PYTHON EXPERIMENT 9

Name : Gini Chacko

Roll : 8942

Class : SE Comps B

Aim: Write a Menu driven pandas program to:
1. Merging dataframes
2. Joining dataframes
3. Concatenation dataframes.

CODE:

```
# importing pandas module
import pandas as pd

while True:
    print("\n\n******Menu driven pandas program******")
    print("1. Merging dataframes")
    print("2. Joining dataframes")
    print("3. Concatenation dataframes")
    print("4. Exit")
    choice = int(input("\nEnter your Choice : "))
    if choice==1:
        # Define a dictionary containing employee data
        print("----------------------------------------")
        print("\nDataset 1 : \n")
        data1 = {'key': ['K0', 'K1', 'K2', 'K3'],
                'Name':['Jai', 'Princi', 'Gaurav', 'Anuj'],
                'Age':[27, 24, 22, 32],}

        # Define a dictionary containing employee data
        data2 = {'key': ['K0', 'K1', 'K2', 'K3'],
```

```python
        'Address':['Nagpur', 'Kanpur', 'Allahabad', 'Kannuaj'],
        'Qualification':['Btech', 'B.A', 'Bcom', 'B.hons']}
  # Convert the dictionary into DataFrame
  df = pd.DataFrame(data1)

  # Convert the dictionary into DataFrame
  df1 = pd.DataFrame(data2)

  print(df, "\n\n Dataset 2 : \n\n", df1)
  print("--------------------------------------")
  # using .merge() function
  res = pd.merge(df, df1, on='key')
  print("\nDataframe after Merging : \n")
  print(res)
  print("--------------------------------------")

elif choice==2:
  # Define a dictionary containing employee data
  print("--------------------------------------")
  print("\nDataset 1 : \n")
  data1 = {'Name':['Jai', 'Princi', 'Gaurav', 'Anuj'],
       'Age':[27, 24, 22, 32]}

  # Define a dictionary containing employee data
  data2 = {'Address':['Allahabad', 'Kannuaj', 'Allahabad', 'Kannuaj'],
       'Qualification':['MCA', 'Phd', 'Bcom', 'B.hons']}
  # Convert the dictionary into DataFrame
  df = pd.DataFrame(data1,index=['K0', 'K1', 'K2', 'K3'])

  # Convert the dictionary into DataFrame
  df1 = pd.DataFrame(data2, index=['K0', 'K2', 'K3', 'K4'])


  print(df, "\n\nDataset 2 : \n\n", df1)
  print("--------------------------------------")
```

GINI CHACKO 8942

```python
        # joining dataframe
        res = df.join(df1)
        print("\nDataframe after Joining : \n")
        print(res)
        print("----------------------------------------")

    elif choice==3:
        # Define a dictionary containing employee data
        print("----------------------------------------")
        print("\nDataset 1 : \n")
        data1 = {'Name':['Jai', 'Princi', 'Gaurav', 'Anuj'],
                'Age':[27, 24, 22, 32],
                'Address':['Nagpur', 'Kanpur', 'Allahabad', 'Kannuaj'],
                'Qualification':['Msc', 'MA', 'MCA', 'Phd']}

        # Define a dictionary containing employee data
        data2 = {'Name':['Abhi', 'Ayushi', 'Dhiraj', 'Hitesh'],
                'Age':[17, 14, 12, 52],
                'Address':['Nagpur', 'Kanpur', 'Allahabad', 'Kannuaj'],
                'Qualification':['Btech', 'B.A', 'Bcom', 'B.hons']}
        # Convert the dictionary into DataFrame
        df = pd.DataFrame(data1,index=[0, 1, 2, 3])

        # Convert the dictionary into DataFrame
        df1 = pd.DataFrame(data2, index=[2, 3, 6, 7])

        print(df, "\n\n Dataset 2 : \n\n", df1)
        print("----------------------------------------")
        # using a .concat() method
        frames = [df, df1]

        res1 = pd.concat(frames)
        print("\nDataframe after Concatinating : \n")
        print(res1)
        print("----------------------------------------")
```

```
    elif choice==4:
        print("Exiting!!")
        print("\n---------------------------------------")
        break

    else:
        print("Wrong Choice!")
        print("\n---------------------------------------")
```

OUTPUT:

```
PS C:\Users\Chacko> & python "c:/GINI/ENGG/2nd Year/Sem 4/Python/LABS/EXP 9/exp_9.py"


******Menu driven pandas program******
1. Merging dataframes
2. Joining dataframes
3. Concatenation dataframes
4. Exit

Enter your Choice : 1
---------------------------------------

Dataset 1 :

  key    Name  Age
0  K0     Jai   27
1  K1  Princi   24
2  K2  Gaurav   22
3  K3    Anuj   32

 Dataset 2 :

   key    Address Qualification
0  K0     Nagpur         Btech
1  K1     Kanpur           B.A
2  K2  Allahabad          Bcom
3  K3    Kannuaj        B.hons
---------------------------------------
```

GINI CHACKO 8942

```
Dataframe after Merging :

  key    Name  Age    Address Qualification
0  K0     Jai   27     Nagpur         Btech
1  K1  Princi   24     Kanpur           B.A
2  K2  Gaurav   22  Allahabad          Bcom
3  K3    Anuj   32    Kannuaj        B.hons
----------------------------------------


******Menu driven pandas program******
1. Merging dataframes
2. Joining dataframes
3. Concatenation dataframes
4. Exit

Enter your Choice : 2
----------------------------------------

Dataset 1 :

      Name  Age
K0     Jai   27
K1  Princi   24
K2  Gaurav   22
K3    Anuj   32

Dataset 2 :

      Address Qualification
K0  Allahabad           MCA
K2    Kannuaj           Phd
K3  Allahabad          Bcom
K4    Kannuaj        B.hons
----------------------------------------

Dataframe after Joining :

      Name  Age    Address Qualification
K0     Jai   27  Allahabad           MCA
K1  Princi   24        NaN           NaN
K2  Gaurav   22    Kannuaj           Phd
K3    Anuj   32  Allahabad          Bcom
----------------------------------------
```

```
******Menu driven pandas program******
1. Merging dataframes
2. Joining dataframes
3. Concatenation dataframes
4. Exit

Enter your Choice : 3
-----------------------------------------


Dataset 1 :

      Name  Age     Address Qualification
0     Jai   27      Nagpur           Msc
1   Princi  24      Kanpur            MA
2   Gaurav  22   Allahabad           MCA
3     Anuj  32     Kannuaj           Phd

 Dataset 2 :

      Name  Age     Address Qualification
2    Abhi   17      Nagpur         Btech
3   Ayushi  14      Kanpur           B.A
6   Dhiraj  12   Allahabad          Bcom
7   Hitesh  52     Kannuaj        B.hons
-----------------------------------------


Dataframe after Concatinating :

      Name  Age     Address Qualification
0     Jai   27      Nagpur           Msc
1   Princi  24      Kanpur            MA
2   Gaurav  22   Allahabad           MCA
3     Anuj  32     Kannuaj           Phd
2    Abhi   17      Nagpur         Btech
3   Ayushi  14      Kanpur           B.A
6   Dhiraj  12   Allahabad          Bcom
7   Hitesh  52     Kannuaj        B.hons
-----------------------------------------
```

GINI CHACKO 8942

```
******Menu driven pandas program******
1. Merging dataframes
2. Joining dataframes
3. Concatenation dataframes
4. Exit

Enter your Choice : 6
Wrong Choice!


----------------------------------------



******Menu driven pandas program******
1. Merging dataframes
2. Joining dataframes
3. Concatenation dataframes
4. Exit

Enter your Choice : 4
Exiting!!


----------------------------------------
```

POSTLAB QUESTIONS

1. **Mention the methods used for checking missing values, filling missing values, dropping missing values with example.**

Ans:

Checking for missing values
1.] using isnull() and notnull()
In order to check missing values in Pandas DataFrame, we use a
function isnull() and notnull(). Both function help in checking whether a value
is NaN or not. These function can also be used in Pandas Series in order to find
null values in a series.

A.] using isnull()
In order to check null values in Pandas DataFrame, we use isnull() function this
function return dataframe of Boolean values which are True for NaN values.

\# importing pandas as pd
import pandas as pd

```
# importing numpy as np
import numpy as np

# dictionary of lists
dict = {'First Score':[100, 90, np.nan, 95],
     'Second Score': [30, 45, 56, np.nan],
     'Third Score':[np.nan, 40, 80, 98]}

# creating a dataframe from list
df = pd.DataFrame(dict)

# using isnull() function
df.isnull()
```

B.] using notnull()
In order to check null values in Pandas Dataframe, we use notnull() function this
function return dataframe of Boolean values which are False for NaN values.

```
# importing pandas as pd
import pandas as pd

# importing numpy as np
import numpy as np

# dictionary of lists
dict = {'First Score':[100, 90, np.nan, 95],
     'Second Score': [30, 45, 56, np.nan],
     'Third Score':[np.nan, 40, 80, 98]}

# creating a dataframe using dictionary
df = pd.DataFrame(dict)

# using notnull() function
df.notnull()
```

<u>Filling missing values</u>
1.] using fillna(),
2.] using replace(),
3.] using interpolate()

GINI CHACKO 8942

In order to fill null values in a datasets, we
use fillna(), replace() and interpolate() function these function replace NaN
values with some value of their own. All these function help in filling a null
values in datasets of a DataFrame. Interpolate() function is basically used to
fill NA values in the dataframe but it uses various interpolation technique to fill
the missing values rather than hard-coding the value.

A.] Example using fillna()
```
# importing pandas as pd
import pandas as pd

# importing numpy as np
import numpy as np

# dictionary of lists
dict = {'First Score':[100, 90, np.nan, 95],
     'Second Score': [30, 45, 56, np.nan],
     'Third Score':[np.nan, 40, 80, 98]}

# creating a dataframe from dictionary
df = pd.DataFrame(dict)

# filling missing value using fillna()
df.fillna(0)
```

B.] Example using replace()
```
# importing pandas package
import pandas as pd

# making data frame from csv file
data = pd.read_csv("employees.csv")
# will replace  Nan value in dataframe with value -99
data.replace(to_replace = np.nan, value = -99)
```

C.] Example using interpolate()
```
# importing pandas as pd
```

```python
import pandas as pd

# Creating the dataframe
df = pd.DataFrame({"A":[12, 4, 5, None, 1],
            "B":[None, 2, 54, 3, None],
            "C":[20, 16, None, 3, 8],
            "D":[14, 3, None, None, 6]})

# Print the dataframe
df

# to interpolate the missing values
df.interpolate(method ='linear', limit_direction ='forward')
```

Dropping missing values
1.] using dropna()
In order to drop a null values from a dataframe, we used dropna() function this function drop Rows/Columns of datasets with Null values in different ways.

```python
# importing pandas as pd
import pandas as pd

# importing numpy as np
import numpy as np

# dictionary of lists
dict = {'First Score':[100, 90, np.nan, 95],
     'Second Score': [30, np.nan, 45, 56],
     'Third Score':[52, 40, 80, 98],
     'Fourth Score':[np.nan, np.nan, np.nan, 65]}

# creating a dataframe from dictionary
df = pd.DataFrame(dict)

# using dropna() function
df.dropna()
```