.156067e+07	1.247502e+07	1.959575e+07	1.222803e+07	6.117870e+06
2016-12-31	1.675861e+07	1.238632e+08	1.107802e+07	1.150493e+07	2.460054e+07	1.338433e+07	6.305259e+06

```
trial_5.de
```

```
date
2015-07-01    1.507832e+07
2015-07-02    1.489782e+07
2015-07-03    1.437188e+07
2015-07-04    1.333821e+07
2015-07-05    1.521015e+07
                  ...
2016-12-27    2.032261e+07
2016-12-28    1.934974e+07
2016-12-29    1.864423e+07
2016-12-30    1.780216e+07
2016-12-31    1.675861e+07
Name: de, Length: 550, dtype: float64
```

```
trial_5.index=pd.to_datetime(trial_5.index)
```

```
trial_5
```

	de	en	es	fr	ja	ru	zh
date							
2015-07-01	1.507832e+07	9.415409e+07	1.618992e+07	9.232359e+06	1.549925e+07	1.170350e+07	5.159275e+06
2015-07-02	1.489782e+07	9.387991e+07	1.551257e+07	9.286675e+06	1.725639e+07	1.186729e+07	5.165336e+06
2015-07-03	1.437188e+07	8.960875e+07	1.433925e+07	8.959763e+06	1.594099e+07	1.116309e+07	5.137253e+06
2015-07-04	1.333821e+07	9.290429e+07	1.351702e+07	9.521535e+06	1.909167e+07	1.063287e+07	5.176900e+06
2015-07-05	1.521015e+07	9.563976e+07	1.462066e+07	9.362161e+06	1.846234e+07	1.117435e+07	5.454677e+06
...	...	...	...	...	...	...	...
2016-12-27	2.032261e+07	1.458706e+08	1.594582e+07	1.528222e+07	1.626896e+07	1.520168e+07	6.487991e+06
2016-12-28	1.934974e+07	1.415205e+08	1.657789e+07	1.378210e+07	1.629641e+07	1.416161e+07	6.522969e+06
2016-12-29	1.864423e+07	1.507996e+08	1.564768e+07	1.340043e+07	1.782839e+07	1.364024e+07	6.051296e+06
2016-12-30	1.780216e+07	1.256468e+08	1.156067e+07	1.247502e+07	1.959575e+07	1.222803e+07	6.117870e+06
2016-12-31	1.675861e+07	1.238632e+08	1.107802e+07	1.150493e+07	2.460054e+07	1.338433e+07	6.305259e+06

550 rows × 7 columns

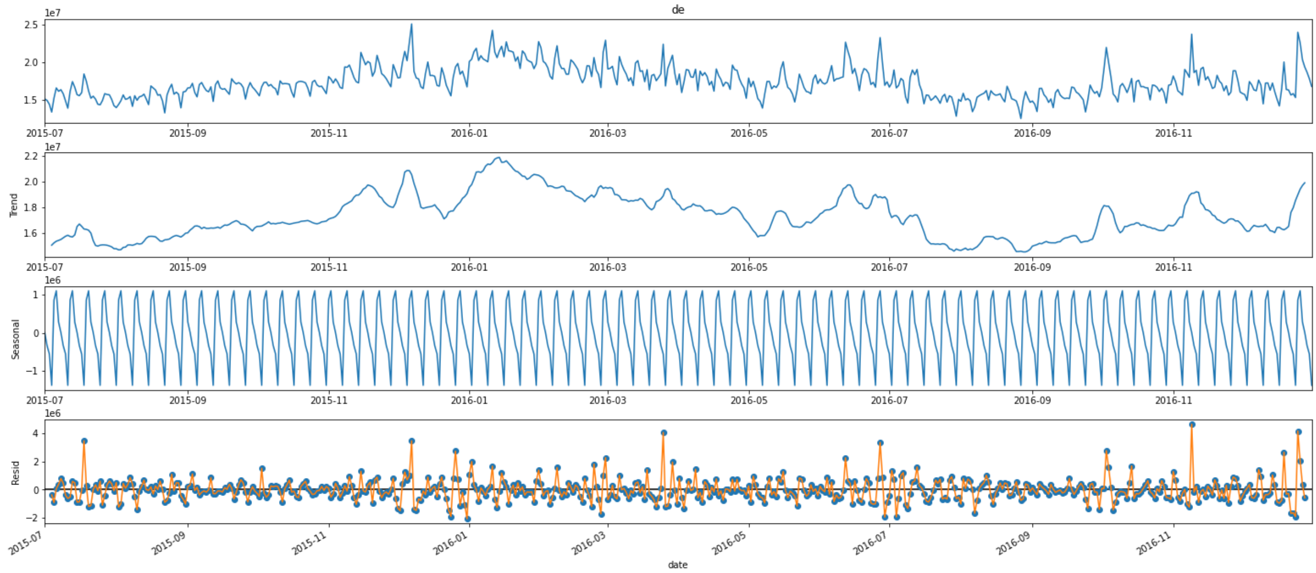
```
import statsmodels.api as sm

model = sm.tsa.seasonal_decompose(trial_5['de'], model='additive')
```

```
plt.rcParams['figure.figsize'] = (20, 10)
```

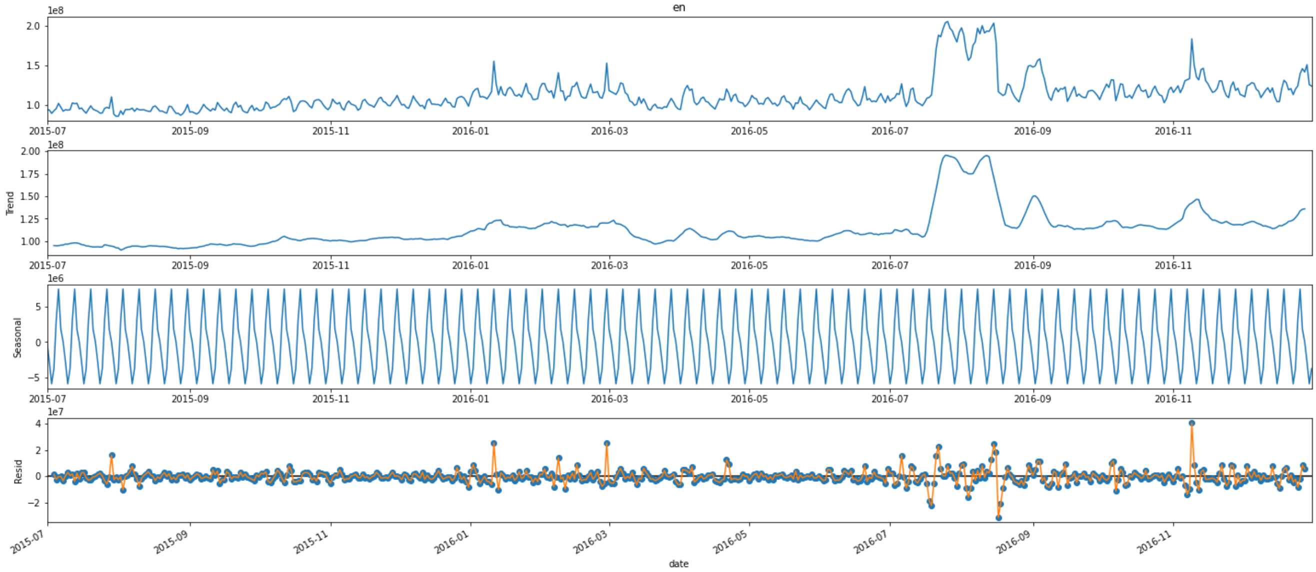
```
# dickey fuller test and decomposition
```

```
for i in trial_5.columns:
  model = sm.tsa.seasonal_decompose(trial_5[i], model='additive')

  model.plot()
  model.resid.plot()
```

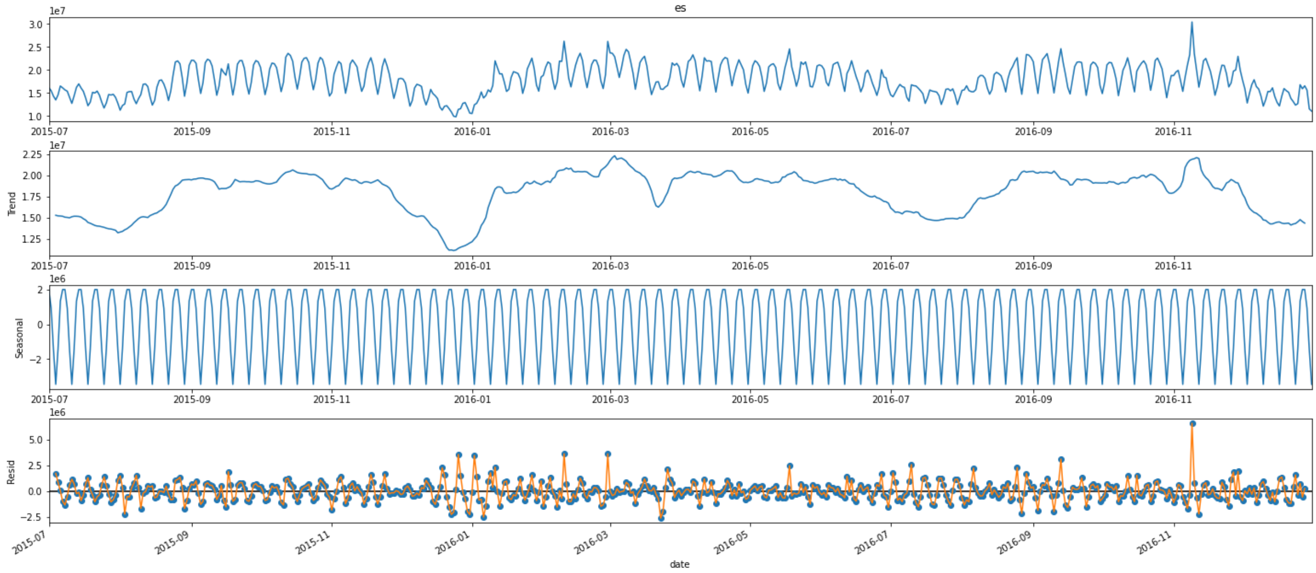
```
plt.show()
pvalue = sm.tsa.stattools.adfuller(model.resid.dropna())[1]
if pvalue <= 0.05:
  print(f'the time series residual for the language {i} is stationary')
else:
  print(f'the time series residual for the language {i} is notstationary')
```



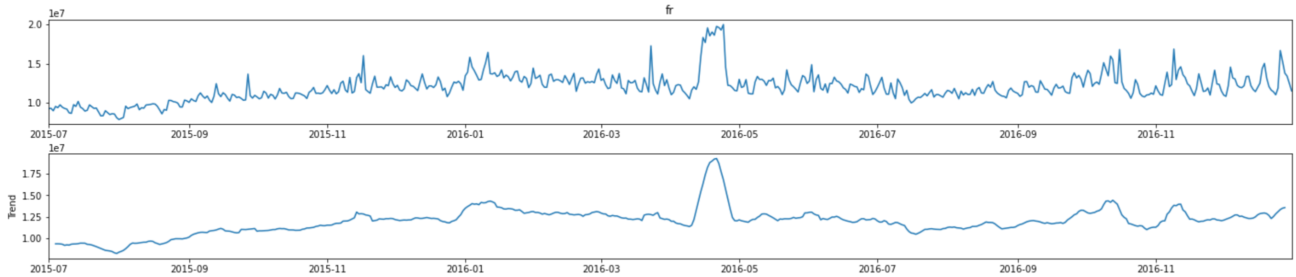
```
the time series residual for the language de is stationary
```



```
the time series residual for the language en is stationary
```

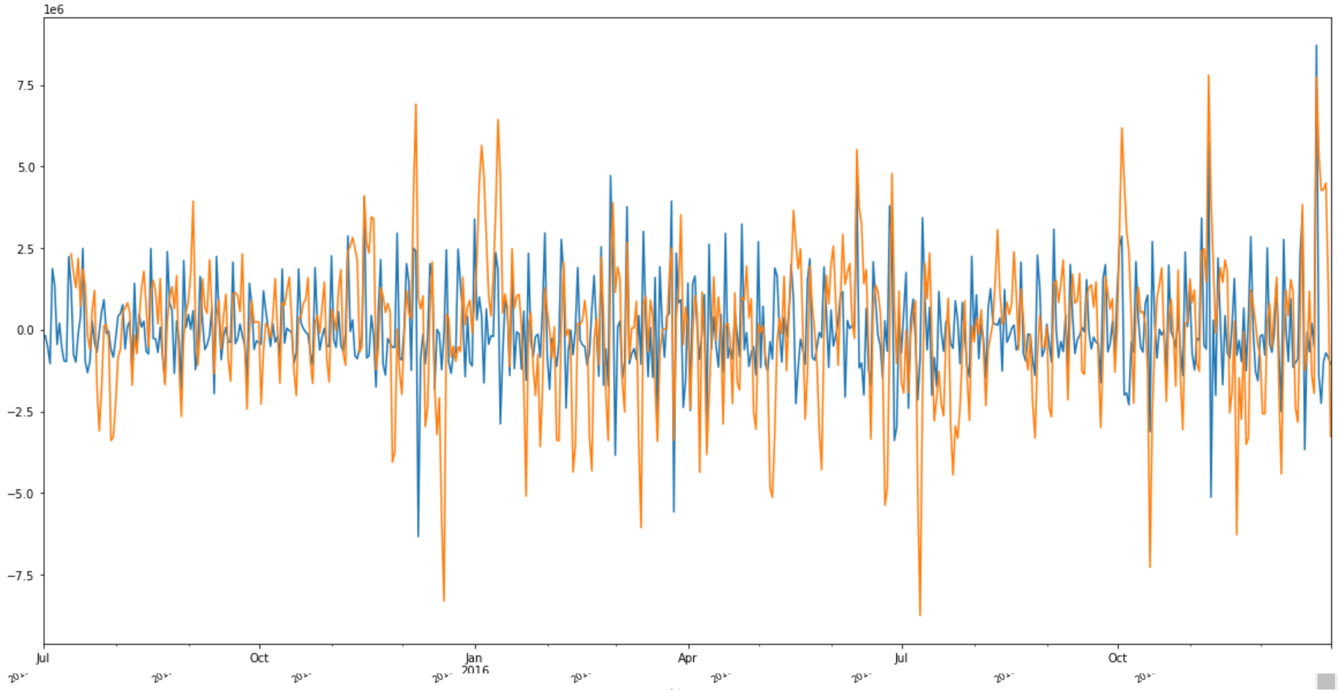


```
the time series residual for the language es is stationary
```



```
trial_5['de'].diff().plot()
trial_5['de'].diff(12).plot()
```

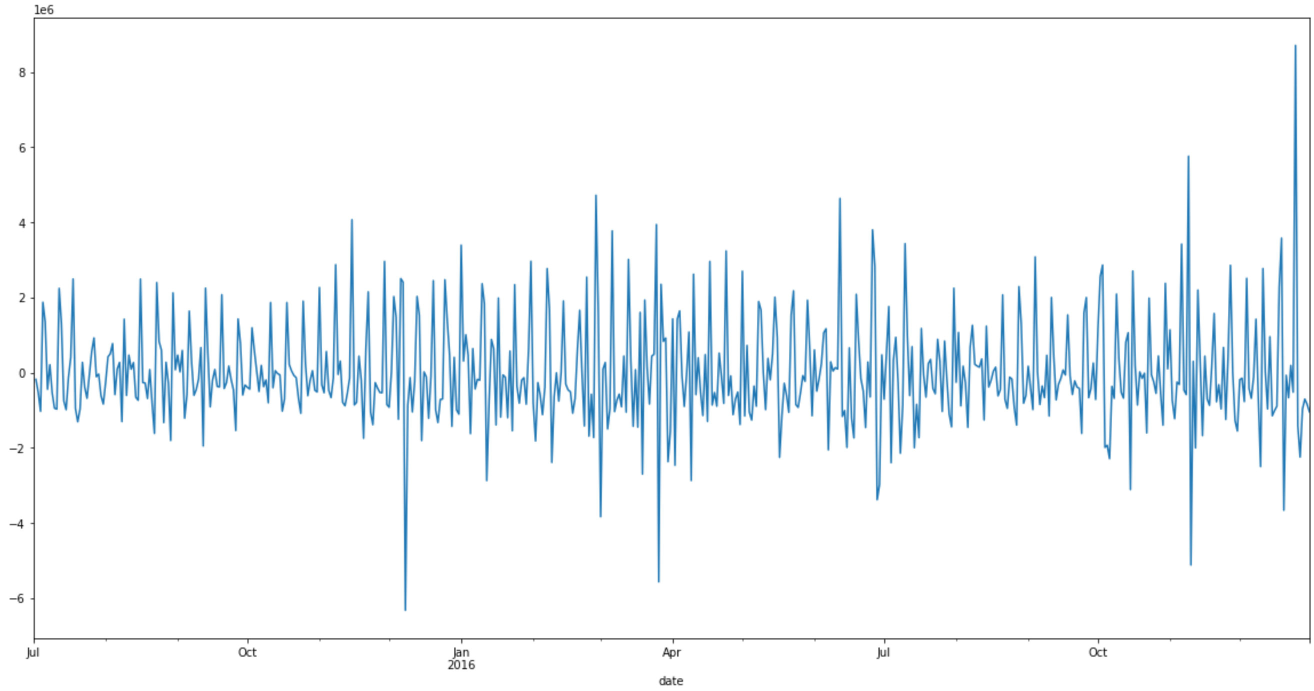
```
<matplotlib.axes._subplots.AxesSubplot at 0x7f17e21f51c0>
```



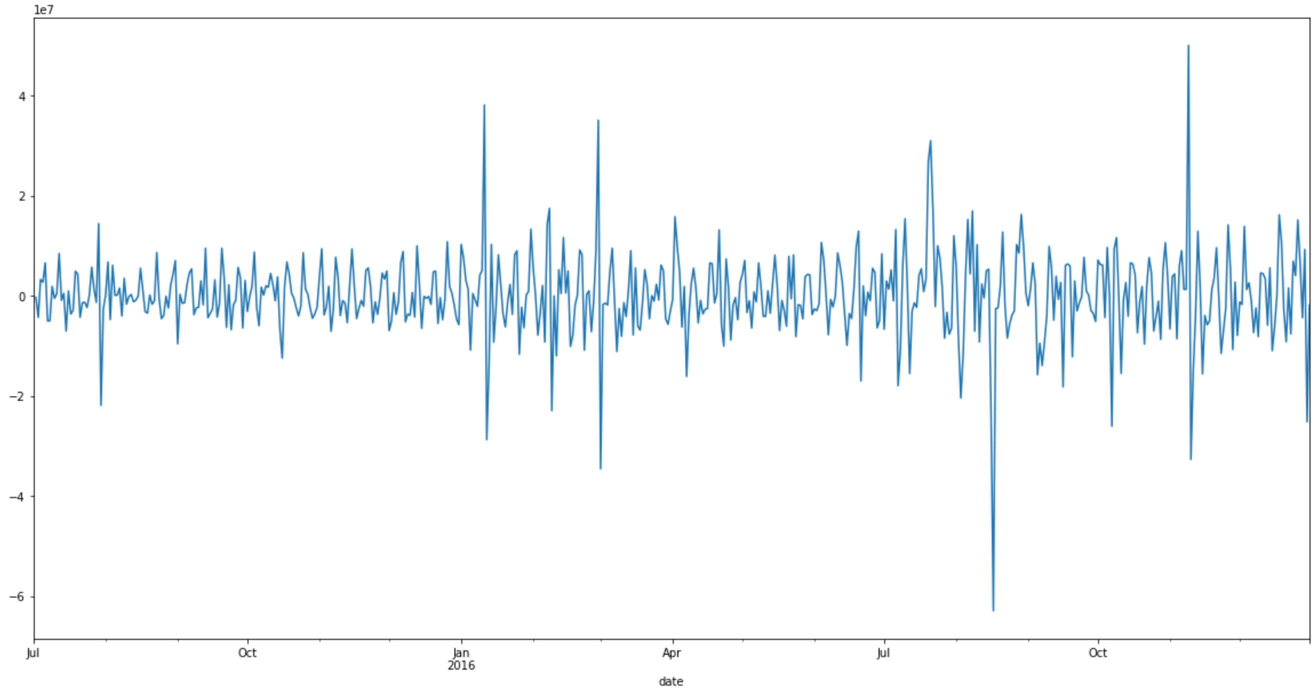
```
trial_5['de'].diff(),trial_5['de'].diff().diff()
```

```
(date
 2015-07-01             NaN
 2015-07-02   -1.805018e+05
 2015-07-03   -5.259441e+05
 2015-07-04   -1.033667e+06
 2015-07-05    1.871935e+06
                   ...
 2016-12-27   -2.251612e+06
 2016-12-28   -9.728757e+05
 2016-12-29   -7.055054e+05
 2016-12-30   -8.420762e+05
 2016-12-31   -1.043543e+06
 Name: de, Length: 550, dtype: float64, date
 2015-07-01             NaN
 2015-07-02             NaN
 2015-07-03   -3.454423e+05
 2015-07-04   -5.077229e+05
 2015-07-05    2.905602e+06
                   ...
 2016-12-27   -8.364752e+05
 2016-12-28    1.278736e+06
 2016-12-29    2.673703e+05
 2016-12-30   -1.365708e+05
 2016-12-31   -2.014666e+05
 Name: de, Length: 550, dtype: float64)
```

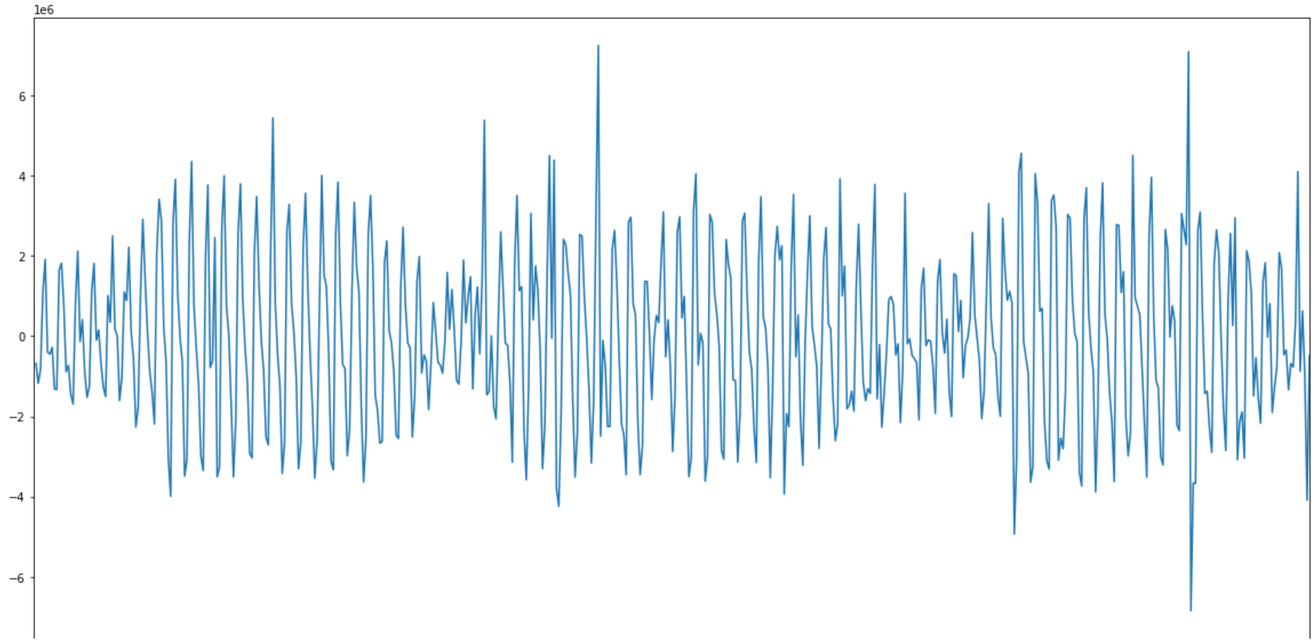
```
for i in trial_5.columns:
  trial_5[i].diff().plot()
  plt.show()
  pvalue = sm.tsa.stattools.adfuller(trial_5[i].diff().diff(1).dropna())[1]
  if pvalue <= 0.05:
    print(f'the time series residual for the language {i} is stationary')
  else:
    print(f'the time series residual for the language {i} is not stationary')
```

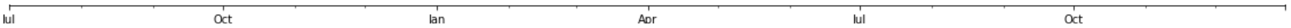


```
the time series residual for the language de is stationary
```



```
the time series residual for the language en is stationary
```

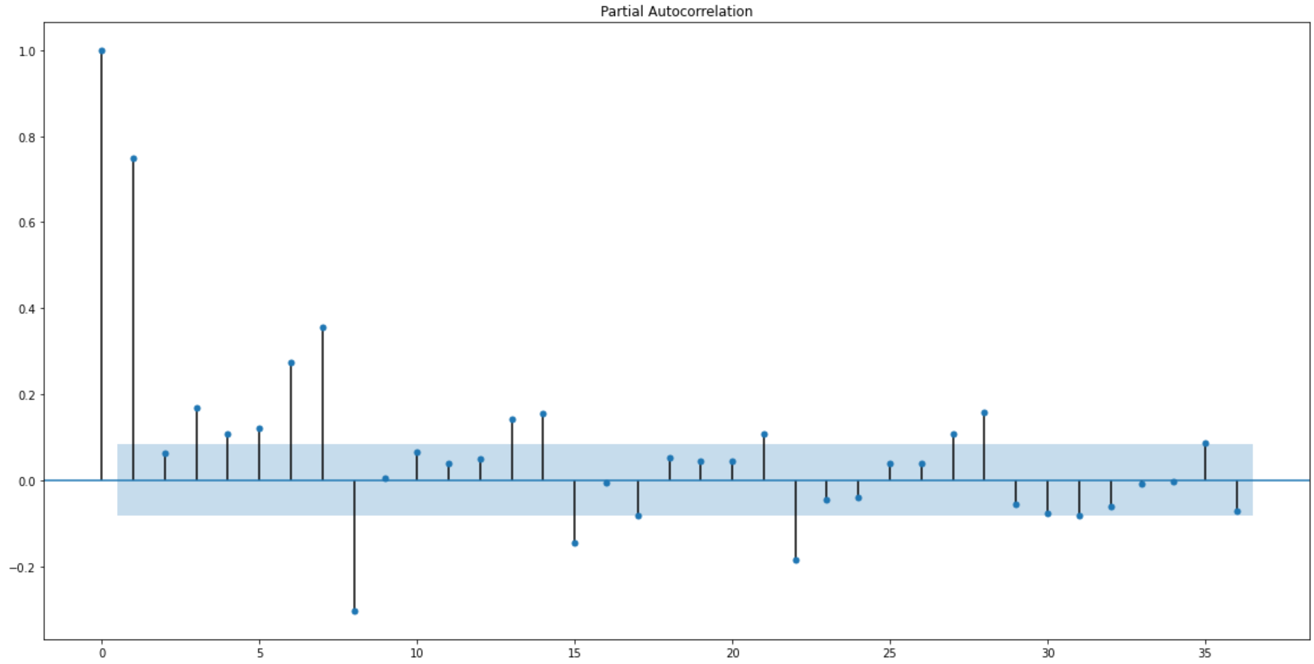




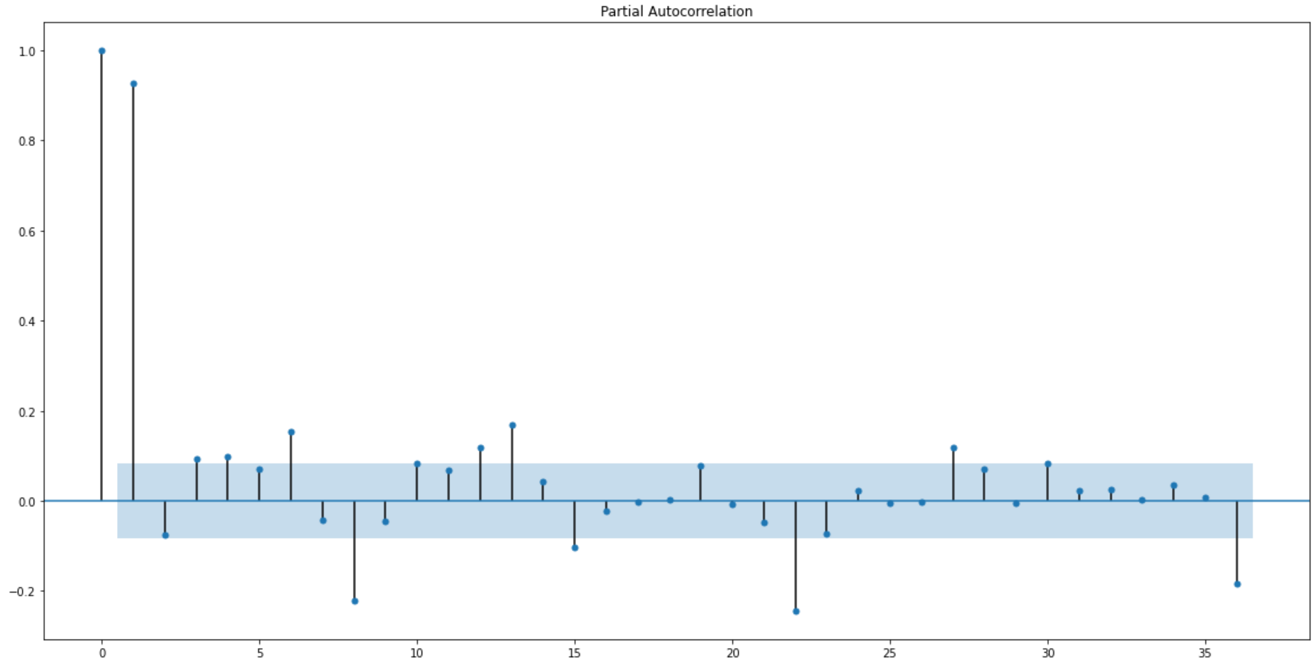
With both decomposition and differentiation we find that the time series with respect to data is stationary

Creating model training and forecasting with ARIMA, SARIMAX

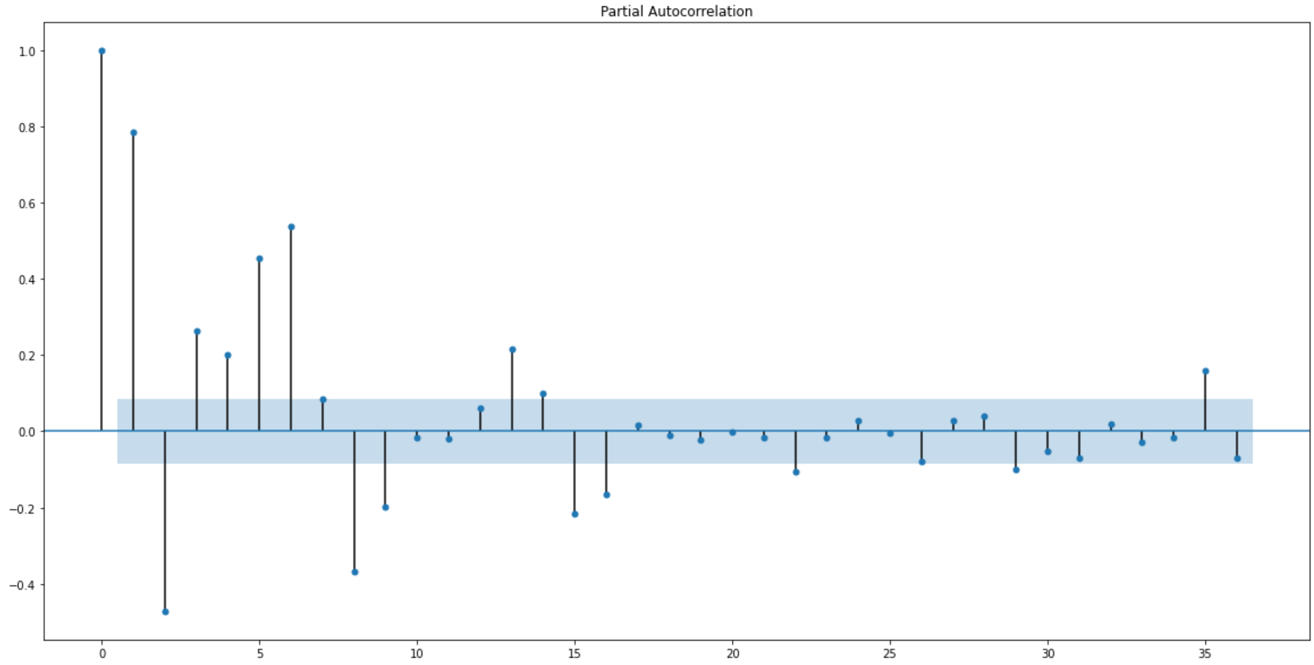
```
from statsmodels.graphics.tsaplots import plot_acf, plot_pacf

for i in trial_5.columns:
  plot_pacf(trial_5[i],lags=36)
  plt.show()
  print(f'the correlation for the language {i}')
```



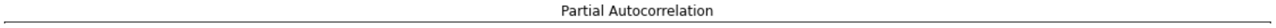
```
the correlation for the language de
```



```
the correlation for the language en
```

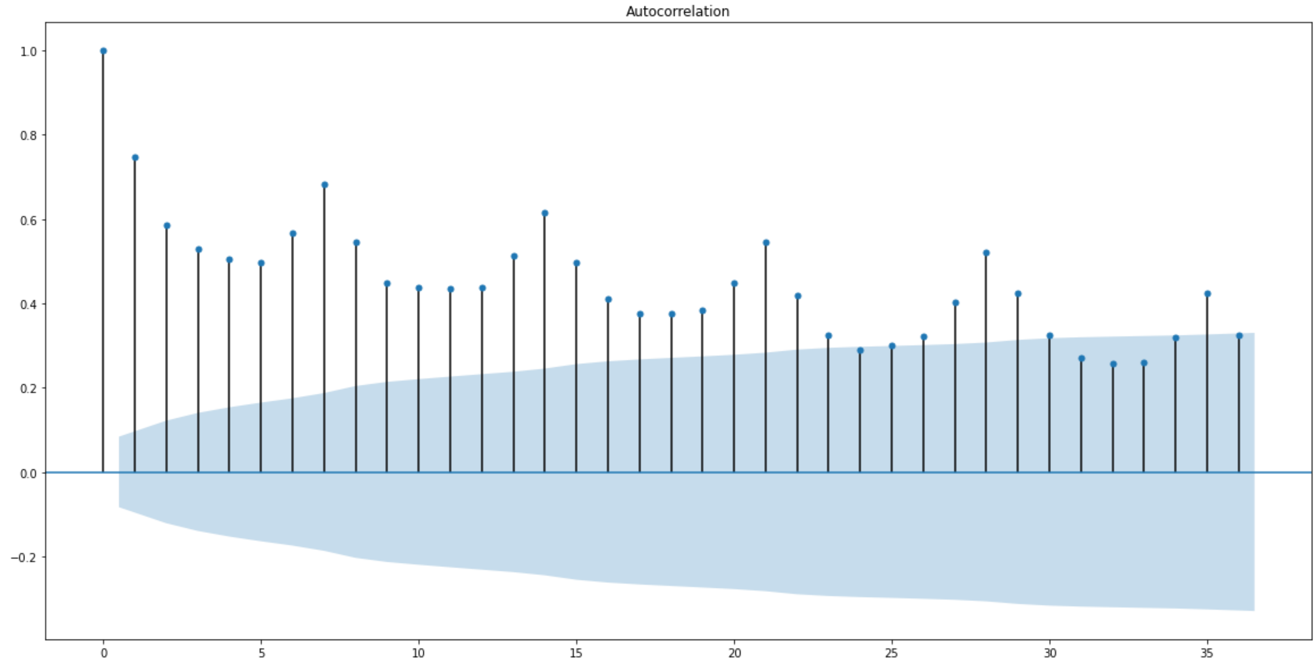


```
the correlation for the language es
```

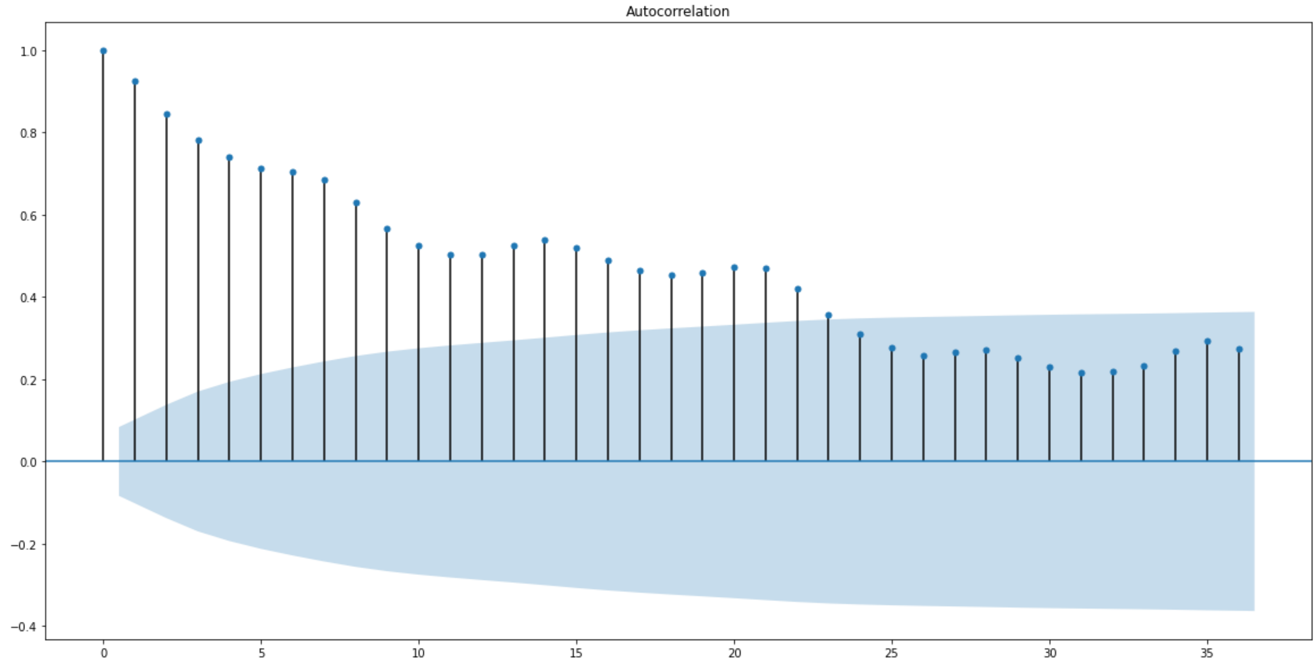


```
# the corresponding p values for each language is  2,2,7,4,3,4,2
```

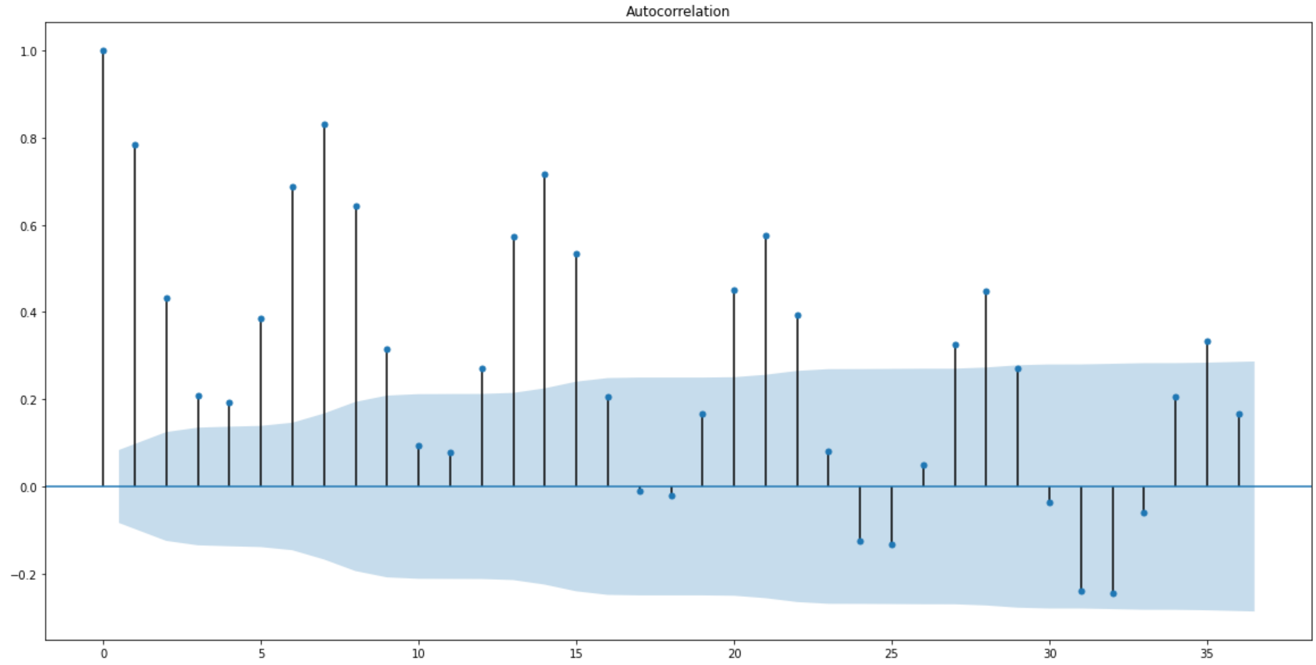
```
for i in trial_5.columns:
  plot_acf(trial_5[i],lags=36)
  plt.show()
  print(f'the correlation for the language {i}')
```



```
the correlation for the language de
```



```
the correlation for the language en
```



```
the correlation for the language es
```

```
# the corresponding q values for each language is 23,22,10,30,22,14,30
```

```
# for arima d is required consider d values
```

```
from statsmodels.tsa.statespace.sarimax import SARIMAX
```

```
p=[2,2,7,4,3,4,2]
```

```
q=[23,22,10,30,22,14,30]
```

```
np.argmax(p)
```

```
2
```

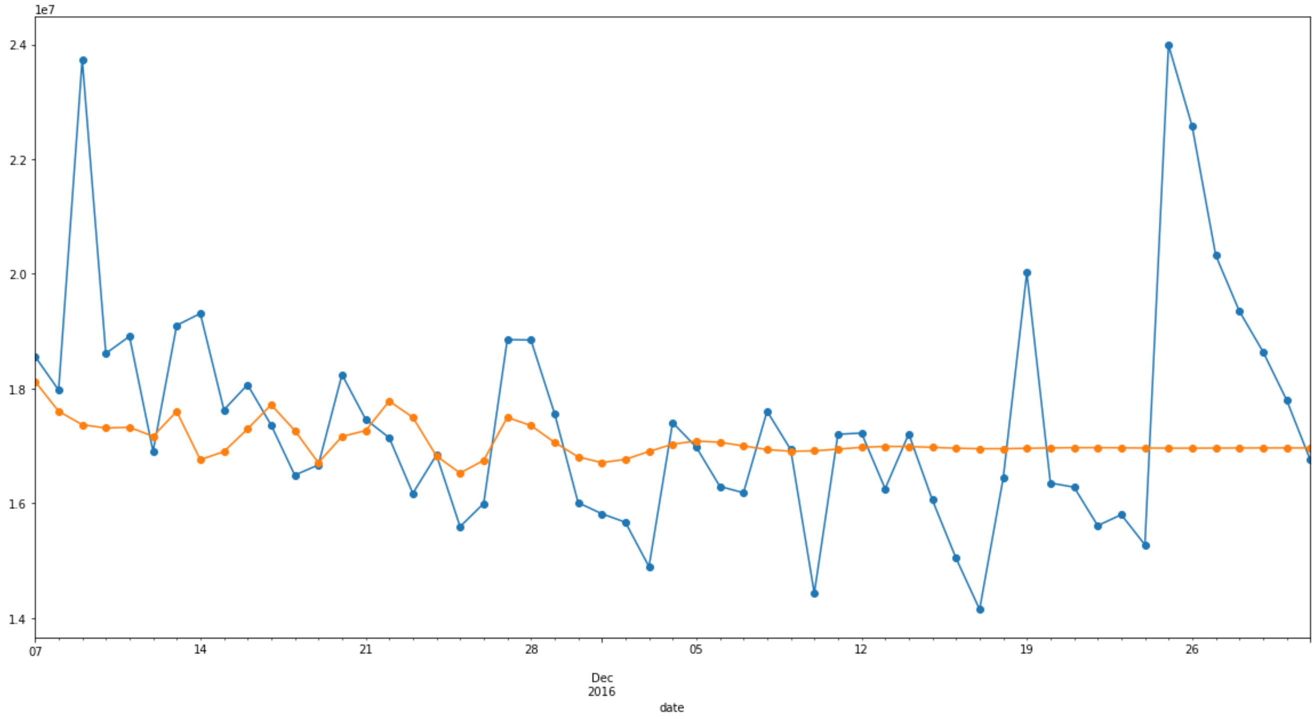
```
trial_5.shape[1]
```

```
7
```

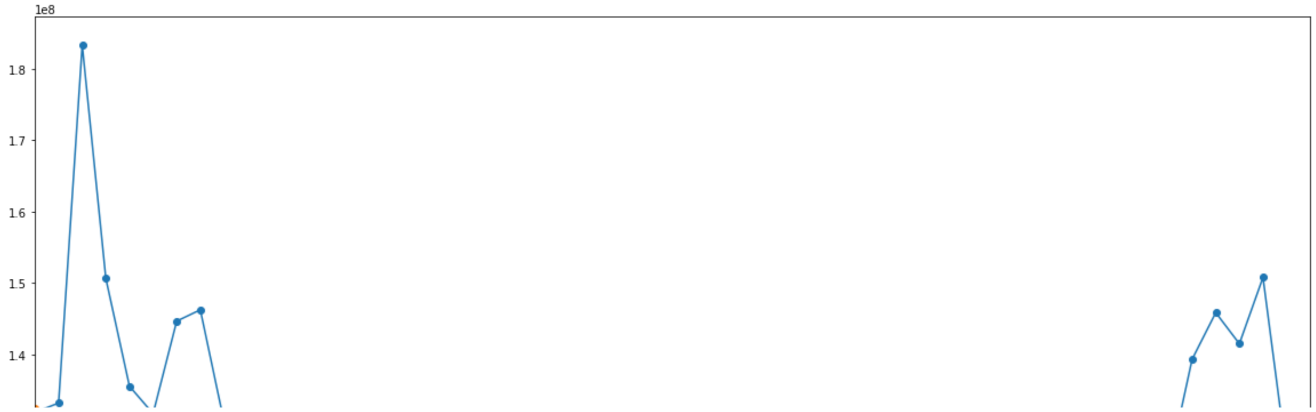
```
from sklearn.metrics import (
    mean_squared_error as mse,
    mean_absolute_error as mae,
    mean_absolute_percentage_error as mape
)

# Creating a function to print values of all these metrics.
def performance(actual, predicted):
    # print('MAE :', round(mae(actual, predicted), 3))
    # print('RMSE :', round(mse(actual, predicted)**0.5, 3))
    return round(mape(actual, predicted), 3)
```

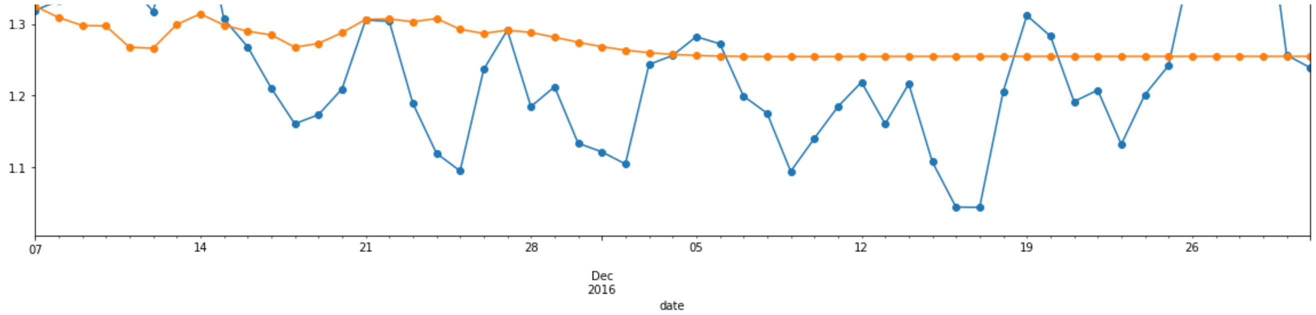
```
for i in range(trial_5.shape[1]):
  train_x = pd.DataFrame(trial_5.loc[trial_5.index < trial_5.index[-55]].copy().iloc[:,i])
  test_x  = pd.DataFrame(trial_5.loc[trial_5.index >= trial_5.index[-55]].copy().iloc[:,i])
  array=[]
  for d in range(1,3):

    model = SARIMAX(train_x.iloc[:,0], order=(p[i], d, q[i]))
    model = model.fit(disp=False)

    test_x.loc[:,'pred'] = model.forecast(steps=55)
    array.append(performance(test_x.iloc[:,0], test_x.loc[:,'pred']))

  best_d=np.argmax(array)
  model = SARIMAX(train_x.iloc[:,0], order=(p[i], best_d, q[i]))
  model = model.fit(disp=False)
  test_x.loc[:,'pred'] = model.forecast(steps=55)
  test_x.iloc[:,0].plot(style='-o')
  test_x.loc[:,'pred'].plot(style='-o')
  plt.show()
  a=performance(test_x.iloc[:,0], test_x.loc[:,'pred'])
  print(f'The performance is {a} for the language {trial_5.columns[i]}')
```

```
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provide
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provide
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provide
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provide
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/statespace/sarimax.py:978: UserWarning: Non-invertible starting MA paramet
  warn('Non-invertible starting MA parameters found.'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provide
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provide
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
```

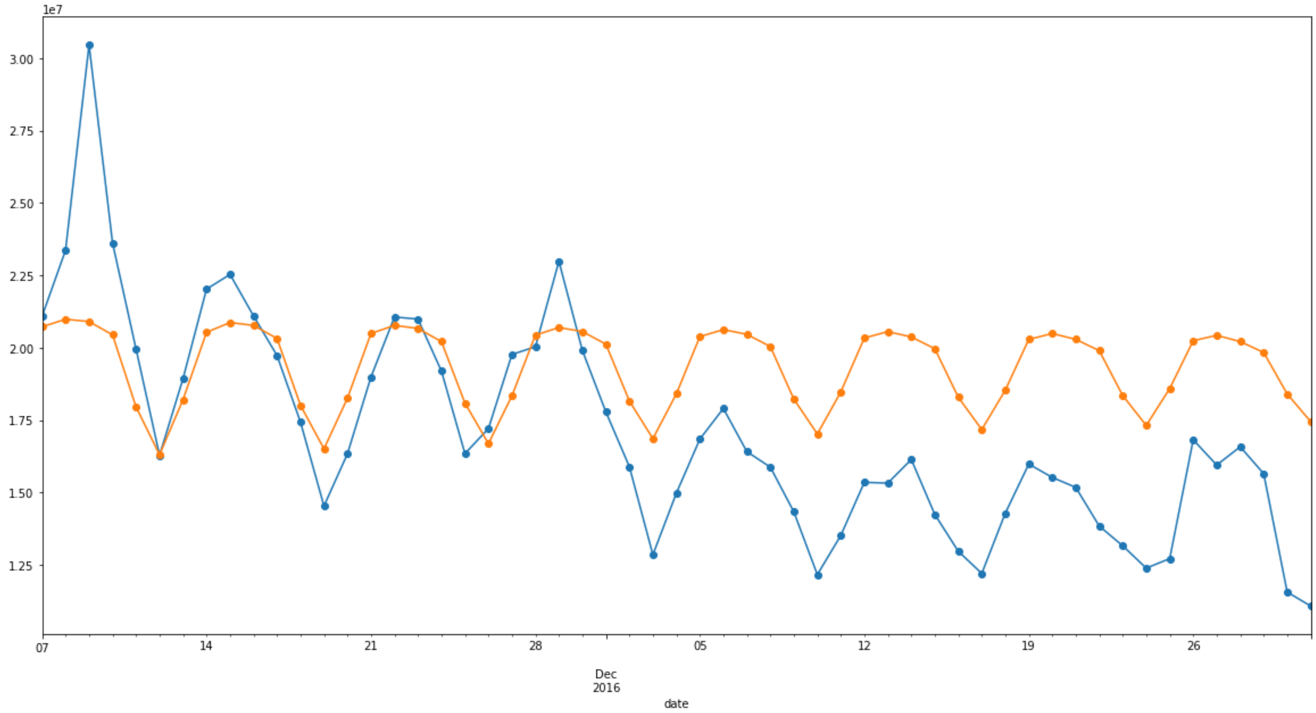


```
The performance is 0.072 for the language de
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provide
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provide
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provide
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provide
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/statespace/sarimax.py:978: UserWarning: Non-invertible starting MA paramet
  warn('Non-invertible starting MA parameters found.'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provide
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provide
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
```

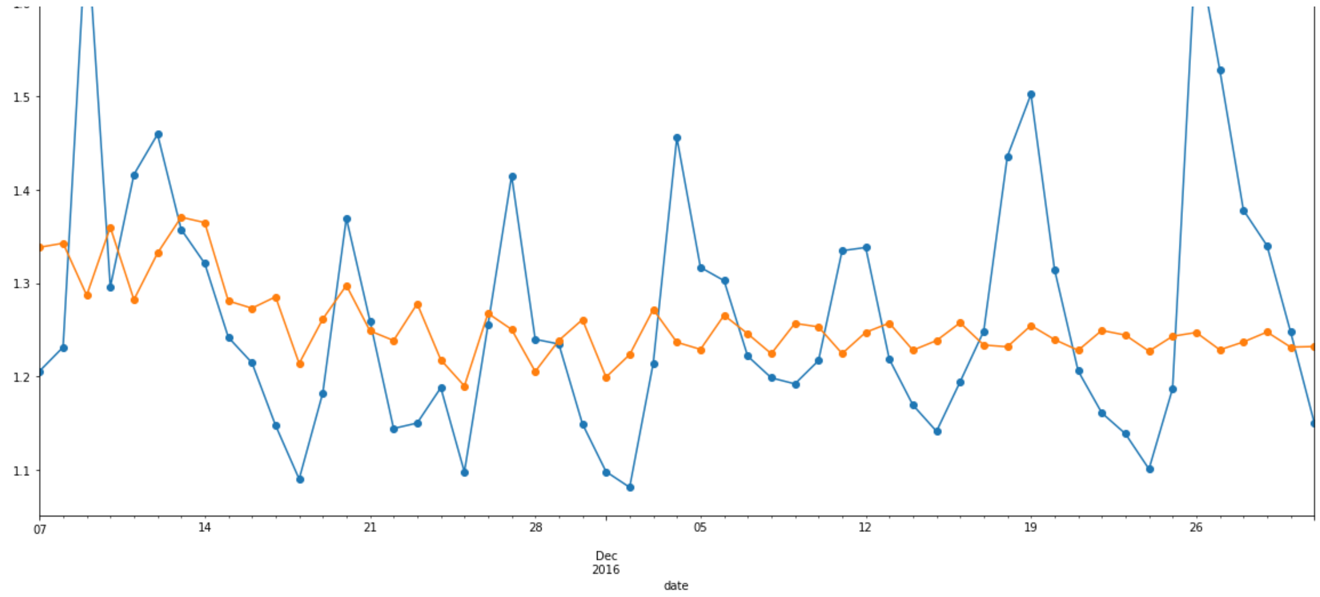




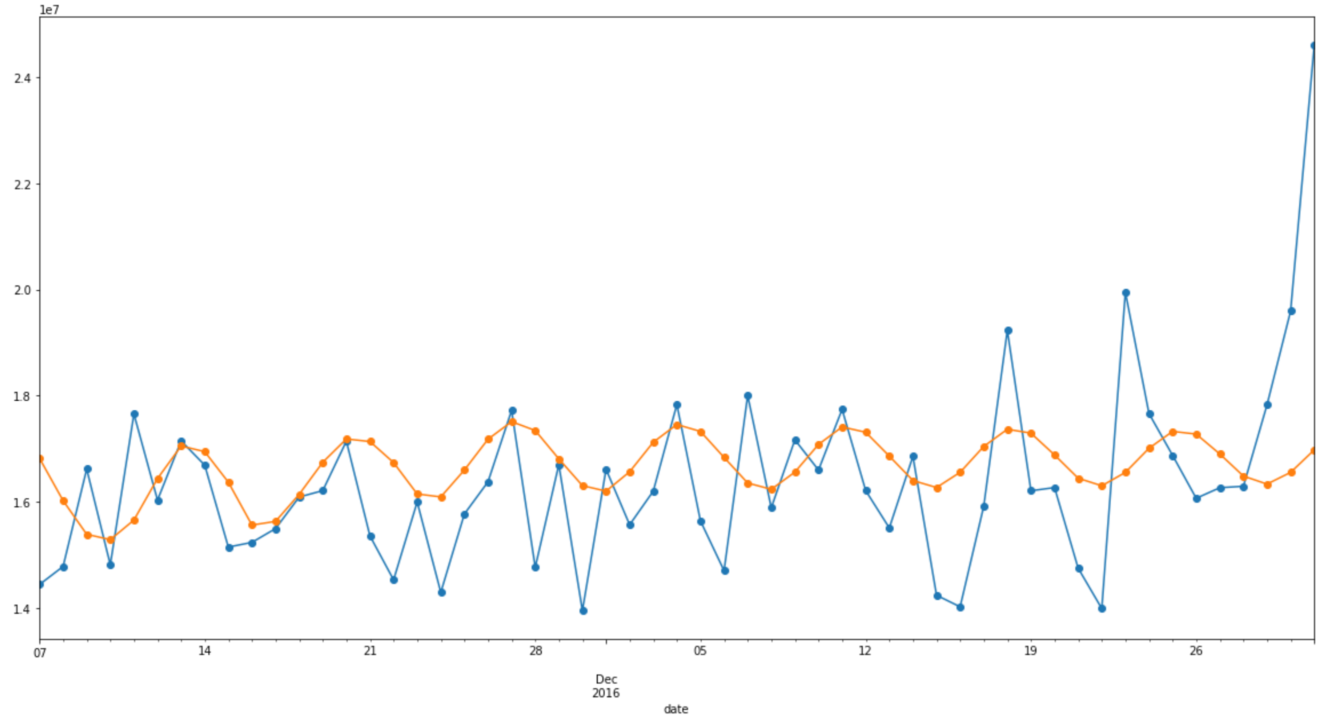
```
The performance is 0.076 for the language en
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
```



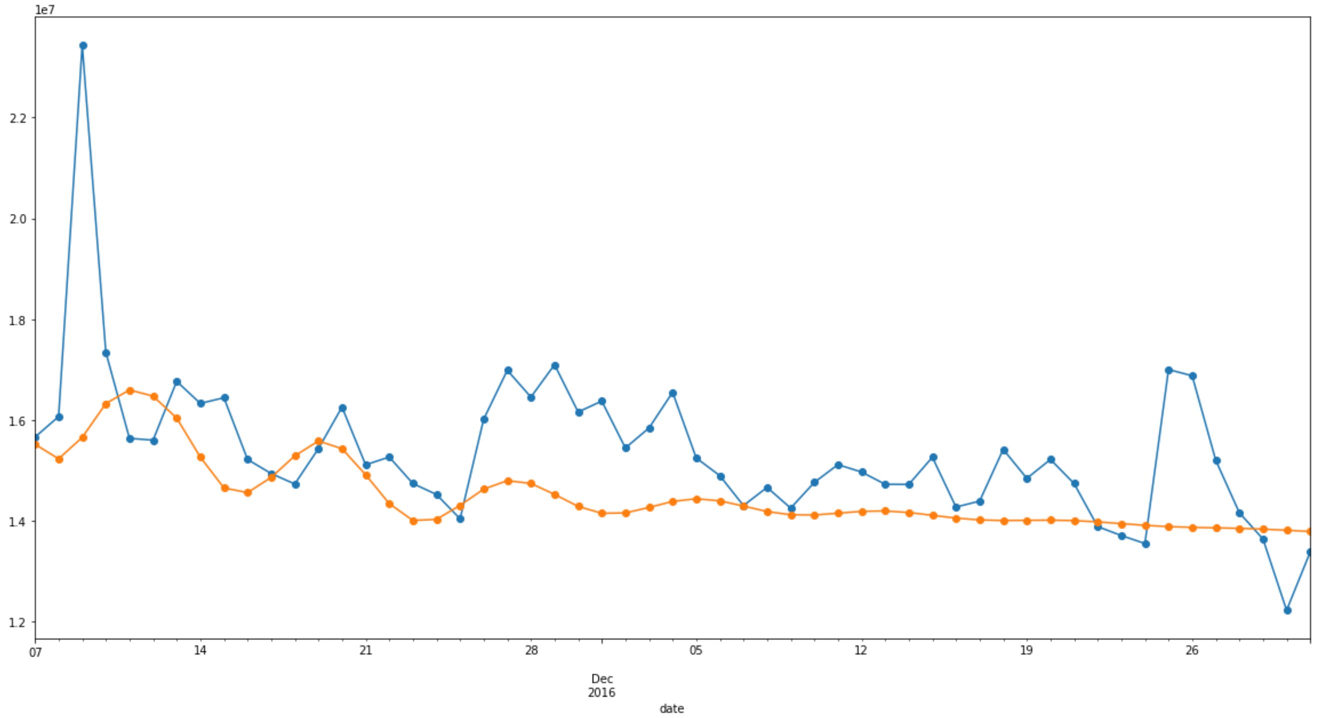
```
The performance is 0.211 for the language es
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/statespace/sarimax.py:978: UserWarning: Non-invertible starting MA paramet
  warn('Non-invertible starting MA parameters found.'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
```



```
The performance is 0.075 for the language fr
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/statespace/sarimax.py:978: UserWarning: Non-invertible starting MA paramet
  warn('Non-invertible starting MA parameters found.'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
```



```
The performance is 0.075 for the language ja
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
```



```
The performance is 0.065 for the language ru
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/statespace/sarimax.py:978: UserWarning: Non-invertible starting MA paramet
  warn('Non-invertible starting MA parameters found.'
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
  warnings.warn("Maximum Likelihood optimization failed to "
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided
  warnings.warn('No frequency information was'
```

```
# trial_5.index
```

```
/usr/local/lib/python3.8/dist-packages/statsmodels/base/model.py:566: ConvergenceWarning: Maximum Likelihood optimization failed
```

```
exog_1.set_index(trial_5.index,inplace=True)
```

9.0

```
exog_1
```

	Exog
date	
2015-07-01	0
2015-07-02	0
2015-07-03	0
2015-07-04	0
2015-07-05	0
...	...
2016-12-27	1
2016-12-28	1
2016-12-29	1
2016-12-30	0
2016-12-31	0

550 rows × 1 columns

```
trial_5=trial_5.join(exog_1)
```

```
trial_5
```

	de	en	es	fr	ja	ru	zh	Exog
date								
2015-07-01	1.507832e+07	9.415409e+07	1.618992e+07	9.232359e+06	1.549925e+07	1.170350e+07	5.159275e+06	0
2015-07-02	1.489782e+07	9.387991e+07	1.551257e+07	9.286675e+06	1.725639e+07	1.186729e+07	5.165336e+06	0
2015-07-03	1.437188e+07	8.960875e+07	1.433925e+07	8.959763e+06	1.594099e+07	1.116309e+07	5.137253e+06	0
2015-07-04	1.333821e+07	9.290429e+07	1.351702e+07	9.521535e+06	1.909167e+07	1.063287e+07	5.176900e+06	0
2015-07-05	1.521015e+07	9.563976e+07	1.462066e+07	9.362161e+06	1.846234e+07	1.117435e+07	5.454677e+06	0
...	...	...	...	...	...	...	...	...
2016-12-27	2.032261e+07	1.458706e+08	1.594582e+07	1.528222e+07	1.626896e+07	1.520168e+07	6.487991e+06	1
2016-12-28	1.934974e+07	1.415205e+08	1.657789e+07	1.378210e+07	1.629641e+07	1.416161e+07	6.522969e+06	1
2016-12-29	1.864423e+07	1.507996e+08	1.564768e+07	1.340043e+07	1.782839e+07	1.364024e+07	6.051296e+06	1
2016-12-30	1.780216e+07	1.256468e+08	1.156067e+07	1.247502e+07	1.959575e+07	1.222803e+07	6.117870e+06	0
2016-12-31	1.675861e+07	1.238632e+08	1.107802e+07	1.150493e+07	2.460054e+07	1.338433e+07	6.305259e+06	0

550 rows × 8 columns

```
# sarimax only for english

train_x = pd.DataFrame(trial_5.loc[trial_5.index < trial_5.index[-55]].copy().iloc[:,[1,-1]])
test_x  = pd.DataFrame(trial_5.loc[trial_5.index >= trial_5.index[-55]].copy().iloc[:,[1,-1]])
```

```
model = SARIMAX(train_x['en'],exog=train_x['Exog'],order=(1,1,1),seasonal_order=(1,1,1,7),enforce_invertibility=False)
```

```
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided,
  warnings.warn('No frequency information was'
/usr/local/lib/python3.8/dist-packages/statsmodels/tsa/base/tsa_model.py:524: ValueWarning: No frequency information was provided,
  warnings.warn('No frequency information was'
```

```
results = model.fit(disp=False)
```

```
exog_forecast = test_x[['Exog']]
predictions = results.predict(start=train_x.shape[0], end=train_x.shape[0]+test_x.shape[0]-1, exog=exog_forecast).rename('Predictions')
```

```
performance(test_x['en'], predictions)
```

```
0.098
```

Forecasting with prophet

```
!pip install pystan~=2.14
!pip install fbprophet
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting pystan~=2.14
  Downloading pystan-2.19.1.1-cp38-cp38-manylinux1_x86_64.whl (62.6 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 62.6/62.6 MB 15.6 MB/s eta 0:00:00
Requirement already satisfied: numpy>=1.7 in /usr/local/lib/python3.8/dist-packages (from pystan~=2.14) (1.21.6)
Requirement already satisfied: Cython!=0.25.1,>=0.22 in /usr/local/lib/python3.8/dist-packages (from pystan~=2.14) (0.29.33)
Installing collected packages: pystan
  Attempting uninstall: pystan
    Found existing installation: pystan 3.3.0
    Uninstalling pystan-3.3.0:
      Successfully uninstalled pystan-3.3.0
Successfully installed pystan-2.19.1.1
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Collecting fbprophet
  Downloading fbprophet-0.7.1.tar.gz (64 kB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 64.0/64.0 KB 2.5 MB/s eta 0:00:00
  Preparing metadata (setup.py) ... done
Requirement already satisfied: Cython>=0.22 in /usr/local/lib/python3.8/dist-packages (from fbprophet) (0.29.33)
Collecting cmdstanpy==0.9.5
  Downloading cmdstanpy-0.9.5-py3-none-any.whl (37 kB)
Requirement already satisfied: pystan>=2.14 in /usr/local/lib/python3.8/dist-packages (from fbprophet) (2.19.1.1)
Requirement already satisfied: numpy>=1.15.4 in /usr/local/lib/python3.8/dist-packages (from fbprophet) (1.21.6)
Requirement already satisfied: pandas>=1.0.4 in /usr/local/lib/python3.8/dist-packages (from fbprophet) (1.3.5)
Requirement already satisfied: matplotlib>=2.0.0 in /usr/local/lib/python3.8/dist-packages (from fbprophet) (3.2.2)
Requirement already satisfied: LunarCalendar>=0.0.9 in /usr/local/lib/python3.8/dist-packages (from fbprophet) (0.0.9)
Requirement already satisfied: convertdate>=2.1.2 in /usr/local/lib/python3.8/dist-packages (from fbprophet) (2.4.0)
Requirement already satisfied: holidays>=0.10.2 in /usr/local/lib/python3.8/dist-packages (from fbprophet) (0.19)
Collecting setuptools-git>=1.2
  Downloading setuptools_git-1.2-py2.py3-none-any.whl (10 kB)
Requirement already satisfied: python-dateutil>=2.8.0 in /usr/local/lib/python3.8/dist-packages (from fbprophet) (2.8.2)
Requirement already satisfied: tqdm>=4.36.1 in /usr/local/lib/python3.8/dist-packages (from fbprophet) (4.64.1)
Requirement already satisfied: pymeeus<=1,>=0.3.13 in /usr/local/lib/python3.8/dist-packages (from convertdate>=2.1.2->fbprophet) (
Requirement already satisfied: korean-lunar-calendar in /usr/local/lib/python3.8/dist-packages (from holidays>=0.10.2->fbprophet) (
Requirement already satisfied: hijri-converter in /usr/local/lib/python3.8/dist-packages (from holidays>=0.10.2->fbprophet) (2.2.4)
Requirement already satisfied: ephem>=3.7.5.3 in /usr/local/lib/python3.8/dist-packages (from LunarCalendar>=0.0.9->fbprophet) (4.1
Requirement already satisfied: pytz in /usr/local/lib/python3.8/dist-packages (from LunarCalendar>=0.0.9->fbprophet) (2022.7.1)
Requirement already satisfied: pyparsing!=2.0.4,!=2.1.2,!=2.1.6,>=2.0.1 in /usr/local/lib/python3.8/dist-packages (from matplotlib>
Requirement already satisfied: kiwisolver>=1.0.1 in /usr/local/lib/python3.8/dist-packages (from matplotlib>=2.0.0->fbprophet) (1.4
Requirement already satisfied: cycler>=0.10 in /usr/local/lib/python3.8/dist-packages (from matplotlib>=2.0.0->fbprophet) (0.11.0)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.8/dist-packages (from python-dateutil>=2.8.0->fbprophet) (1.15.0)
Building wheels for collected packages: fbprophet
  Building wheel for fbprophet (setup.py) ... done
  Created wheel for fbprophet: filename=fbprophet-0.7.1-py3-none-any.whl size=9537455 sha256=08785b29ec967aceda6e84114d239b2e87e8ac
  Stored in directory: /root/.cache/pip/wheels/d0/d2/ae/c579b7fd160999d35908f3cb8ebcad7ef64ecaca7b78e4c3c8
Successfully built fbprophet
Installing collected packages: setuptools-git, cmdstanpy, fbprophet
  Attempting uninstall: cmdstanpy
    Found existing installation: cmdstanpy 1.1.0
    Uninstalling cmdstanpy-1.1.0:
      Successfully uninstalled cmdstanpy-1.1.0
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the sc
prophet 1.1.2 requires cmdstanpy>=1.0.4, but you have cmdstanpy 0.9.5 which is incompatible.
Successfully installed cmdstanpy-0.9.5 fbprophet-0.7.1 setuptools-git-1.2
```

```
exog=exog_1['Exog'].to_numpy()
```

```
trial_6= trial_5.iloc[:,[1]].copy().reset_index()
```

```
trial_6
```

	date	en
0	2015-07-01	9.415409e+07
1	2015-07-02	9.387991e+07
2	2015-07-03	8.960875e+07
3	2015-07-04	9.290429e+07
4	2015-07-05	9.563976e+07
...	...	...
545	2016-12-27	1.458706e+08
546	2016-12-28	1.415205e+08
547	2016-12-29	1.507996e+08
548	2016-12-30	1.256468e+08
549	2016-12-31	1.238632e+08

550 rows × 2 columns

```
trial_6.columns = [['ds', 'y']]
```

```
# trial_6=df.copy()
trial_6['exog'] = exog
trial_6.columns = ['ds', 'y', 'exog']
# trial_6.head()
```

```
from prophet import Prophet
```

```
model=Prophet(weekly_seasonality=True)
model.add_regressor('exog')
model.fit(trial_6[:-55])
forecast = model.predict(trial_6)
fig = model.plot(forecast)
```