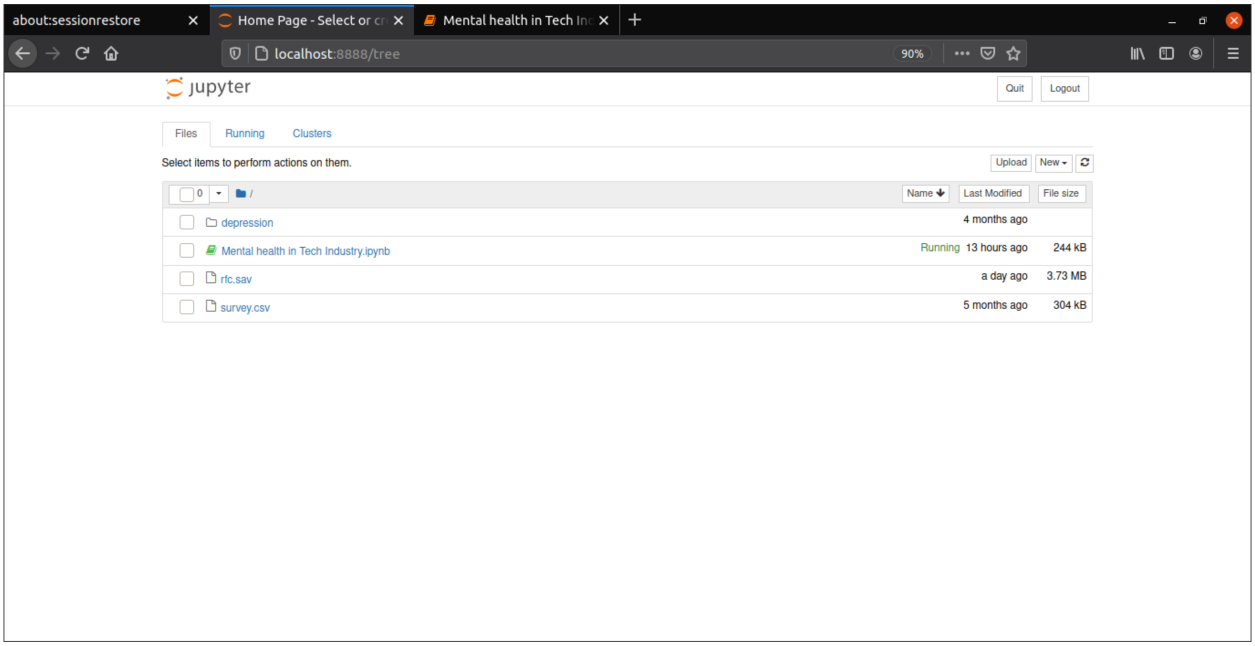
Name: Colin Mburugu

Batch: LISUMO1

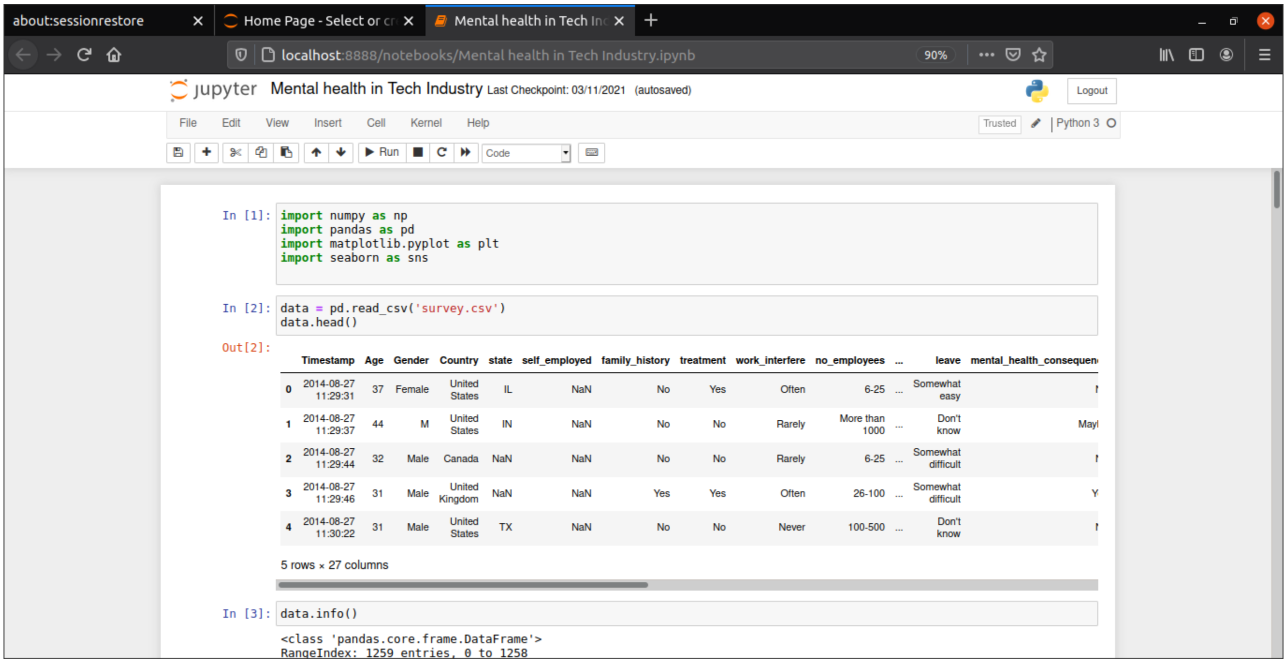
Submission_date : 4/07/2021

Submission_to : Data Glacier



Select items to perform actions on them.

Name		Last Modified	File size
depression		4 months ago	
Mental health in Tech Industry.ipynb	Running	13 hours ago	244 kB
rfc.sav		a day ago	3.73 MB
survey.csv		5 months ago	304 kB


about:sessionrestore
Home Page - Select or cr
Mental health in Tech Inc
localhost:8888/notebooks/Mental health in Tech Industry.ipynb
90%
jupyter
Mental health in Tech Industry Last Checkpoint: 03/11/2021 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Help
Trusted
Python 3
Run
Code

In [1]:
```
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
```

In [2]:
```
data = pd.read_csv('survey.csv')
data.head()
```

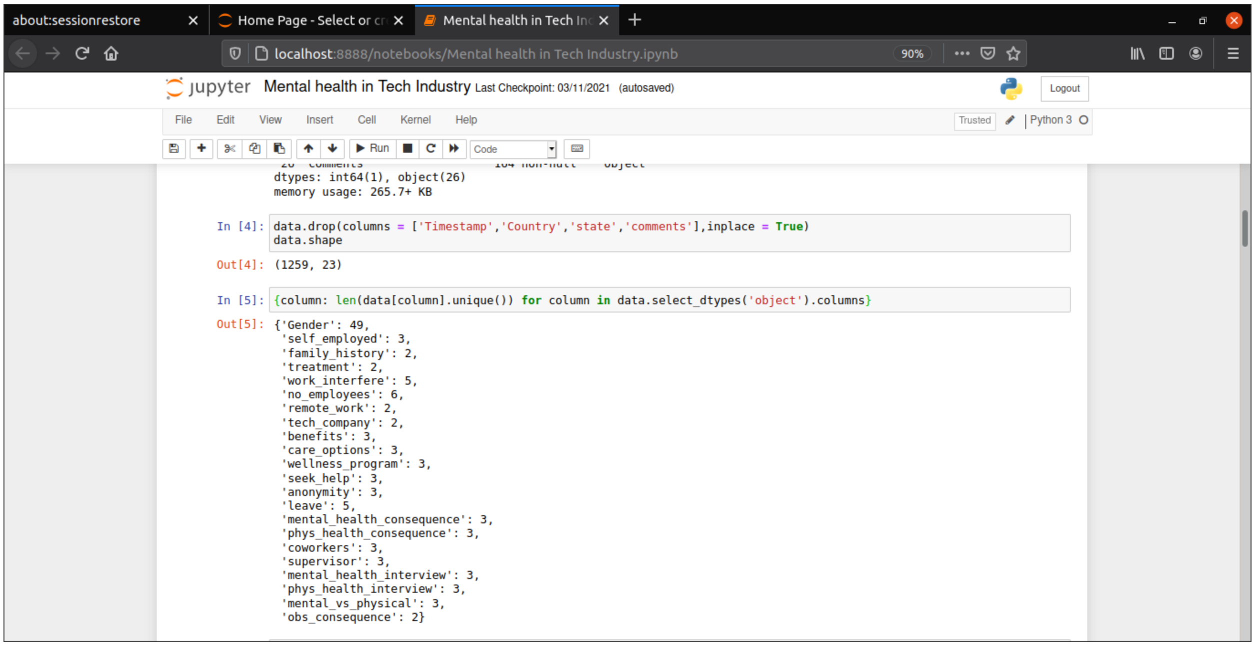
Out[2]:

	Timestamp	Age	Gender	Country	state	self_employed	family_history	treatment	work_interfere	no_employees	...	leave	mental_health_consequen
0	2014-08-27 11:29:31	37	Female	United States	IL	NaN	No	Yes	Often	6-25	...	Somewhat easy	N
1	2014-08-27 11:29:37	44	M	United States	IN	NaN	No	No	Rarely	More than 1000	...	Don't know	May
2	2014-08-27 11:29:44	32	Male	Canada	NaN	NaN	No	No	Rarely	6-25	...	Somewhat difficult	N
3	2014-08-27 11:29:46	31	Male	United Kingdom	NaN	NaN	Yes	Yes	Often	26-100	...	Somewhat difficult	Y
4	2014-08-27 11:30:22	31	Male	United States	TX	NaN	No	No	Never	100-500	...	Don't know	N

5 rows × 27 columns

In [3]:
```
data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1259 entries, 0 to 1258
```


about:sessionrestore
Home Page - Select or cr
Mental health in Tech Inc
localhost:8888/notebooks/Mental health in Tech Industry.ipynb
90%
jupyter
Mental health in Tech Industry Last Checkpoint: 03/11/2021 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Help
Trusted
Python 3
Run
Code

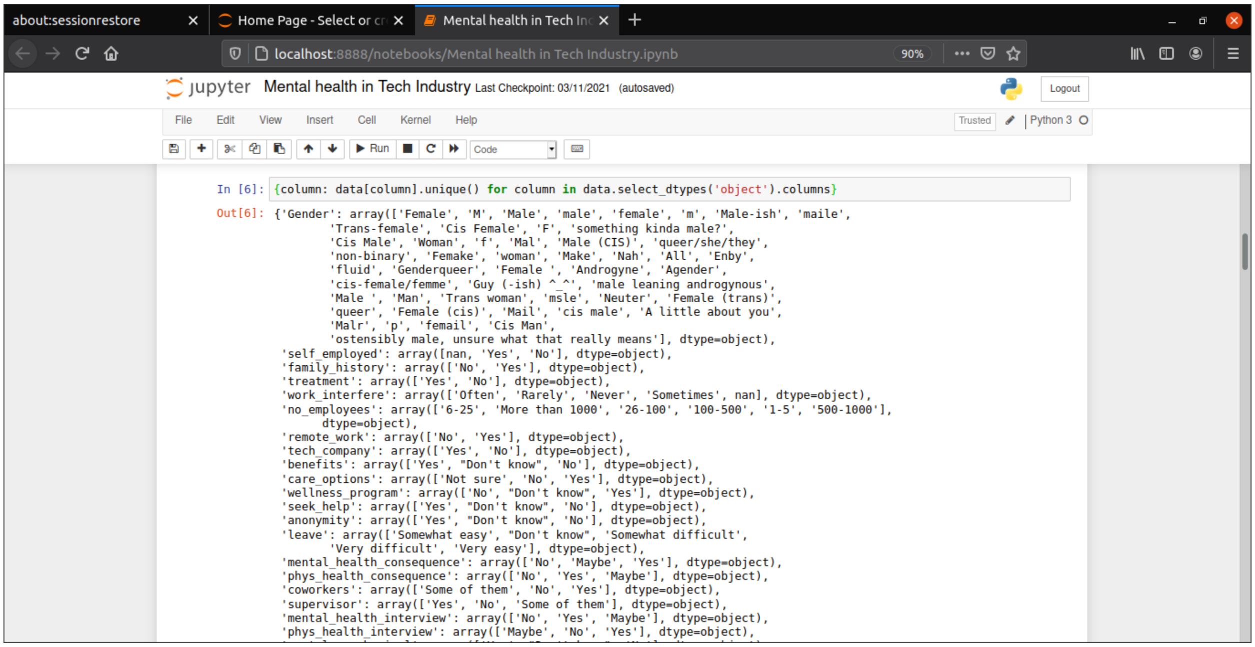
```
 26  comments                   164 non-null    object
dtypes: int64(1), object(26)
memory usage: 265.7+ KB
```

In [4]:
```
data.drop(columns = ['Timestamp','Country','state','comments'],inplace = True)
data.shape
```

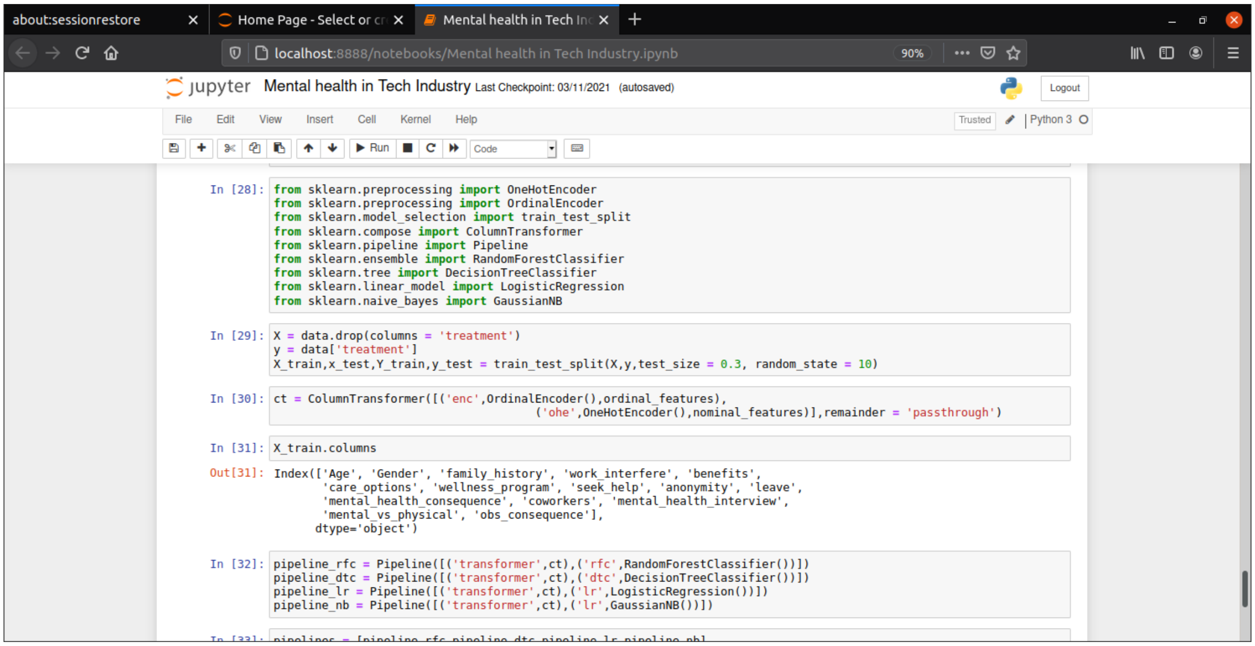
Out[4]:
```
(1259, 23)
```

In [5]:
```
{column: len(data[column].unique()) for column in data.select_dtypes('object').columns}
```

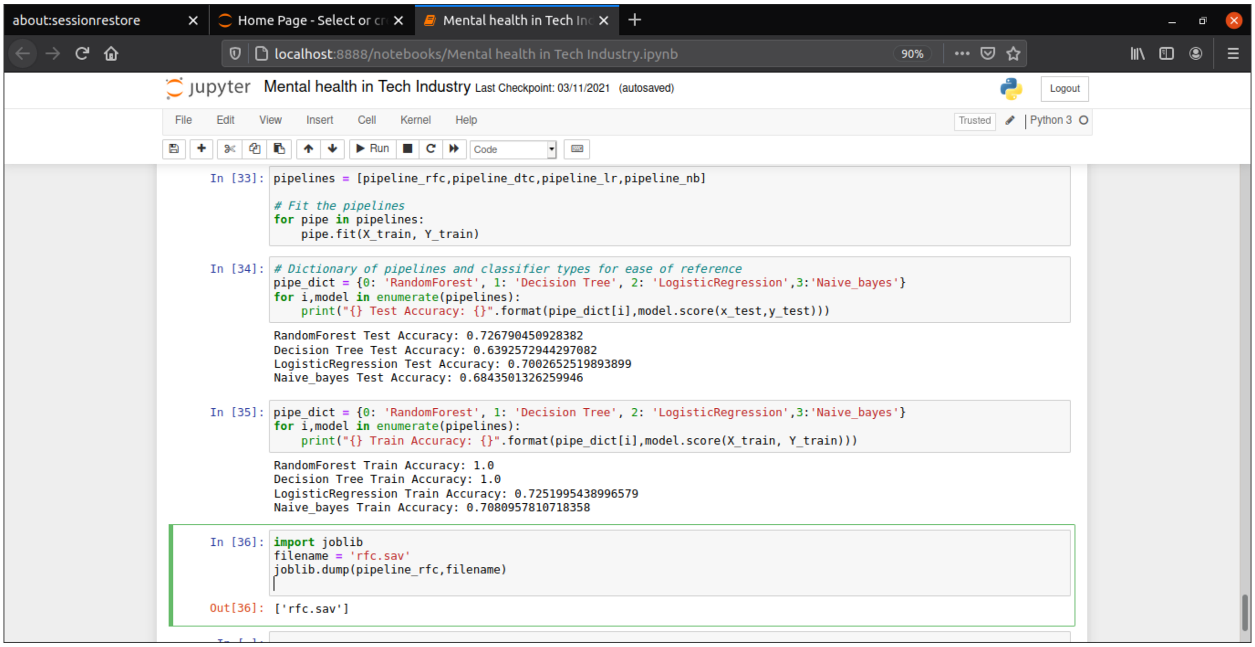
Out[5]:
```
{'Gender': 49,
 'self_employed': 3,
 'family_history': 2,
 'treatment': 2,
 'work_interfere': 5,
 'no_employees': 6,
 'remote_work': 2,
 'tech_company': 2,
 'benefits': 3,
 'care_options': 3,
 'wellness_program': 3,
 'seek_help': 3,
 'anonymity': 3,
 'leave': 5,
 'mental_health_consequence': 3,
 'phys_health_consequence': 3,
 'coworkers': 3,
 'supervisor': 3,
 'mental_health_interview': 3,
 'phys_health_interview': 3,
 'mental_vs_physical': 3,
 'obs_consequence': 2}
```


about:sessionrestore
Home Page - Select or cr
Mental health in Tech Inc
localhost:8888/notebooks/Mental health in Tech Industry.ipynb
90%
jupyter
Mental health in Tech Industry Last Checkpoint: 03/11/2021 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Help
Trusted
Python 3
Run
Code

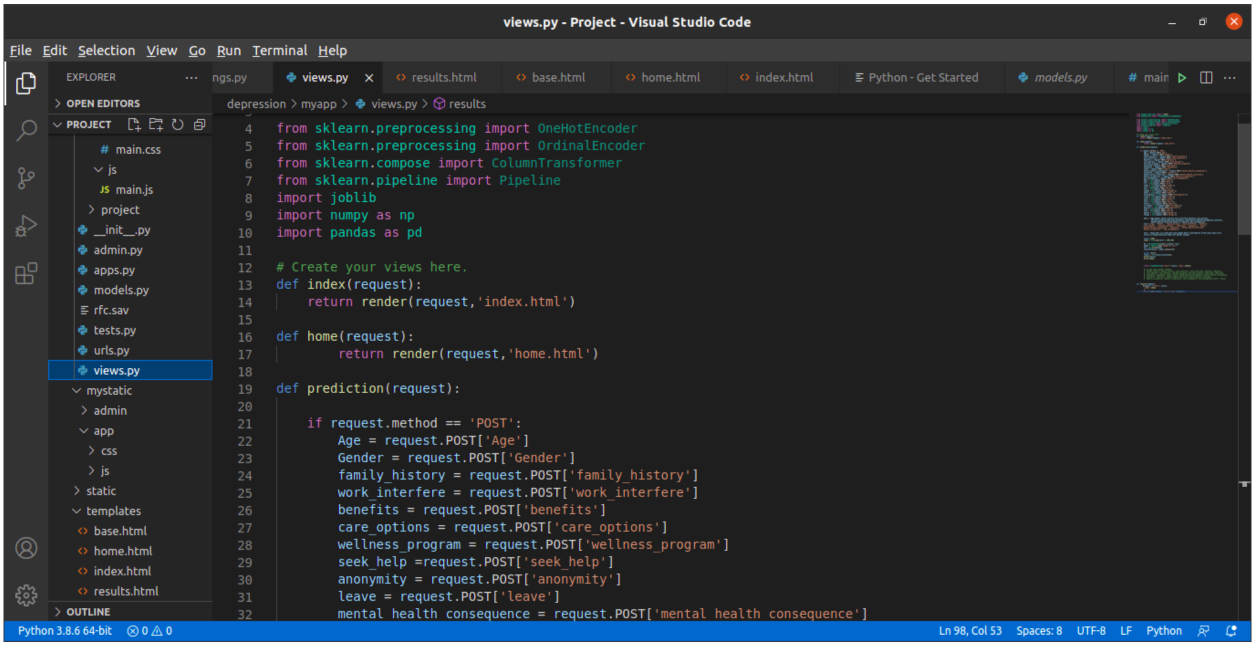
```
In [6]: {column: data[column].unique() for column in data.select_dtypes('object').columns}

Out[6]: {'Gender': array(['Female', 'M', 'Male', 'male', 'female', 'm', 'Male-ish', 'maile',
                'Trans-female', 'Cis Female', 'F', 'something kinda male?',
                'Cis Male', 'Woman', 'f', 'Mal', 'Male (CIS)', 'queer/she/they',
                'non-binary', 'Femake', 'woman', 'Make', 'Nah', 'All', 'Enby',
                'fluid', 'Genderqueer', 'Female ', 'Androgyne', 'Agender',
                'cis-female/femme', 'Guy (-ish) ^_^', 'male leaning androgynous',
                'Male ', 'Man', 'Trans woman', 'msle', 'Neuter', 'Female (trans)',
                'queer', 'Female (cis)', 'Mail', 'cis male', 'A little about you',
                'Malr', 'p', 'femail', 'Cis Man',
                'ostensibly male, unsure what that really means'], dtype=object),
         'self_employed': array([nan, 'Yes', 'No'], dtype=object),
         'family_history': array(['No', 'Yes'], dtype=object),
         'treatment': array(['Yes', 'No'], dtype=object),
         'work_interfere': array(['Often', 'Rarely', 'Never', 'Sometimes', nan], dtype=object),
         'no_employees': array(['6-25', 'More than 1000', '26-100', '100-500', '1-5', '500-1000'],
               dtype=object),
         'remote_work': array(['No', 'Yes'], dtype=object),
         'tech_company': array(['Yes', 'No'], dtype=object),
         'benefits': array(['Yes', "Don't know", 'No'], dtype=object),
         'care_options': array(['Not sure', 'No', 'Yes'], dtype=object),
         'wellness_program': array(['No', "Don't know", 'Yes'], dtype=object),
         'seek_help': array(['Yes', "Don't know", 'No'], dtype=object),
         'anonymity': array(['Yes', "Don't know", 'No'], dtype=object),
         'leave': array(['Somewhat easy', "Don't know", 'Somewhat difficult',
                'Very difficult', 'Very easy'], dtype=object),
         'mental_health_consequence': array(['No', 'Maybe', 'Yes'], dtype=object),
         'phys_health_consequence': array(['No', 'Yes', 'Maybe'], dtype=object),
         'coworkers': array(['Some of them', 'No', 'Yes'], dtype=object),
         'supervisor': array(['Yes', 'No', 'Some of them'], dtype=object),
         'mental_health_interview': array(['No', 'Yes', 'Maybe'], dtype=object),
         'phys_health_interview': array(['Maybe', 'No', 'Yes'], dtype=object),
```


about:sessionrestore
Home Page - Select or cr
Mental health in Tech Inc
localhost:8888/notebooks/Mental health in Tech Industry.ipynb
90%
jupyter
Mental health in Tech Industry Last Checkpoint: 03/11/2021 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Help
Trusted
Python 3
Run
Code

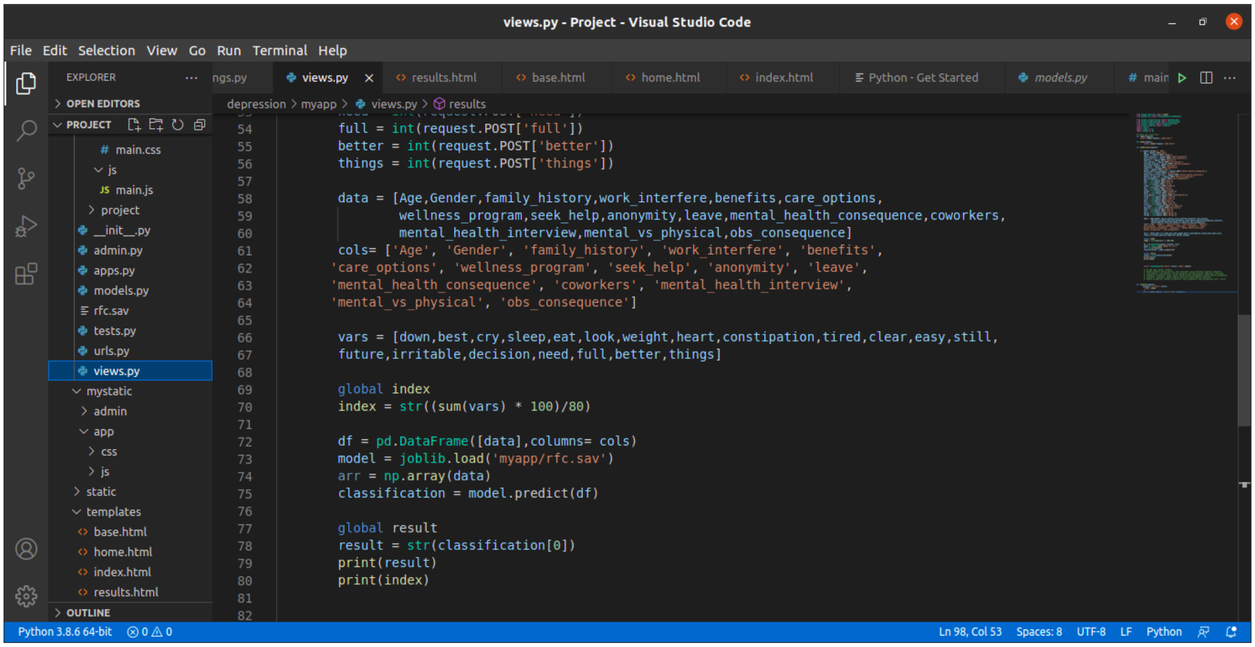
```
In [28]: from sklearn.preprocessing import OneHotEncoder
         from sklearn.preprocessing import OrdinalEncoder
         from sklearn.model_selection import train_test_split
         from sklearn.compose import ColumnTransformer
         from sklearn.pipeline import Pipeline
         from sklearn.ensemble import RandomForestClassifier
         from sklearn.tree import DecisionTreeClassifier
         from sklearn.linear_model import LogisticRegression
         from sklearn.naive_bayes import GaussianNB

In [29]: X = data.drop(columns = 'treatment')
         y = data['treatment']
         X_train,x_test,Y_train,y_test = train_test_split(X,y,test_size = 0.3, random_state = 10)

In [30]: ct = ColumnTransformer([('enc',OrdinalEncoder(),ordinal_features),
                                      ('ohe',OneHotEncoder(),nominal_features)],remainder = 'passthrough')

In [31]: X_train.columns

Out[31]: Index(['Age', 'Gender', 'family_history', 'work_interfere', 'benefits',
                'care_options', 'wellness_program', 'seek_help', 'anonymity', 'leave',
                'mental_health_consequence', 'coworkers', 'mental_health_interview',
                'mental_vs_physical', 'obs_consequence'],
               dtype='object')

In [32]: pipeline_rfc = Pipeline([('transformer',ct),('rfc',RandomForestClassifier())])
         pipeline_dtc = Pipeline([('transformer',ct),('dtc',DecisionTreeClassifier())])
         pipeline_lr = Pipeline([('transformer',ct),('lr',LogisticRegression())])
         pipeline_nb = Pipeline([('transformer',ct),('lr',GaussianNB())])

In [33]: pipelines = [pipeline_rfc,pipeline_dtc,pipeline_lr,pipeline_nb]
```


about:sessionrestore
Home Page - Select or cr
Mental health in Tech Inc
localhost:8888/notebooks/Mental health in Tech Industry.ipynb
90%
jupyter Mental health in Tech Industry Last Checkpoint: 03/11/2021 (autosaved)
Logout
File
Edit
View
Insert
Cell
Kernel
Help
Trusted
Python 3
Run
Code

```
In [33]: pipelines = [pipeline_rfc,pipeline_dtc,pipeline_lr,pipeline_nb]

         # Fit the pipelines
         for pipe in pipelines:
             pipe.fit(X_train, Y_train)

In [34]: # Dictionary of pipelines and classifier types for ease of reference
         pipe_dict = {0: 'RandomForest', 1: 'Decision Tree', 2: 'LogisticRegression',3:'Naive_bayes'}
         for i,model in enumerate(pipelines):
             print("{} Test Accuracy: {}".format(pipe_dict[i],model.score(x_test,y_test)))

         RandomForest Test Accuracy: 0.726790450928382
         Decision Tree Test Accuracy: 0.6392572944297082
         LogisticRegression Test Accuracy: 0.7002652519893899
         Naive_bayes Test Accuracy: 0.6843501326259946

In [35]: pipe_dict = {0: 'RandomForest', 1: 'Decision Tree', 2: 'LogisticRegression',3:'Naive_bayes'}
         for i,model in enumerate(pipelines):
             print("{} Train Accuracy: {}".format(pipe_dict[i],model.score(X_train, Y_train)))

         RandomForest Train Accuracy: 1.0
         Decision Tree Train Accuracy: 1.0
         LogisticRegression Train Accuracy: 0.7251995438996579
         Naive_bayes Train Accuracy: 0.7080957810718358

In [36]: import joblib
         filename = 'rfc.sav'
         joblib.dump(pipeline_rfc,filename)

Out[36]: ['rfc.sav']
```



```
from sklearn.preprocessing import OneHotEncoder
from sklearn.preprocessing import OrdinalEncoder
from sklearn.compose import ColumnTransformer
from sklearn.pipeline import Pipeline
import joblib
import numpy as np
import pandas as pd

# Create your views here.
def index(request):
    return render(request,'index.html')

def home(request):
        return render(request,'home.html')

def prediction(request):

    if request.method == 'POST':
        Age = request.POST['Age']
        Gender = request.POST['Gender']
        family_history = request.POST['family_history']
        work_interfere = request.POST['work_interfere']
        benefits = request.POST['benefits']
        care_options = request.POST['care_options']
        wellness_program = request.POST['wellness_program']
        seek_help =request.POST['seek_help']
        anonymity = request.POST['anonymity']
        leave = request.POST['leave']
        mental_health_consequence = request.POST['mental_health_consequence']
```



```
        need = int(request.POST['need'])
        full = int(request.POST['full'])
        better = int(request.POST['better'])
        things = int(request.POST['things'])

        data = [Age,Gender,family_history,work_interfere,benefits,care_options,
                wellness_program,seek_help,anonymity,leave,mental_health_consequence,coworkers,
                mental_health_interview,mental_vs_physical,obs_consequence]
        cols= ['Age', 'Gender', 'family_history', 'work_interfere', 'benefits',
       'care_options', 'wellness_program', 'seek_help', 'anonymity', 'leave',
       'mental_health_consequence', 'coworkers', 'mental_health_interview',
       'mental_vs_physical', 'obs_consequence']

        vars = [down,best,cry,sleep,eat,look,weight,heart,constipation,tired,clear,easy,still,
        future,irritable,decision,need,full,better,things]

        global index
        index = str((sum(vars) * 100)/80)

        df = pd.DataFrame([data],columns= cols)
        model = joblib.load('myapp/rfc.sav')
        arr = np.array(data)
        classification = model.predict(df)

        global result
        result = str(classification[0])
        print(result)
        print(index)
```