تعریف مسئله:

پیش بینی دوز روزانه تعریف شده هر آنتی بیوتیک در سال ۲۰۱۶

In [35]:
```
import joblib as jl
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
pd.set_option('display.max_rows', 50000)
pd.set_option('display.max_columns', 500)
```

در این مرحله کتابخانه های مورد نیاز برای کار با داده و رسم نمودار هارا فراخوانی کرده ایم

In [3]:
```
DDD_monthly=pd.read_excel('S4 Table_v1_copy.xlsx',sheet_name='Sheet5_1')
DiD_monthly=pd.read_excel('S4 Table_v1_copy.xlsx',sheet_name='Sheet5_2')

print(DDD_monthly.shape)
print(DiD_monthly.shape)
```

```
(60, 33)
(60, 33)
```

In [4]:
```
DDD_monthly.head()
```

Out[4]:

	year	month	J01AA	J01BA	J01CA	J01CE	J01CF	J01CG	J01CI
0	2011	1	808432.000	82178.000	2293639.733	1581290.866	127553.930	5972.5	9162924.18
1	2011	2	616620.000	45122.249	1157493.301	696282.870	61136.109	2862.5	4769411.29
2	2011	3	802755.000	86423.583	1548117.088	1330648.932	90434.895	7289.5	5617689.06
3	2011	4	736725.667	92561.667	1671628.510	1363893.906	115792.608	6653.0	8017712.68
4	2011	5	1038556.000	73520.334	1444498.204	1482620.945	110954.716	8070.0	5187866.11

In [5]:
```
DiD_monthly.tail()
```

Out[5]:

	year	month	J01AA	J01BA	J01CA	J01CE	J01CF	J01CG	J01CR	J01DA
55	2015	8	0.314844	0.006068	0.369357	0.481403	0.021986	0.000289	1.747810	0.016423
56	2015	9	0.308779	0.007435	0.432951	0.468047	0.023989	0.000295	3.072444	0.022249
57	2015	10	0.294813	0.005639	0.302966	0.361268	0.017249	0.000312	1.253339	0.011034
58	2015	11	0.304965	0.006024	0.417773	0.507047	0.019695	0.000220	2.135304	0.019047
59	2015	12	0.393283	0.008865	0.484667	0.417484	0.021520	0.000453	2.753057	0.022413

- ***دراین مرحله داده های شیت 5-1 و 5-1 خوانده می شود***
-
- **در این داده ها سال و ماه در سطر قرار دارند و دسته بندی ریز آنتی بیوتیک در ستون**
-
- **مقدار هر سلول میزان دوز مصرفی آنتی بیوتیک را نشان می دهد**

In [11]:
```
DDD_monthly.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 60 entries, 0 to 59
Data columns (total 33 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   year    60 non-null     int64  
 1   month   60 non-null     int64  
 2   J01AA   60 non-null     float64
 3   J01BA   60 non-null     float64
 4   J01CA   60 non-null     float64
 5   J01CE   60 non-null     float64
 6   J01CF   60 non-null     float64
 7   J01CG   60 non-null     float64
 8   J01CR   60 non-null     float64
 9   J01DA   60 non-null     float64
 10  J01DB   60 non-null     float64
 11  J01DC   60 non-null     float64
 12  J01DD   60 non-null     float64
 13  J01DE   60 non-null     float64
 14  J01DF   60 non-null     float64
 15  J01DH   60 non-null     float64
 16  J01DI   60 non-null     float64
 17  J01EA   37 non-null     float64
 18  J01EB   60 non-null     int64  
 19  J01EC   60 non-null     float64
 20  J01EE   27 non-null     float64
 21  J01FA   60 non-null     float64
 22  J01FF   60 non-null     float64
 23  J01FG   60 non-null     float64
 24  J01GA   60 non-null     int64  
 25  J01GB   60 non-null     float64
 26  J01MA   60 non-null     float64
 27  J01XA   60 non-null     float64
 28  J01XB   34 non-null     float64
 29  J01XC   60 non-null     float64
 30  J01XD   60 non-null     float64
 31  J01XE   60 non-null     float64
 32  J01XX   60 non-null     float64
dtypes: float64(29), int64(4)
memory usage: 15.6 KB
```

In [12]:
```
DiD_monthly.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 60 entries, 0 to 59
Data columns (total 33 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   year    60 non-null     int64  
 1   month   60 non-null     int64  
 2   J01AA   60 non-null     float64
 3   J01BA   60 non-null     float64
 4   J01CA   60 non-null     float64
 5   J01CE   60 non-null     float64
 6   J01CF   60 non-null     float64
 7   J01CG   60 non-null     float64
 8   J01CR   60 non-null     float64
 9   J01DA   60 non-null     float64
 10  J01DB   60 non-null     float64
 11  J01DC   60 non-null     float64
 12  J01DD   60 non-null     float64
 13  J01DE   60 non-null     float64
 14  J01DF   60 non-null     float64
 15  J01DH   60 non-null     float64
 16  J01DI   60 non-null     float64
 17  J01EA   60 non-null     float64
 18  J01EB   60 non-null     float64
 19  J01EC   60 non-null     float64
 20  J01EE   60 non-null     float64
 21  J01FA   60 non-null     float64
 22  J01FF   60 non-null     float64
 23  J01FG   60 non-null     float64
 24  J01GA   60 non-null     float64
 25  J01GB   60 non-null     float64
 26  J01MA   60 non-null     float64
 27  J01XA   60 non-null     float64
 28  J01XB   60 non-null     float64
 29  J01XC   60 non-null     float64
 30  J01XD   60 non-null     float64
 31  J01XE   60 non-null     float64
 32  J01XX   60 non-null     float64
dtypes: float64(31), int64(2)
memory usage: 15.6 KB
```

In [13]:
```
DDD_monthly.dtypes
```

Out[13]:
```
year       int64
month      int64
J01AA    float64
J01BA    float64
J01CA    float64
J01CE    float64
J01CF    float64
J01CG    float64
J01CR    float64
J01DA    float64
J01DB    float64
J01DC    float64
J01DD    float64
J01DE    float64
J01DF    float64
J01DH    float64
J01DI    float64
J01EA    float64
J01EB      int64
J01EC    float64
J01EE    float64
J01FA    float64
J01FF    float64
J01FG    float64
J01GA      int64
J01GB    float64
J01MA    float64
J01XA    float64
J01XB    float64
J01XC    float64
J01XD    float64
J01XE    float64
J01XX    float64
dtype: object
```

In [14]:
```
DDD_monthly.columns
```

Out[14]:
```
Index(['year', 'month', 'J01AA', 'J01BA', 'J01CA', 'J01CE', 'J01CF', 'J01
CG',
       'J01CR', 'J01DA', 'J01DB', 'J01DC', 'J01DD', 'J01DE', 'J01DF', 'J0
1DH',
       'J01DI', 'J01EA', 'J01EB', 'J01EC', 'J01EE', 'J01FA', 'J01FF', 'J0
1FG',
       'J01GA', 'J01GB', 'J01MA', 'J01XA', 'J01XB', 'J01XC', 'J01XD', 'J0
1XE',
       'J01XX'],
      dtype='object')
```

In [15]:
```
DiD_monthly.columns
```

Out[15]:
```
Index(['year', 'month', 'J01AA', 'J01BA', 'J01CA', 'J01CE', 'J01CF', 'J01
CG',
       'J01CR', 'J01DA', 'J01DB', 'J01DC', 'J01DD', 'J01DE', 'J01DF', 'J0
1DH',
       'J01DI', 'J01EA', 'J01EB', 'J01EC', 'J01EE', 'J01FA', 'J01FF', 'J0
1FG',
       'J01GA', 'J01GB', 'J01MA', 'J01XA', 'J01XB', 'J01XC', 'J01XD', 'J0
1XE',
       'J01XX'],
      dtype='object')
```

در این مرحله خلاصه ایی از اطلاعات داده ها را چاپ کردیم

In [16]:
```
DDD_monthly=DDD_monthly.astype({'year':'int64', 'month':'int64', 'J01AA':'fl
                    'J01CE':'float64', 'J01CF':'float64', 'J01CG':'float64',
                    'J01DB':'float64', 'J01DC':'float64', 'J01DD':'float64',
                    'J01DH':'float64','J01DI':'float64', 'J01EA':'float64',
                    'J01EB':'float64', 'J01EC':'float64', 'J01EE':'float64',
                    'J01FG':'float64', 'J01GA':'float64', 'J01GB':'float64',
                    'J01XB':'float64', 'J01XC':'float64', 'J01XD':'float64',
DDD_monthly.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 60 entries, 0 to 59
Data columns (total 33 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   year    60 non-null     int64  
 1   month   60 non-null     int64  
 2   J01AA   60 non-null     float64
 3   J01BA   60 non-null     float64
 4   J01CA   60 non-null     float64
 5   J01CE   60 non-null     float64
 6   J01CF   60 non-null     float64
 7   J01CG   60 non-null     float64
 8   J01CR   60 non-null     float64
 9   J01DA   60 non-null     float64
 10  J01DB   60 non-null     float64
 11  J01DC   60 non-null     float64
 12  J01DD   60 non-null     float64
 13  J01DE   60 non-null     float64
 14  J01DF   60 non-null     float64
 15  J01DH   60 non-null     float64
 16  J01DI   60 non-null     float64
 17  J01EA   37 non-null     float64
 18  J01EB   60 non-null     float64
 19  J01EC   60 non-null     float64
 20  J01EE   27 non-null     float64
 21  J01FA   60 non-null     float64
 22  J01FF   60 non-null     float64
 23  J01FG   60 non-null     float64
 24  J01GA   60 non-null     float64
 25  J01GB   60 non-null     float64
 26  J01MA   60 non-null     float64
 27  J01XA   60 non-null     float64
 28  J01XB   34 non-null     float64
 29  J01XC   60 non-null     float64
 30  J01XD   60 non-null     float64
 31  J01XE   60 non-null     float64
 32  J01XX   60 non-null     float64
dtypes: float64(31), int64(2)
memory usage: 15.6 KB
```

In [18]:
```
DiD_monthly=DiD_monthly.astype({'year':'int64', 'month':'int64', 'J01AA':'fl
                    'J01CE':'float64', 'J01CF':'float64', 'J01CG':'float64',
                    'J01DB':'float64', 'J01DC':'float64', 'J01DD':'float64',
                    'J01DH':'float64','J01DI':'float64', 'J01EA':'float64',
                    'J01EB':'float64', 'J01EC':'float64', 'J01EE':'float64',
                    'J01FG':'float64', 'J01GA':'float64', 'J01GB':'float64',
                    'J01XB':'float64', 'J01XC':'float64', 'J01XD':'float64',
DiD_monthly.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 60 entries, 0 to 59
Data columns (total 33 columns):
 #   Column  Non-Null Count  Dtype  
---  ------  --------------  -----  
 0   year    60 non-null     int64  
 1   month   60 non-null     int64  
 2   J01AA   60 non-null     float64
 3   J01BA   60 non-null     float64
 4   J01CA   60 non-null     float64
 5   J01CE   60 non-null     float64
 6   J01CF   60 non-null     float64
 7   J01CG   60 non-null     float64
 8   J01CR   60 non-null     float64
 9   J01DA   60 non-null     float64
 10  J01DB   60 non-null     float64
 11  J01DC   60 non-null     float64
 12  J01DD   60 non-null     float64
 13  J01DE   60 non-null     float64
```

در این مرحله همسان سازی داده ها انجام میشود

In [19]:
```
pd.set_option('display.float_format', lambda x: '%.5f' % x)
```

In [20]:
```
DDD_monthly.describe()
```

Out[20]:

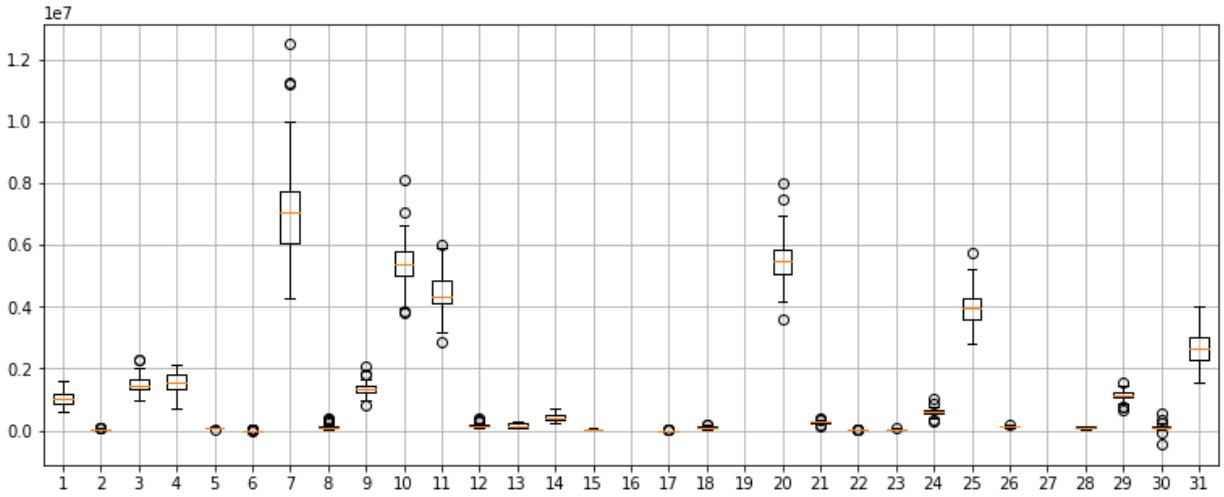
	year	month	J01AA	J01BA	J01CA	J01CE	J01
count	60.00000	60.00000	60.00000	60.00000	60.00000	60.00000	60.000
mean	2013.00000	6.50000	1011680.66390	37748.65650	1508265.79248	1545272.98183	93518.475
std	1.42615	3.48118	209879.19055	19488.94360	275238.30976	335184.90944	16559.393
min	2011.00000	1.00000	616620.00000	10993.16700	960667.05300	696282.87000	48937.393
25%	2012.00000	3.75000	861723.45850	25999.41700	1323105.54800	1358293.02950	86436.839
50%	2013.00000	6.50000	1025744.91650	31491.45750	1448611.05000	1567270.16500	93480.893
75%	2014.00000	9.25000	1172548.75025	45965.85550	1675403.09075	1813291.75250	104994.562
max	2015.00000	12.00000	1599060.00000	92561.66700	2311695.44900	2118915.42800	129818.715

In [21]:
```
DiD_monthly.describe()
```

Out[21]:

	year	month	J01AA	J01BA	J01CA	J01CE	J01CF	J01CG	J01CR
count	60.00000	60.00000	60.00000	60.00000	60.00000	60.00000	60.00000	60.00000	60.00000
mean	2013.00000	6.50000	0.23520	0.00856	0.34987	0.35923	0.02163	0.00041	1.66723
std	1.42615	3.48118	0.05287	0.00393	0.06673	0.08577	0.00363	0.00043	0.40844
min	2011.00000	1.00000	0.13945	0.00270	0.24626	0.15747	0.01254	0.00001	1.07863
25%	2012.00000	3.75000	0.19628	0.00604	0.30239	0.30079	0.01954	0.00017	1.41193
50%	2013.00000	6.50000	0.23688	0.00737	0.33060	0.34508	0.02200	0.00026	1.65412
75%	2014.00000	9.25000	0.27119	0.00992	0.38789	0.42115	0.02400	0.00041	1.80603
max	2015.00000	12.00000	0.39328	0.01954	0.55308	0.55416	0.02962	0.00176	3.07244

In [23]:
```
x1=DDD_monthly.loc[:,['J01AA', 'J01BA', 'J01CA', 'J01CE', 'J01CF', 'J01CG',
       'J01CR', 'J01DA', 'J01DB', 'J01DC', 'J01DD', 'J01DE', 'J01DF', 'J01DH',
       'J01DI', 'J01EA', 'J01EB', 'J01EC', 'J01EE', 'J01FA', 'J01FF', 'J01FG',
       'J01GA', 'J01GB', 'J01MA', 'J01XA', 'J01XB', 'J01XC', 'J01XD', 'J01XE',
       'J01XX']]
labels=str(['J01AA','J01BA', 'J01CA', 'J01CE', 'J01CF', 'J01CG','J01CR', 'J01DA',
        'J01DB', 'J01DC', 'J01DD', 'J01DE', 'J01DF', 'J01DH',       'J01DI',
        'J01EA', 'J01EB', 'J01EC', 'J01EE', 'J01FA', 'J01FF', 'J01FG',
        'J01GA', 'J01GB', 'J01MA', 'J01XA', 'J01XB', 'J01XC', 'J01XD', 'J01XE',
plt.figure(figsize=(13,5))
plt.boxplot(x1)
plt.grid()
plt.show()
```



در این مرحله به توصیف آماری داده ها پرداخته شده است

In [24]:
```
number_of_DDD_ATC_4=pd.DataFrame()
for fcn in DDD_monthly.columns[2:] :
    df_temp2=DDD_monthly.loc[:,['year','month']]
    df_temp2['fcn_DDD']=fcn
    df_temp2['fcnv_DDD']=DDD_monthly.loc[:,[fcn]]
    number_of_DDD_ATC_4=pd.concat([number_of_DDD_ATC_4,df_temp2],axis=0)
number_of_DDD_ATC_4.tail()
```

Out[24]:

	year	month	fcn_DDD	fcnv_DDD
55	2015	8	J01XX	3195654.28400
56	2015	9	J01XX	3008060.50500
57	2015	10	J01XX	2425091.74400
58	2015	11	J01XX	3173443.28300
59	2015	12	J01XX	3501452.32800

In [25]:
```
monthly_DID_in_sample_ATC_4=pd.DataFrame()
for fcn in DiD_monthly.columns[2:] :
    df_temp2=DiD_monthly.loc[:,['year','month']]
    df_temp2['fcn_DID']=fcn
    df_temp2['fcnv_DID']=DiD_monthly.loc[:,[fcn]]
    monthly_DID_in_sample_ATC_4=pd.concat([monthly_DID_in_sample_ATC_4,df_te
monthly_DID_in_sample_ATC_4.head()
```

Out[25]:

	year	month	fcn_DID	fcnv_DID
0	2011	1	J01AA	0.17064
1	2011	2	J01AA	0.13945
2	2011	3	J01AA	0.16945
3	2011	4	J01AA	0.15551
4	2011	5	J01AA	0.21922

In [26]:
```
number_of_DDD_ATC_4.shape
```

Out[26]: (1860, 4)

In [27]:
```
monthly_DID_in_sample_ATC_4.shape
```

Out[27]: (1860, 4)

در این مرحله ریز آنتی بیوتیک ها به عنوان ستون جدید در نظر گرفته میشود

In [29]:
```
DDD_DID=pd.concat([number_of_DDD_ATC_4,monthly_DID_in_sample_ATC_4['fcnv_DII
DDD_DID.columns=['year','month','fcn','fcnv_DDD','fcnv_DID']
DDD_DID.head()
```

Out[29]:

	year	month	fcn	fcnv_DDD	fcnv_DID
0	2011	1	J01AA	808432.00000	0.17064
1	2011	2	J01AA	616620.00000	0.13945
2	2011	3	J01AA	802755.00000	0.16945
3	2011	4	J01AA	736725.66700	0.15551
4	2011	5	J01AA	1038556.00000	0.21922

In [30]:
```
DDD_DID.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 1860 entries, 0 to 59
Data columns (total 5 columns):
 #   Column    Non-Null Count  Dtype  
---  ------    --------------  -----  
 0   year      1860 non-null   int64  
 1   month     1860 non-null   int64  
 2   fcn       1860 non-null   object 
 3   fcnv_DDD  1778 non-null   float64
 4   fcnv_DID  1860 non-null   float64
dtypes: float64(2), int64(2), object(1)
memory usage: 87.2+ KB
```

In [31]:
```
print(DDD_DID.isnull().sum())
```

```
year         0
month        0
fcn          0
fcnv_DDD    82
fcnv_DID     0
dtype: int64
```

In [32]:
```
DDD_DID[DDD_DID['fcnv_DDD'].isna()]
```

Out[32]:

	year	month	fcn	fcnv_DDD	fcnv_DID
20	2012	9	J01EA	NaN	0.00000
24	2013	1	J01EA	NaN	0.00000
27	2013	4	J01EA	NaN	0.00000
28	2013	5	J01EA	NaN	0.00000
30	2013	7	J01EA	NaN	0.00000
33	2013	10	J01EA	NaN	0.00000
34	2013	11	J01EA	NaN	0.00000
36	2014	1	J01EA	NaN	0.00000
37	2014	2	J01EA	NaN	0.00000
38	2014	3	J01EA	NaN	0.00000
40	2014	5	J01EA	NaN	0.00000
42	2014	7	J01EA	NaN	0.00000

In [33]:
```
DDD_DID[DDD_DID['fcnv_DDD'].isna()].groupby('fcn').count()
```

Out[33]:

fcn	year	month	fcnv_DDD	fcnv_DID
J01EA	23	23	0	23
J01EE	33	33	0	33
J01XB	26	26	0	26

In [34]:
```
DDD_DID=DDD_DID.copy()
print(s.shape)
sheet_512_copy=sheet_512_copy.dropna().copy()
print(sheet_512_copy.shape)
fcn_DDD=sheet_512_copy.fcnv_DDD
columns_name=['year', 'month', 'fcn', 'fcnv_DDD', 'fcnv_DID']
```

```
(1860, 5)
(1778, 5)
```

در این مرحله داده های گم شده را پیدا کرده و ان هارا از بین میبریم

In [37]:
```
jl.dump(DDD_DID,'DDD_DID_DataPreparation.pkl')
```

Out[37]:
```
['DDD_DID_DataPreparation.pkl']
```

In [ ]:
```

```