In [29]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

In [2]:

```
df=pd.read_csv('job.csv')
df.head()
```

Out[2]:

	enrollee_id	city	city_development_index	gender	relevent_experience	enrolled_university	educat
0	8949	city_103	0.920	Male	Has relevent experience	no_enrollment	
1	29725	city_40	0.776	Male	No relevent experience	no_enrollment	
2	11561	city_21	0.624	NaN	No relevent experience	Full time course	
3	33241	city_115	0.789	NaN	No relevent experience	NaN	
4	666	city_162	0.767	Male	Has relevent experience	no_enrollment	

In [7]:

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

Out[7]:

```
enrollee_id                  0
city                         0
city_development_index     479
gender                    4508
relevent_experience          0
enrolled_university        386
education_level            460
major_discipline          2813
experience                  65
company_size              5938
company_type              6140
training_hours             766
target                       0
dtype: int64
```

In [8]:

```
df.isnull().mean()*100
```

Out[8]:

```
enrollee_id                0.000000
city                       0.000000
city_development_index     2.500261
gender                    23.530640
relevent_experience        0.000000
enrolled_university        2.014824
education_level            2.401086
major_discipline          14.683161
experience                 0.339284
company_size              30.994885
company_type              32.049274
training_hours             3.998330
target                     0.000000
dtype: float64
```

In [9]:

```
df.shape
```

Out[9]:

```
(19158, 13)
```

In [10]:

```
df.columns
```

Out[10]:

```
Index(['enrollee_id', 'city', 'city_development_index', 'gender',
       'relevent_experience', 'enrolled_university', 'education_level',
       'major_discipline', 'experience', 'company_size', 'company_type',
       'training_hours', 'target'],
      dtype='object')
```

In [13]:

```
# kun chai column ma 5% vanda kam data missing cha tyo khojeko
col=[]
for colum in df.columns:
    if df[colum].isnull().mean()>0 and df[colum].isnull().mean()<0.05:
        col.append(colum)
col
```

Out[13]:

```
['city_development_index',
 'enrolled_university',
 'education_level',
 'experience',
 'training_hours']
```

In [15]:

```
df[col].sample(7)
```

Out[15]:

	city_development_index	enrolled_university	education_level	experience	training_hours
1972	0.884	Part time course	Graduate	20.0	6.0
6585	0.920	no_enrollment	Graduate	3.0	125.0
17611	0.855	Full time course	High School	4.0	17.0
9019	0.924	no_enrollment	Graduate	14.0	26.0
7746	0.624	NaN	NaN	5.0	98.0
6712	0.487	no_enrollment	Masters	19.0	52.0
10197	0.897	Full time course	Masters	5.0	86.0

In [20]:

```
df[col].isna().sum()
```

Out[20]:

```
city_development_index    479
enrolled_university       386
education_level           460
experience                 65
training_hours            766
dtype: int64
```

In [22]:

```
total=0
for x in col:
    total=total+df[x].isna().sum()
total
```

Out[22]:

```
2156
```

In [18]:

```
len(df)
```

Out[18]:

```
19158
```

In [24]:

```
1-(2156/19158) # yedi null value vayeko column hatayo vane 88% data baki huneracha
```

Out[24]:

```
0.8874621568013362
```

In [26]:

```
new_data=df[col].dropna()  # 5% data missing dataframe bata all null remove gardeko
df.shape, new_data.shape
```

Out[26]:

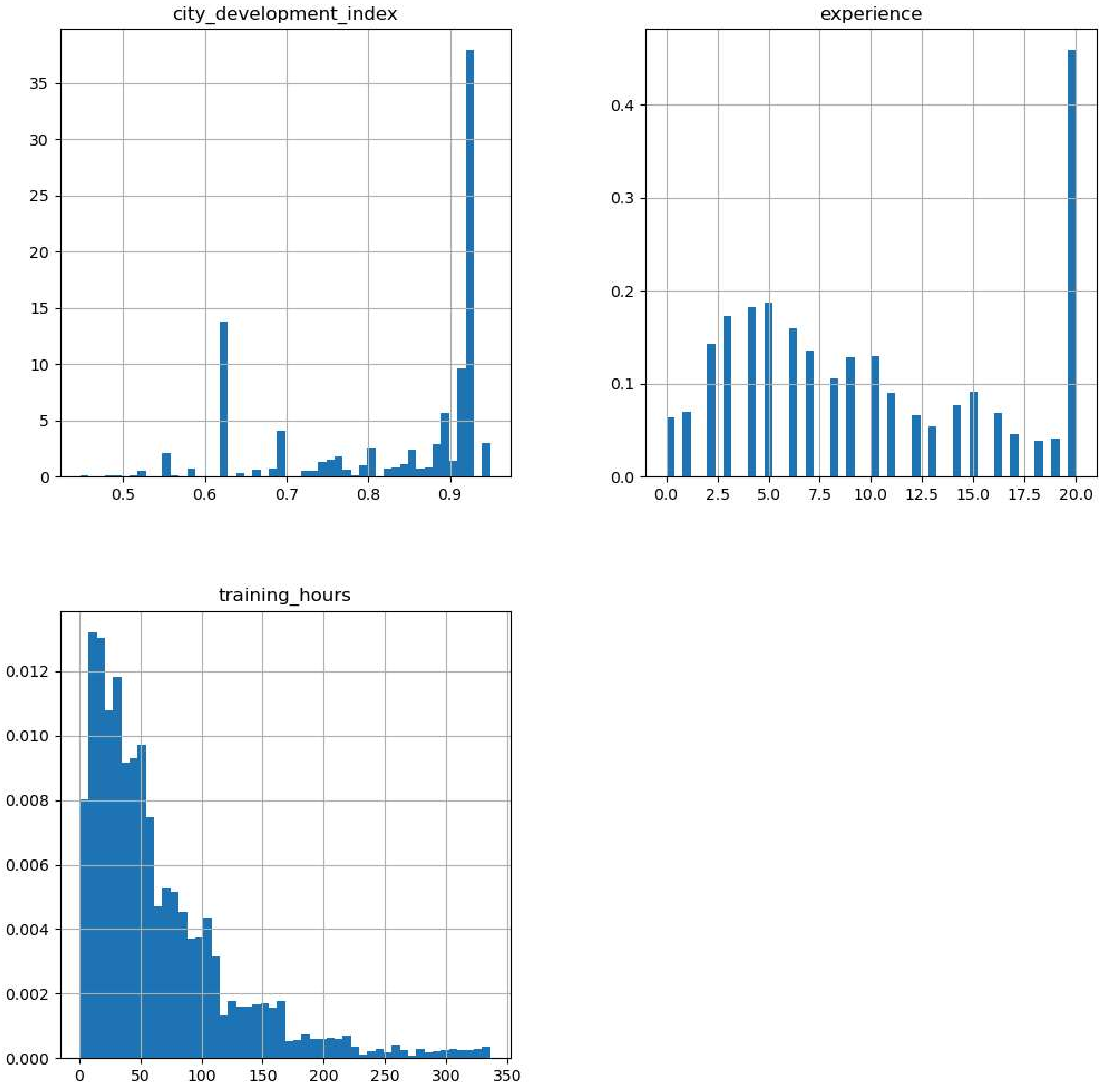
```
((19158, 13), (17182, 5))
```

In [27]:

```
#histogram plot
```

In [30]:

```
new_data.hist(bins=50, density=True,figsize=(12,12))
plt.show()
```



In [36]:

```
case1=df['enrolled_university'].value_counts()/len(df)
case1
```

Out[36]:

```
no_enrollment       0.721213
Full time course    0.196106
Part time course    0.062533
Name: enrolled_university, dtype: float64
```

In [37]:

```
case2=new_data['enrolled_university'].value_counts()/len(new_data)
case2
```

Out[37]:

```
no_enrollment       0.735188
Full time course    0.200733
Part time course    0.064079
Name: enrolled_university, dtype: float64
```

In [43]:

```
comparision=pd.concat([case1,case2],names=['case1','case2'],axis=1)
comparision
```

Out[43]:

	enrolled_university	enrolled_university
no_enrollment	0.721213	0.735188
Full time course	0.196106	0.200733
Part time course	0.062533	0.064079

If ratio is same before removing missing value and after removing missing value then it is missing value at random. If ratio is not same then we cannot remove missing value ,rather we should try to fill those value with imputation techniques

In [ ]:

```
```