G. H. Raisoni College of Engineering, Nagpur

(An Autonomous Institution)

Department of Artificial Intelligent

Class Assessment Examination-I Winter - 2020

Subject : Data Pre-processing

[Time: 2 hr.] **[Max. Marks:20]**

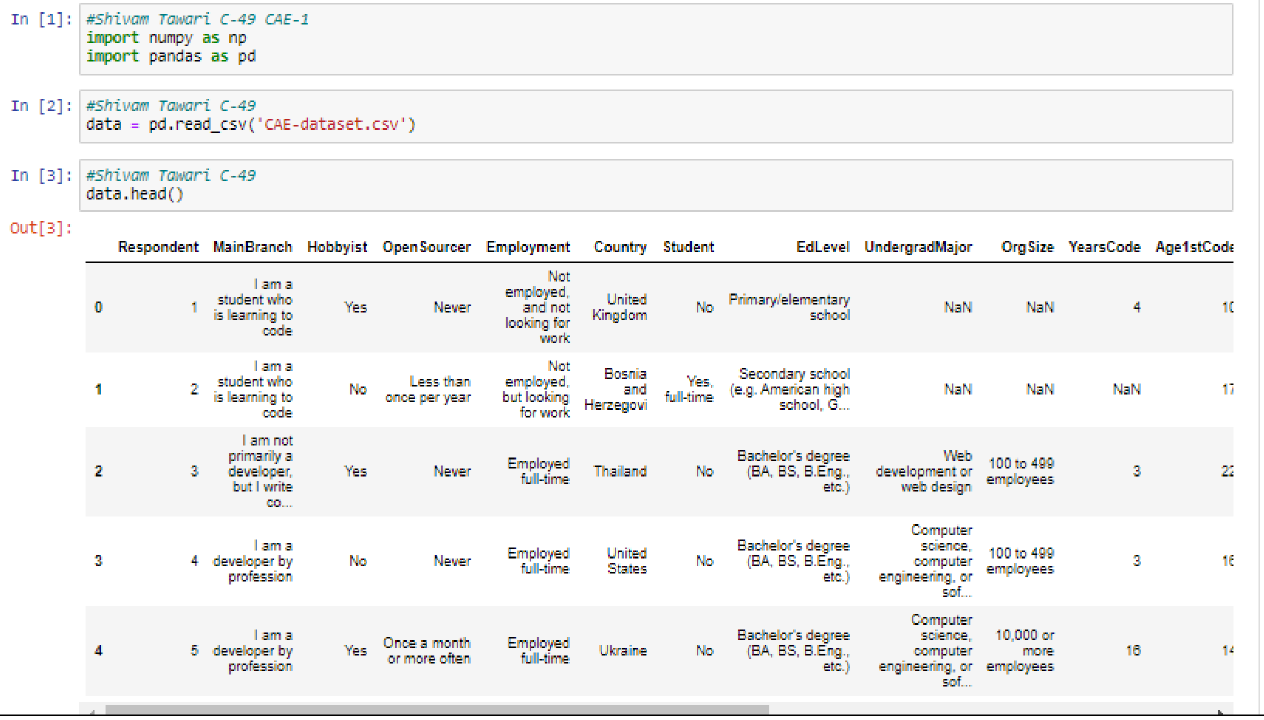
Name of student : Shivam Tawari **Roll No: C-49**

Instructions to Candidate:

1) [CO1/CO2/CO…] at the beginning of question / sub-question indicates the course outcome related to the question.
2) All questions carry marks as indicated.
3) All questions are **compulsory**.

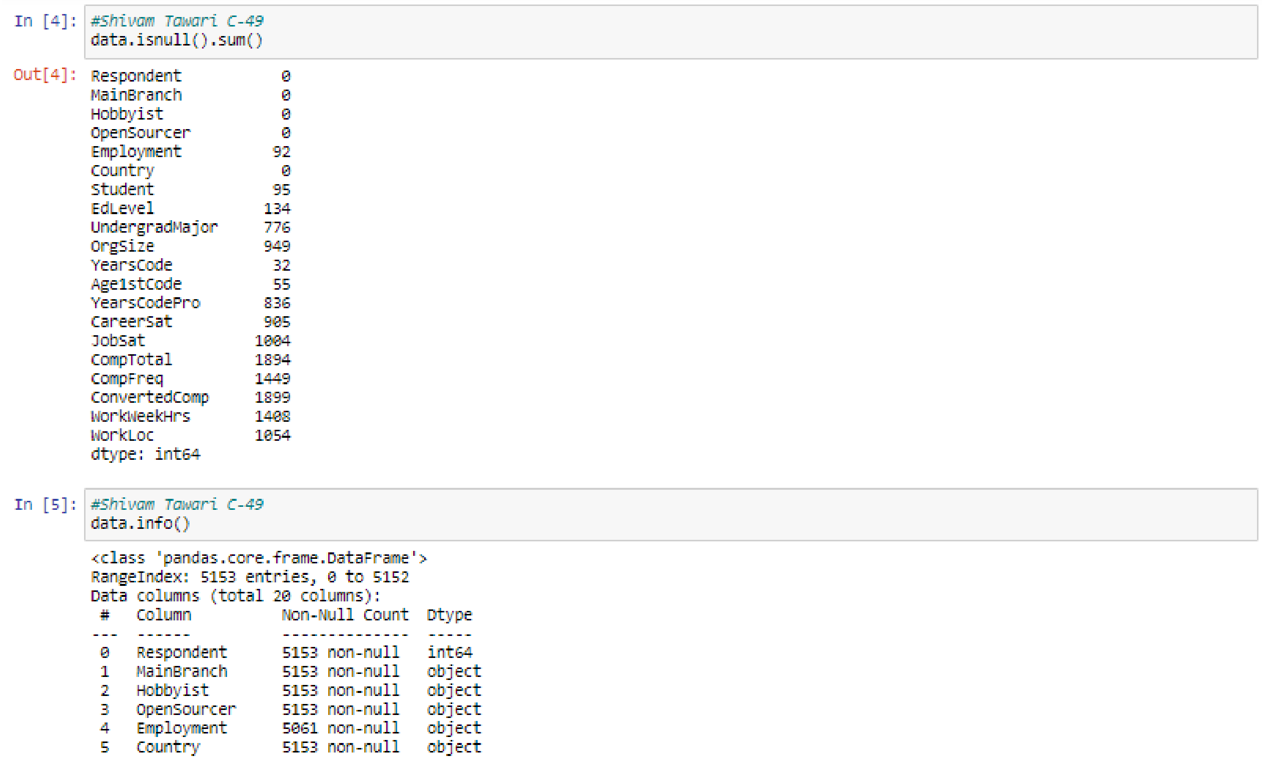
				Marks
1			Based on provided dataset , perform the following task	
	A		Properly handling Null Values	**2**

Ans:



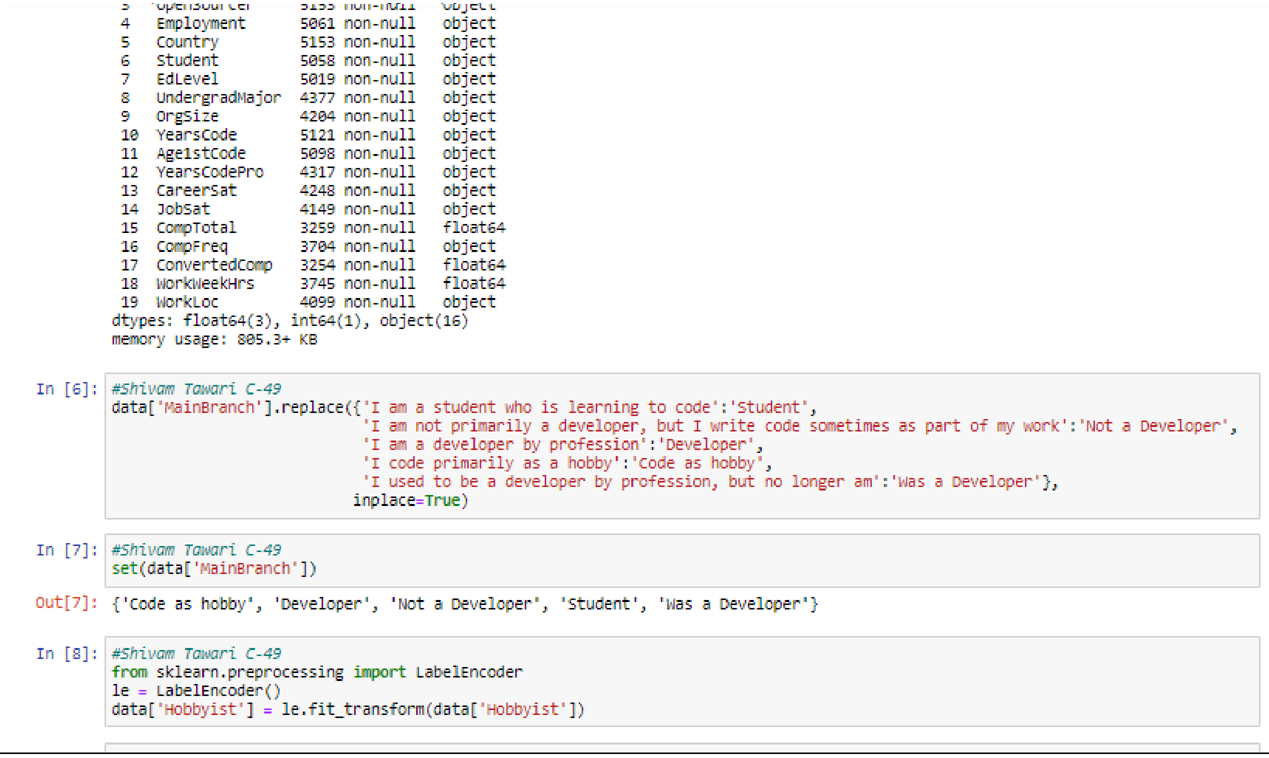
```
In [1]: #Shivam Tawari C-49 CAE-1
        import numpy as np
        import pandas as pd

In [2]: #Shivam Tawari C-49
        data = pd.read_csv('CAE-dataset.csv')

In [3]: #Shivam Tawari C-49
        data.head()

Out[3]:
```

	Respondent	MainBranch	Hobbyist	OpenSourcer	Employment	Country	Student	EdLevel	UndergradMajor	OrgSize	YearsCode	Age1stCode
0	1	I am a student who is learning to code	Yes	Never	Not employed, and not looking for work	United Kingdom	No	Primary/elementary school	NaN	NaN	4	10
1	2	I am a student who is learning to code	No	Less than once per year	Not employed, but looking for work	Bosnia and Herzegovi	Yes, full-time	Secondary school (e.g. American high school, G...	NaN	NaN	NaN	17
2	3	I am not primarily a developer, but I write co...	Yes	Never	Employed full-time	Thailand	No	Bachelor's degree (BA, BS, B.Eng., etc.)	Web development or web design	100 to 499 employees	3	22
3	4	I am a developer by profession	No	Never	Employed full-time	United States	No	Bachelor's degree (BA, BS, B.Eng., etc.)	Computer science, computer engineering, or sof...	100 to 499 employees	3	16
4	5	I am a developer by profession	Yes	Once a month or more often	Employed full-time	Ukraine	No	Bachelor's degree (BA, BS, B.Eng., etc.)	Computer science, computer engineering, or sof...	10,000 or more employees	16	14



```
In [4]: #Shivam Tawari C-49
        data.isnull().sum()

Out[4]: Respondent          0
        MainBranch          0
        Hobbyist            0
        OpenSourcer         0
        Employment         92
        Country             0
        Student            95
        EdLevel           134
        UndergradMajor    776
        OrgSize           949
        YearsCode          32
        Age1stCode         55
        YearsCodePro      836
        CareerSat         905
        JobSat           1004
        CompTotal        1894
        CompFreq         1449
        ConvertedComp    1899
        WorkWeekHrs      1408
        WorkLoc          1054
        dtype: int64
```

```
In [5]: #Shivam Tawari C-49
        data.info()

        <class 'pandas.core.frame.DataFrame'>
        RangeIndex: 5153 entries, 0 to 5152
        Data columns (total 20 columns):
         #   Column          Non-Null Count  Dtype
        ---  ------          --------------  -----
         0   Respondent      5153 non-null   int64
         1   MainBranch      5153 non-null   object
         2   Hobbyist        5153 non-null   object
         3   OpenSourcer     5153 non-null   object
         4   Employment      5061 non-null   object
         5   Country         5153 non-null   object
```



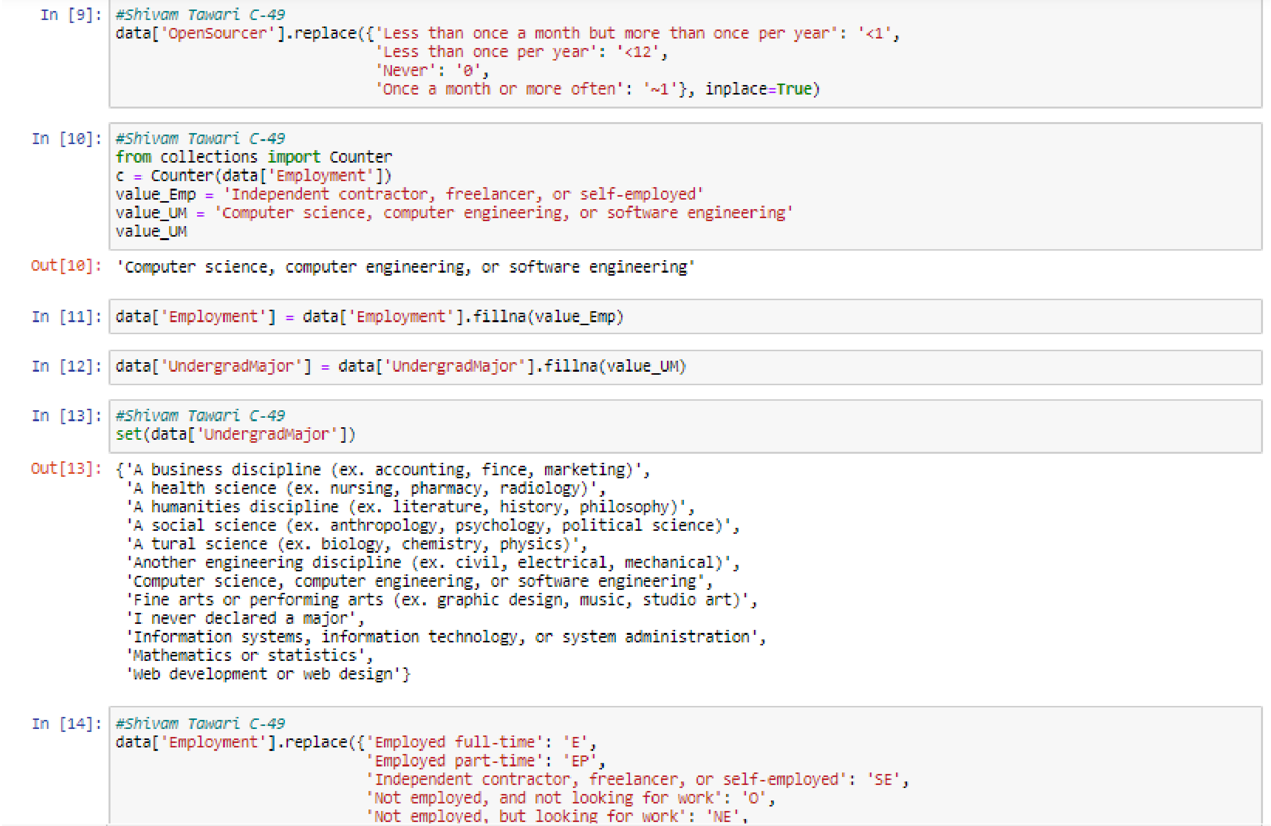
```
         3   OpenSourcer     5153 non-null   object
         4   Employment      5061 non-null   object
         5   Country         5153 non-null   object
```

```
         6   Student         5058 non-null   object
         7   EdLevel         5019 non-null   object
         8   UndergradMajor  4377 non-null   object
         9   OrgSize         4204 non-null   object
         10  YearsCode       5121 non-null   object
         11  Age1stCode      5098 non-null   object
         12  YearsCodePro    4317 non-null   object
         13  CareerSat       4248 non-null   object
         14  JobSat          4149 non-null   object
         15  CompTotal       3259 non-null   float64
         16  CompFreq        3704 non-null   object
         17  ConvertedComp   3254 non-null   float64
         18  WorkWeekHrs     3745 non-null   float64
         19  WorkLoc         4099 non-null   object
        dtypes: float64(3), int64(1), object(16)
        memory usage: 805.3+ KB
```

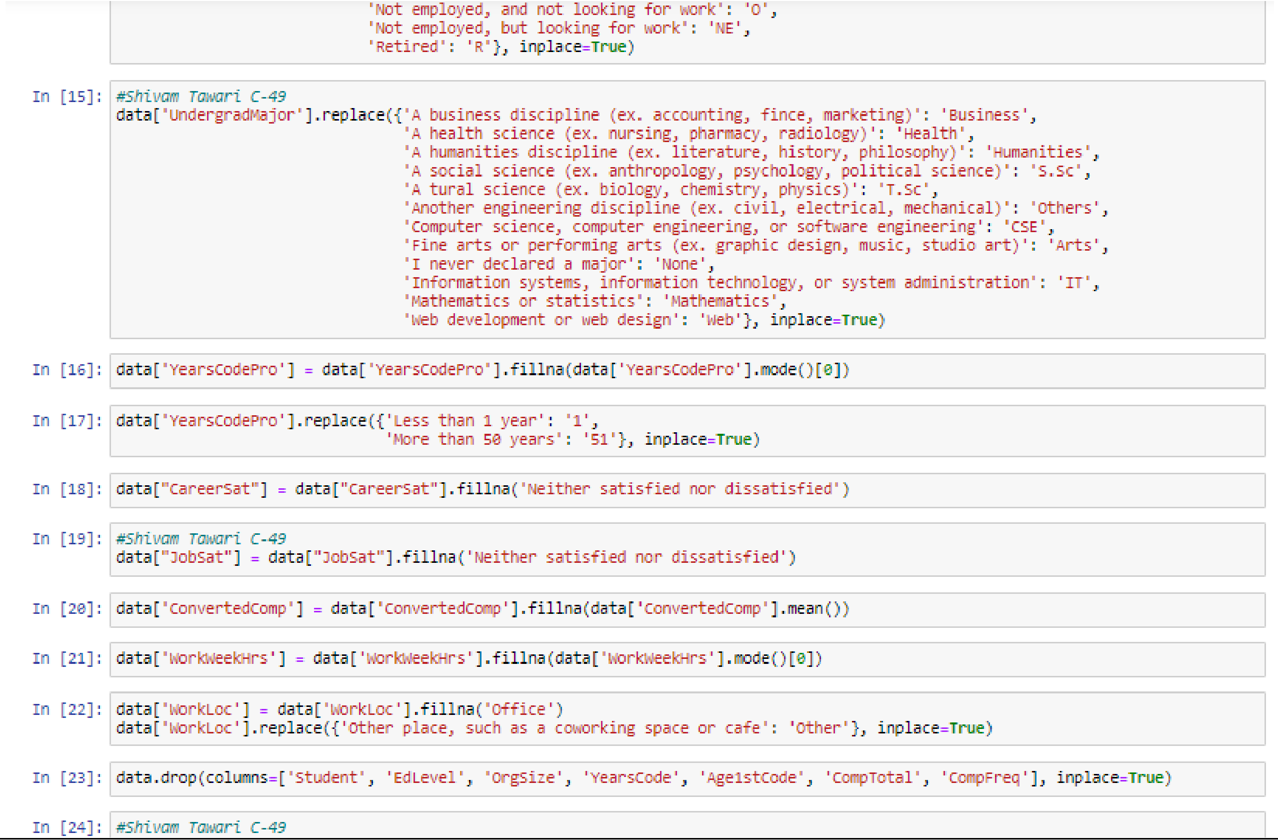
```
In [6]: #Shivam Tawari C-49
        data['MainBranch'].replace({'I am a student who is learning to code':'Student',
                                    'I am not primarily a developer, but I write code sometimes as part of my work':'Not a Developer',
                                    'I am a developer by profession':'Developer',
                                    'I code primarily as a hobby':'Code as hobby',
                                    'I used to be a developer by profession, but no longer am':'Was a Developer'},
                                   inplace=True)
```

```
In [7]: #Shivam Tawari C-49
        set(data['MainBranch'])

Out[7]: {'Code as hobby', 'Developer', 'Not a Developer', 'Student', 'Was a Developer'}
```

```
In [8]: #Shivam Tawari C-49
        from sklearn.preprocessing import LabelEncoder
        le = LabelEncoder()
        data['Hobbyist'] = le.fit_transform(data['Hobbyist'])
```

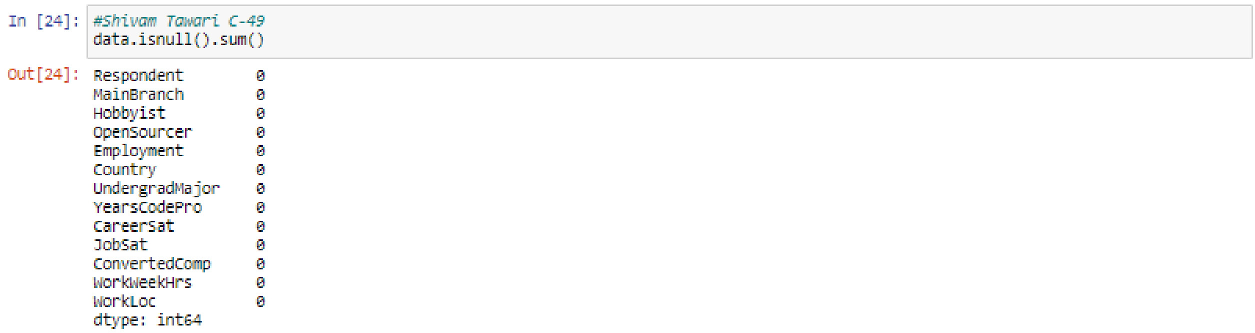


```
In [9]:  #Shivam Tawari C-49
         data['OpenSourcer'].replace({'Less than once a month but more than once per year': '<1',
                                      'Less than once per year': '<12',
                                      'Never': '0',
                                      'Once a month or more often': '~1'}, inplace=True)

In [10]: #Shivam Tawari C-49
         from collections import Counter
         c = Counter(data['Employment'])
         value_Emp = 'Independent contractor, freelancer, or self-employed'
         value_UM = 'Computer science, computer engineering, or software engineering'
         value_UM

Out[10]: 'Computer science, computer engineering, or software engineering'

In [11]: data['Employment'] = data['Employment'].fillna(value_Emp)

In [12]: data['UndergradMajor'] = data['UndergradMajor'].fillna(value_UM)

In [13]: #Shivam Tawari C-49
         set(data['UndergradMajor'])

Out[13]: {'A business discipline (ex. accounting, fince, marketing)',
          'A health science (ex. nursing, pharmacy, radiology)',
          'A humanities discipline (ex. literature, history, philosophy)',
          'A social science (ex. anthropology, psychology, political science)',
          'A tural science (ex. biology, chemistry, physics)',
          'Another engineering discipline (ex. civil, electrical, mechanical)',
          'Computer science, computer engineering, or software engineering',
          'Fine arts or performing arts (ex. graphic design, music, studio art)',
          'I never declared a major',
          'Information systems, information technology, or system administration',
          'Mathematics or statistics',
          'Web development or web design'}

In [14]: #Shivam Tawari C-49
         data['Employment'].replace({'Employed full-time': 'E',
                                     'Employed part-time': 'EP',
                                     'Independent contractor, freelancer, or self-employed': 'SE',
                                     'Not employed, and not looking for work': 'O',
                                     'Not employed, but looking for work': 'NE',
```



'Not employed, and not looking for work': 'O',
'Not employed, but looking for work': 'NE',

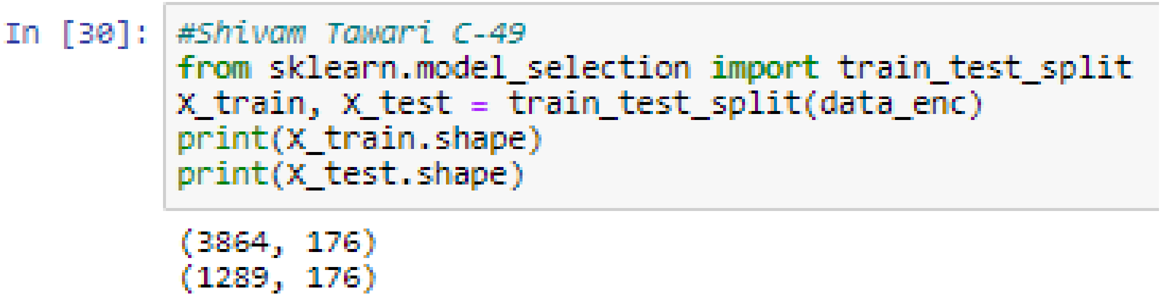
```
                                     'Retired': 'R'}, inplace=True)

In [15]: #Shivam Tawari C-49
         data['UndergradMajor'].replace({'A business discipline (ex. accounting, fince, marketing)': 'Business',
                                         'A health science (ex. nursing, pharmacy, radiology)': 'Health',
                                         'A humanities discipline (ex. literature, history, philosophy)': 'Humanities',
                                         'A social science (ex. anthropology, psychology, political science)': 'S.Sc',
                                         'A tural science (ex. biology, chemistry, physics)': 'T.Sc',
                                         'Another engineering discipline (ex. civil, electrical, mechanical)': 'Others',
                                         'Computer science, computer engineering, or software engineering': 'CSE',
                                         'Fine arts or performing arts (ex. graphic design, music, studio art)': 'Arts',
                                         'I never declared a major': 'None',
                                         'Information systems, information technology, or system administration': 'IT',
                                         'Mathematics or statistics': 'Mathematics',
                                         'Web development or web design': 'Web'}, inplace=True)

In [16]: data['YearsCodePro'] = data['YearsCodePro'].fillna(data['YearsCodePro'].mode()[0])

In [17]: data['YearsCodePro'].replace({'Less than 1 year': '1',
                                       'More than 50 years': '51'}, inplace=True)

In [18]: data["CareerSat"] = data["CareerSat"].fillna('Neither satisfied nor dissatisfied')

In [19]: #Shivam Tawari C-49
         data["JobSat"] = data["JobSat"].fillna('Neither satisfied nor dissatisfied')

In [20]: data['ConvertedComp'] = data['ConvertedComp'].fillna(data['ConvertedComp'].mean())

In [21]: data['WorkWeekHrs'] = data['WorkWeekHrs'].fillna(data['WorkWeekHrs'].mode()[0])

In [22]: data['WorkLoc'] = data['WorkLoc'].fillna('Office')
         data['WorkLoc'].replace({'Other place, such as a coworking space or cafe': 'Other'}, inplace=True)

In [23]: data.drop(columns=['Student', 'EdLevel', 'OrgSize', 'YearsCode', 'Age1stCode', 'CompTotal', 'CompFreq'], inplace=True)

In [24]: #Shivam Tawari C-49
```



```
In [24]: #Shivam Tawari C-49
         data.isnull().sum()

Out[24]: Respondent        0
         MainBranch        0
         Hobbyist          0
         OpenSourcer       0
         Employment        0
         Country           0
         UndergradMajor    0
         YearsCodePro      0
         CareerSat         0
         JobSat            0
         ConvertedComp     0
         WorkWeekHrs       0
         WorkLoc           0
         dtype: int64
```

B Performing random sampling 3

Ans:

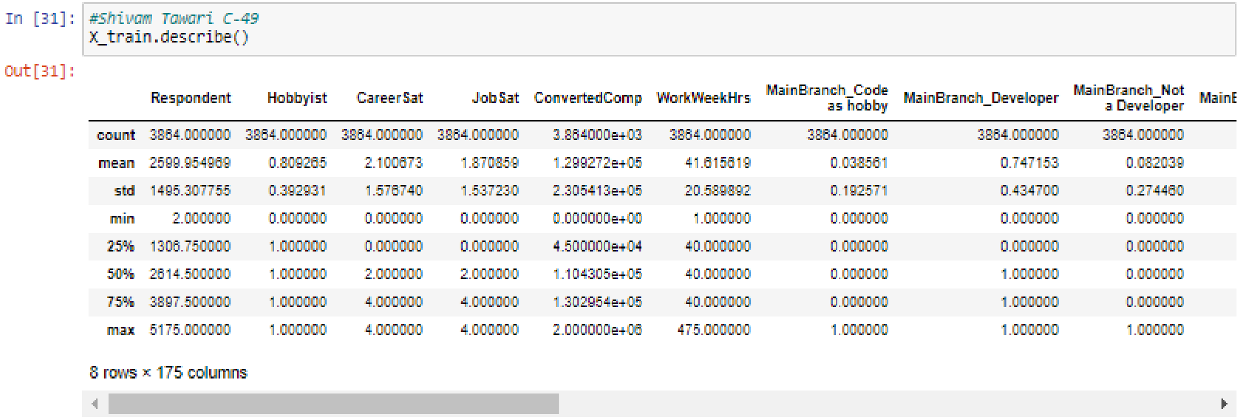


```
In [30]: #Shivam Tawari C-49
         from sklearn.model_selection import train_test_split
         X_train, X_test = train_test_split(data_enc)
         print(X_train.shape)
         print(X_test.shape)

         (3864, 176)
         (1289, 176)
```

2

A Mean, Median, standard Deviation for numeric attribute 3

Ans :



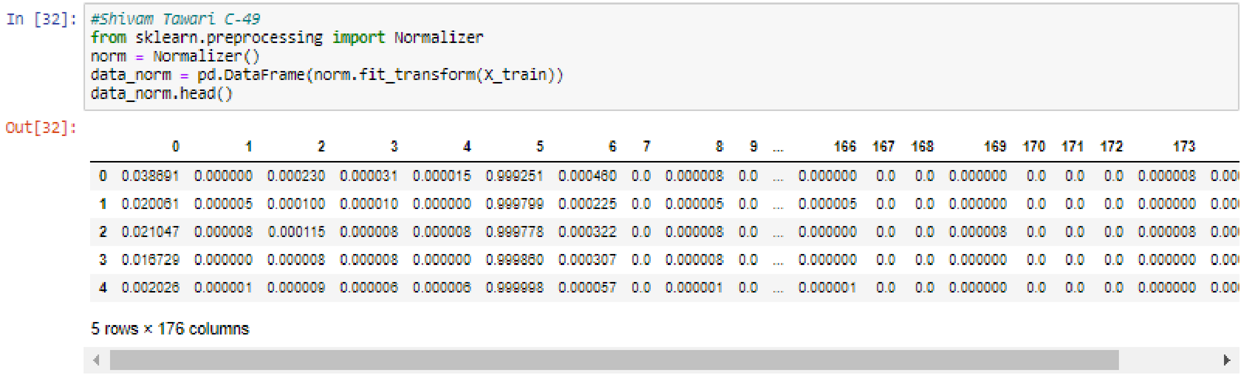
```
In [31]: #Shivam Tawari C-49
         X_train.describe()

Out[31]:
```

	Respondent	Hobbyist	CareerSat	JobSat	ConvertedComp	WorkWeekHrs	MainBranch_Code as hobby	MainBranch_Developer	MainBranch_Not a Developer	MainE
count	3864.000000	3864.000000	3864.000000	3864.000000	3.864000e+03	3864.000000	3864.000000	3864.000000	3864.000000	
mean	2599.954969	0.809265	2.100673	1.870859	1.299272e+05	41.615619	0.038561	0.747153	0.082039	
std	1495.307755	0.392931	1.576740	1.537230	2.305413e+05	20.589892	0.192571	0.434700	0.274460	
min	2.000000	0.000000	0.000000	0.000000	0.000000e+00	1.000000	0.000000	0.000000	0.000000	
25%	1306.750000	1.000000	0.000000	0.000000	4.500000e+04	40.000000	0.000000	0.000000	0.000000	
50%	2614.500000	1.000000	2.000000	2.000000	1.104305e+05	40.000000	0.000000	1.000000	0.000000	
75%	3897.500000	1.000000	4.000000	4.000000	1.302954e+05	40.000000	0.000000	1.000000	0.000000	
max	5175.000000	1.000000	4.000000	4.000000	2.000000e+06	475.000000	1.000000	1.000000	1.000000	

8 rows × 175 columns

B Working with Normalization 3

Ans :



```
In [32]: #Shivam Tawari C-49
         from sklearn.preprocessing import Normalizer
         norm = Normalizer()
         data_norm = pd.DataFrame(norm.fit_transform(X_train))
         data_norm.head()

Out[32]:
```

	0	1	2	3	4	5	6	7	8	9	...	166	167	168	169	170	171	172	173	
0	0.038691	0.000000	0.000230	0.000031	0.000015	0.999251	0.000460	0.0	0.000008	0.0	...	0.000000	0.0	0.0	0.000000	0.0	0.0	0.0	0.000008	0.00
1	0.020061	0.000005	0.000100	0.000010	0.000000	0.999799	0.000225	0.0	0.000005	0.0	...	0.000005	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00
2	0.021047	0.000008	0.000115	0.000008	0.000008	0.999778	0.000322	0.0	0.000008	0.0	...	0.000000	0.0	0.0	0.000008	0.0	0.0	0.0	0.000008	0.00
3	0.016729	0.000000	0.000008	0.000008	0.000000	0.999860	0.000307	0.0	0.000008	0.0	...	0.000000	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00
4	0.002026	0.000001	0.000009	0.000006	0.000006	0.999998	0.000057	0.0	0.000001	0.0	...	0.000001	0.0	0.0	0.000000	0.0	0.0	0.0	0.000000	0.00

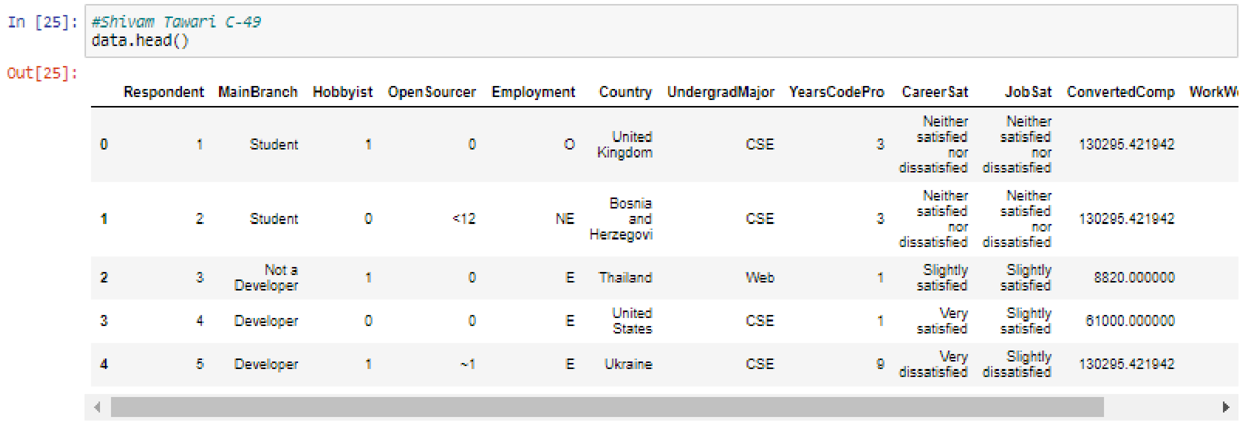
5 rows × 176 columns

C

Implementation of Discretization

3

Ans :



In [25]:
```
#Shivam Tawari C-49
data.head()
```

Out[25]:

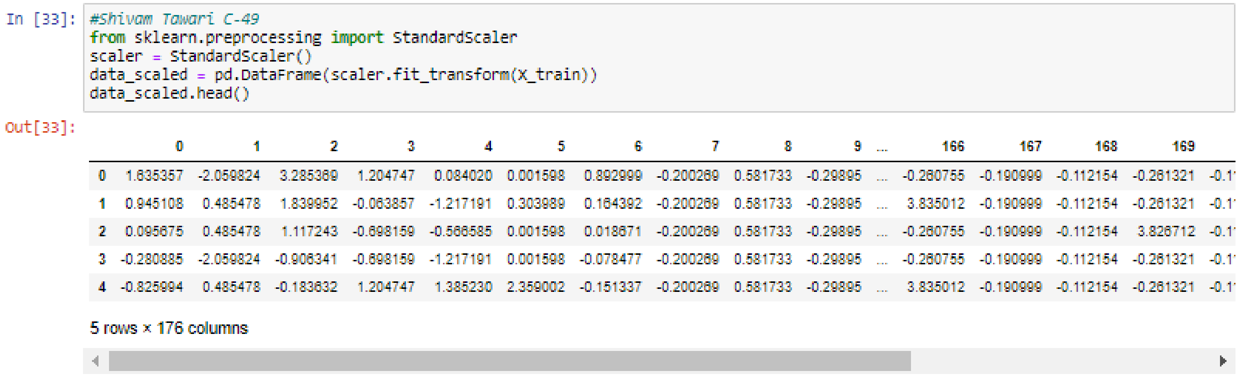
	Respondent	MainBranch	Hobbyist	OpenSourcer	Employment	Country	UndergradMajor	YearsCodePro	CareerSat	JobSat	ConvertedComp	WorkW
0	1	Student	1	0	O	United Kingdom	CSE	3	Neither satisfied nor dissatisfied	Neither satisfied nor dissatisfied	130295.421942	
1	2	Student	0	<12	NE	Bosnia and Herzegovi	CSE	3	Neither satisfied nor dissatisfied	Neither satisfied nor dissatisfied	130295.421942	
2	3	Not a Developer	1	0	E	Thailand	Web	1	Slightly satisfied	Slightly satisfied	8820.000000	
3	4	Developer	0	0	E	United States	CSE	1	Very satisfied	Slightly satisfied	61000.000000	
4	5	Developer	1	~1	E	Ukraine	CSE	9	Very dissatisfied	Slightly dissatisfied	130295.421942	

D

Apply standardization to given dataset

3

Ans :



In [33]:
```
#Shivam Tawari C-49
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
data_scaled = pd.DataFrame(scaler.fit_transform(X_train))
data_scaled.head()
```

Out[33]:

	0	1	2	3	4	5	6	7	8	9	...	166	167	168	169	
0	1.635357	-2.059824	3.285369	1.204747	0.084020	0.001598	0.892999	-0.200269	0.581733	-0.29895	...	-0.260755	-0.190999	-0.112154	-0.261321	-0.1
1	0.945108	0.485478	1.839952	-0.063857	-1.217191	0.303989	0.164392	-0.200269	0.581733	-0.29895	...	3.835012	-0.190999	-0.112154	-0.261321	-0.1
2	0.095675	0.485478	1.117243	-0.698159	-0.566585	0.001598	0.018671	-0.200269	0.581733	-0.29895	...	-0.260755	-0.190999	-0.112154	3.826712	-0.1
3	-0.280885	-2.059824	-0.906341	-0.698159	-1.217191	0.001598	-0.078477	-0.200269	0.581733	-0.29895	...	-0.260755	-0.190999	-0.112154	-0.261321	-0.1
4	-0.825994	0.485478	-0.183632	1.204747	1.385230	2.359002	-0.151337	-0.200269	0.581733	-0.29895	...	3.835012	-0.190999	-0.112154	-0.261321	-0.1

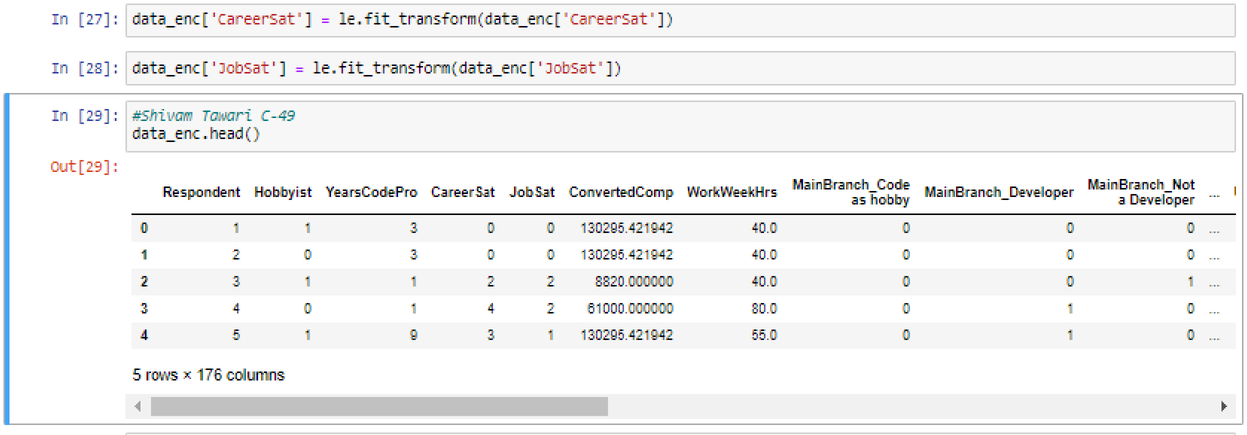
5 rows × 176 columns

E

Apply Binarization (One hot encoding)

3

Ans :



In [27]:
```
data_enc['CareerSat'] = le.fit_transform(data_enc['CareerSat'])
```

In [28]:
```
data_enc['JobSat'] = le.fit_transform(data_enc['JobSat'])
```

In [29]:
```
#Shivam Tawari C-49
data_enc.head()
```

Out[29]:

	Respondent	Hobbyist	YearsCodePro	CareerSat	JobSat	ConvertedComp	WorkWeekHrs	MainBranch_Code as hobby	MainBranch_Developer	MainBranch_Not a Developer	...
0	1	1	3	0	0	130295.421942	40.0	0	0	0	...
1	2	0	3	0	0	130295.421942	40.0	0	0	0	...
2	3	1	1	2	2	8820.000000	40.0	0	0	1	...
3	4	0	1	4	2	61000.000000	80.0	0	1	0	...
4	5	1	9	3	1	130295.421942	55.0	0	1	0	...

5 rows × 176 columns