In [ ]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
%matplotlib inline
from catboost import CatBoostRegressor
```

In [4]:
```
df= pd.read_csv("H2HBABBA1822.csv")
```

In [5]:
```
df
```

Out[5]:

	business_code	cust_number	name_customer	clear_date	buisness_year	doc_id	posting_date	document_create_date	document_cr
0	U001	0200772595	SAFEW trust	NaN	2020.0	1.930774e+09	2020-04-10	20200410	
1	CA02	0140104409	LOB co	2019-07-19 00:00:00	2019.0	2.960560e+09	2019-07-06	20190706	
2	U001	0200769623	WAL-MAR llc	2019-08-13 00:00:00	2019.0	1.929691e+09	2019-08-01	20190802	
3	U001	0200769623	WAL-MAR	2019-05-13 00:00:00	2019.0	1.929234e+09	2019-05-01	20190430	
4	U001	0200752302	KROGER llc	NaN	2020.0	1.930764e+09	2020-04-08	20200408	
5	U001	0200769623	WAL-MAR foundation	2019-12-19 00:00:00	2019.0	1.930250e+09	2019-12-08	20191208	

In [6]:

```
df.dtypes
```

Out[6]:

```
business_code              object
cust_number                object
name_customer              object
clear_date                 object
buisness_year             float64
doc_id                    float64
posting_date               object
document_create_date        int64
document_create_date.1      int64
due_in_date               float64
invoice_currency           object
document type              object
posting_id                float64
area_business             float64
total_open_amount         float64
baseline_create_date      float64
cust_payment_terms         object
invoice_id                float64
isOpen                      int64
dtype: object
```

In [9]:

```
df.columns
```

Out[9]:

```
Index(['business_code', 'cust_number', 'name_customer', 'clear_date',
       'buisness_year', 'doc_id', 'posting_date', 'document_create_date',
       'document_create_date.1', 'due_in_date', 'invoice_currency',
       'document type', 'posting_id', 'area_business', 'total_open_amount',
       'baseline_create_date', 'cust_payment_terms', 'invoice_id', 'isOpen'],
      dtype='object')
```

In [11]:
```
df.isnull().sum()
```

Out[11]:
```
business_code                0
cust_number                  0
name_customer                0
clear_date               10000
buisness_year                0
doc_id                       0
posting_date                 0
document_create_date         0
document_create_date.1       0
due_in_date                  0
invoice_currency             0
document type                0
posting_id                   0
area_business            50000
total_open_amount            0
baseline_create_date         0
cust_payment_terms           0
invoice_id                   3
isOpen                       0
dtype: int64
```

In [13]:
```
# Removing the unneccesary columns from train and test sata set
df = df.drop(['business_code','cust_number','name_customer','clear_date'],axis=1)
```

In [14]:

```
df.isnull().sum()
```

Out[14]:

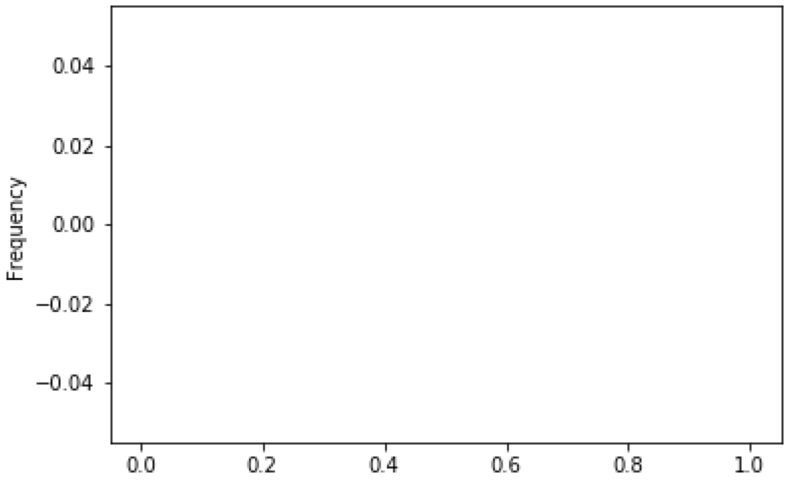
```
buisness_year              0
doc_id                     0
posting_date               0
document_create_date       0
document_create_date.1     0
due_in_date                0
invoice_currency           0
document type              0
posting_id                 0
area_business          50000
total_open_amount          0
baseline_create_date       0
cust_payment_terms         0
invoice_id                 3
isOpen                     0
dtype: int64
```

In [7]:

```
df['area_business'].plot.hist()
```

Out[7]:

```
<matplotlib.axes._subplots.AxesSubplot at 0x4918a41748>
```



In [16]:

```
corr= df.corr()
```

In [17]:
```
corr
```

Out[17]:

	buisness_year	doc_id	document_create_date	document_create_date.1	due_in_date	posting_id	area_business	total_ope
buisness_year	1.000000	-0.017986	0.977894	0.984271	0.988509	NaN	NaN	
doc_id	-0.017986	1.000000	-0.014540	-0.016459	-0.020422	NaN	NaN	
document_create_date	0.977894	-0.014540	1.000000	0.993208	0.973517	NaN	NaN	
document_create_date.1	0.984271	-0.016459	0.993208	1.000000	0.979290	NaN	NaN	
due_in_date	0.988509	-0.020422	0.973517	0.979290	1.000000	NaN	NaN	
posting_id	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
area_business	NaN	NaN	NaN	NaN	NaN	NaN	NaN	
total_open_amount	-0.003081	0.184665	-0.000125	-0.001315	-0.003668	NaN	NaN	
baseline_create_date	0.984469	-0.014626	0.992539	0.999327	0.979868	NaN	NaN	
invoice_id	-0.017293	1.000000	-0.013712	-0.015651	-0.019649	NaN	NaN	
isOpen	0.750656	-0.017144	0.760008	0.759480	0.750456	NaN	NaN	

In [18]:
```
df.isnull().sum()
```

Out[18]:
```
buisness_year                0
doc_id                       0
posting_date                 0
document_create_date         0
document_create_date.1       0
due_in_date                  0
invoice_currency             0
document type                0
posting_id                   0
area_business            50000
total_open_amount            0
baseline_create_date         0
cust_payment_terms           0
invoice_id                   3
isOpen                       0
dtype: int64
```

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

Out[20]:

	buisness_year	doc_id	document_create_date	document_create_date.1	due_in_date	posting_id	area_business	total_open_amount	ba
count	50000.000000	5.000000e+04	5.000000e+04	5.000000e+04	5.000000e+04	50000.0	0.0	5.000000e+04	
mean	2019.307320	2.010386e+09	2.019352e+07	2.019355e+07	2.019370e+07	1.0	NaN	3.245430e+04	
std	0.461388	2.804714e+08	4.502663e+03	4.488096e+03	4.479337e+03	0.0	NaN	3.965969e+04	
min	2019.000000	1.928511e+09	2.018123e+07	2.018123e+07	2.018113e+07	1.0	NaN	7.200000e-01	
25%	2019.000000	1.929341e+09	2.019051e+07	2.019051e+07	2.019052e+07	1.0	NaN	4.882028e+03	
50%	2019.000000	1.929973e+09	2.019091e+07	2.019091e+07	2.019093e+07	1.0	NaN	1.754630e+04	
75%	2020.000000	1.930619e+09	2.020013e+07	2.020013e+07	2.020022e+07	1.0	NaN	4.709228e+04	
max	2020.000000	9.500000e+09	2.020052e+07	2.020052e+07	2.020071e+07	1.0	NaN	1.010291e+06	

In [21]:
```
df['area_business'].fillna(0, inplace=True)
df['invoice_id'].fillna(0, inplace=True)
```

In [24]:
```
df['area_business'].fillna(df['area_business'].mean(), inplace=True)
```

In [26]:

```
df.isnull().sum()
```

Out[26]:

```
buisness_year             0
doc_id                    0
posting_date              0
document_create_date      0
document_create_date.1    0
due_in_date               0
invoice_currency          0
document type             0
posting_id                0
area_business             0
total_open_amount         0
baseline_create_date      0
cust_payment_terms        0
invoice_id                0
isOpen                    0
dtype: int64
```

In [ ]:

In [27]:

```
```

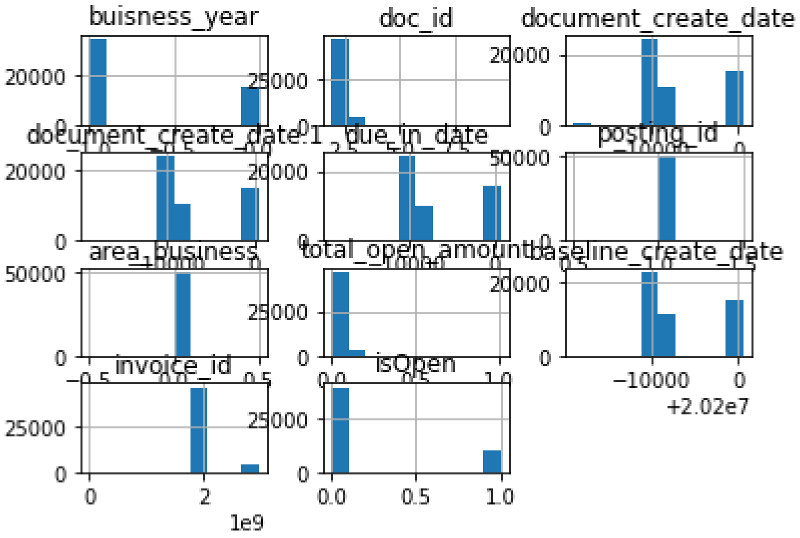
Out[27]:

	buisness_year	doc_id	posting_date	document_create_date	document_create_date.1	due_in_date	invoice_currency	document type	posting
0	2020.0	1.930774e+09	2020-04-10	20200410	20200410	20200425.0	USD	RV	
1	2019.0	2.960560e+09	2019-07-06	20190706	20190706	20190716.0	CAD	RV	
2	2019.0	1.929691e+09	2019-08-01	20190802	20190801	20190816.0	USD	RV	
3	2019.0	1.929234e+09	2019-05-01	20190430	20190501	20190516.0	USD	RV	
4	2020.0	1.930764e+09	2020-04-08	20200408	20200408	20200423.0	USD	RV	
...	...	...	...	...	...	...	...	...	
49995	2019.0	1.929542e+09	2019-06-25	20190626	20190625	20190710.0	USD	RV	
49996	2019.0	2.960520e+09	2019-01-08	20190108	20190108	20190129.0	CAD	RV	
49997	2020.0	1.930653e+09	2020-03-16	20200314	20200316	20200331.0	USD	RV	
49998	2019.0	1.930209e+09	2019-12-02	20191202	20191202	20191217.0	USD	RV	
49999	2019.0	1.929549e+09	2019-06-28	20190626	20190628	20190713.0	USD	RV	

50000 rows × 15 columns

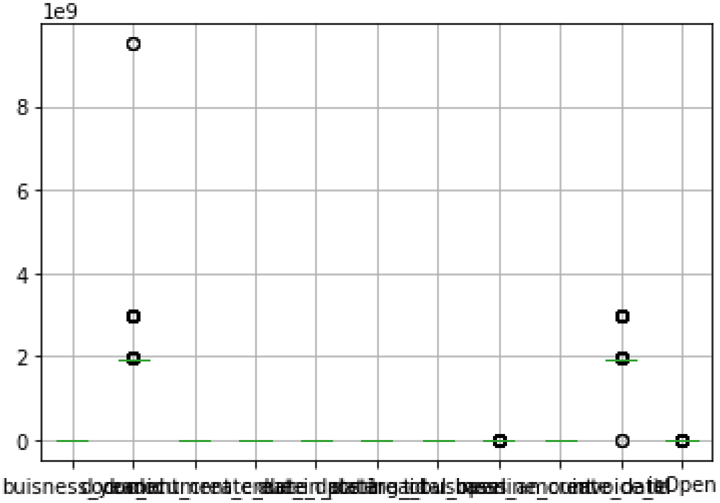
```
In [36]: df.hist()
```

```
Out[36]: array([[<AxesSubplot:title={'center':'buisness_year'}>,
                 <AxesSubplot:title={'center':'doc_id'}>,
                 <AxesSubplot:title={'center':'document_create_date'}>],
                [<AxesSubplot:title={'center':'document_create_date.1'}>,
                 <AxesSubplot:title={'center':'due_in_date'}>,
                 <AxesSubplot:title={'center':'posting_id'}>],
                [<AxesSubplot:title={'center':'area_business'}>,
                 <AxesSubplot:title={'center':'total_open_amount'}>,
                 <AxesSubplot:title={'center':'baseline_create_date'}>],
                [<AxesSubplot:title={'center':'invoice_id'}>,
                 <AxesSubplot:title={'center':'isOpen'}>, <AxesSubplot:>]],
               dtype=object)
```



In [32]:
```
df.boxplot()
```

Out[32]:
```
<AxesSubplot:>
```



In [37]:

```
df.info
```

Out[37]:

```
<bound method DataFrame.info of        buisness_year       doc_id posting_date  document_create_date  \
0             2020.0  1.930774e+09   2020-04-10              20200410   
1             2019.0  2.960560e+09   2019-07-06              20190706   
2             2019.0  1.929691e+09   2019-08-01              20190802   
3             2019.0  1.929234e+09   2019-05-01              20190430   
4             2020.0  1.930764e+09   2020-04-08              20200408   
...              ...           ...          ...                   ...   
49995         2019.0  1.929542e+09   2019-06-25              20190626   
49996         2019.0  2.960520e+09   2019-01-08              20190108   
49997         2020.0  1.930653e+09   2020-03-16              20200314   
49998         2019.0  1.930209e+09   2019-12-02              20191202   
49999         2019.0  1.929549e+09   2019-06-28              20190626   

       document_create_date.1  due_in_date invoice_currency document type  \
0                    20200410   20200425.0              USD            RV   
1                    20190706   20190716.0              CAD            RV   
2                    20190801   20190816.0              USD            RV   
3                    20190501   20190516.0              USD            RV   
4                    20200408   20200423.0              USD            RV   
...                       ...          ...              ...           ...   
49995                20190625   20190710.0              USD            RV   
49996                20190108   20190129.0              CAD            RV   
49997                20200316   20200331.0              USD            RV   
49998                20191202   20191217.0              USD            RV   
49999                20190628   20190713.0              USD            RV   

       posting_id  area_business  total_open_amount  baseline_create_date  \
0             1.0            0.0           53832.63            20200410.0   
1             1.0            0.0           87919.94            20190706.0   
2             1.0            0.0           48750.33            20190801.0   
3             1.0            0.0           15757.02            20190501.0   
4             1.0            0.0           77824.16            20200408.0   
...           ...            ...                ...                   ...   
49995         1.0            0.0           24007.65            20190625.0   
49996         1.0            0.0          316484.77            20190119.0   
49997         1.0            0.0           67018.26            20200316.0   
49998         1.0            0.0           38619.69            20191202.0   
49999         1.0            0.0           73814.57            20190628.0   
```

```
      cust_payment_terms    invoice_id  isOpen
0                   NAA8  1.930774e+09       1
1                   CA10  2.960560e+09       0
2                   NAH4  1.929691e+09       0
3                   NAH4  1.929234e+09       0
4                   NAA8  1.930764e+09       1
...                  ...           ...     ...
49995               NAA8  1.929542e+09       0
49996               CA10  2.960520e+09       0
49997               NAA8  1.930653e+09       1
49998               NAA8  1.930209e+09       0
49999               NAH4  1.929549e+09       0

[50000 rows x 15 columns]>
```

```
In [ ]:
```