**DATA WRANGLING**

**Ex.No. :5 230901039**

**Date :07-03-2025**

**AIM:**

To do data wrangling functions.

**SOFTWARE USED:**

Jupyter notebook

**DATA WRANGLING-1**

**DESCRIPTION:**

1. Using join function to join dataframes
2. Using combine function to combine two dataframes.
3. Using merge function to merge two dataframes.
4. Using melt function to reshape dataframe dimensions.

**PROGRAM:**

**import** pandas **as** pd

d1**=**{"name":["Sai","Sub",'Poo'],"age":[20,21,22]}

d2**=**{"qualified":[**True**,**True**,**False**]}

df1**=**pd**.**DataFrame(d1)

print(df1)

df2**=**pd**.**DataFrame(d2)

print(df2)

**import** pandas **as** pd

d1**=**{"name":["Sai","Rishi",'Thara'],"age":[20,21,22]}

d2**=**{"name":["Yuvi","Rithi","Sanjay"],"age":[20,21,22]}

df1**=**pd**.**DataFrame(d1)

print(df1)

df2**=**pd**.**DataFrame(d2)

print(df2)

newdf1**=**(df2**.**merge(df1,on**=**"age"))

newdf2**=**(df1**.**merge(df2,on**=**"age"))

print(newdf1)

print(newdf2)

df**=**pd**.**DataFrame({'team':['A','B','C','D'],

'points':[90,56,78,88],

'assists':[12,34,23,44],

'rebounds':[56,43,55,12]})

print(df)

df**=**pd**.**melt(df,id\_vars**=**'team',value\_vars**=**['points','assists','rebounds'])

print("RESHAPED DATA FRAME")

print(df)

df1**=**pd**.**DataFrame([[11,12],[3,4]])

df2**=**pd**.**DataFrame([[5,6],[7,8]])

**def** myfunc(a,b):

**if**(a**.**sum()**>**b**.**sum()):

**return** a

**else**:

**return** b

print(df1)

print(df2)

print(df1**.**combine(df2,myfunc))

d1**=**{

"name":["sai","sub","poo"],

"age":[20,21,22]

}

d2**=**{

"name":["yuvi","rithika","sanjay"],

"age":[15,16,17]

}

df1**=**pd**.**DataFrame(d1)

print("DF1:\n",df1)

df2**=**pd**.**DataFrame(d2)

print("DF2:\n",df2)

newdf1**=**df1**.**merge(df2, how**=**'left')

newdf2**=**df1**.**merge(df2, how**=**'right')

print("Left Merge:\n",newdf1)

print("Rgiht Merge:\n",newdf2)

**OUTPUT:**

name age

0 Sai 20

1 Sub 21

2 Poo 22

qualified

0 True

1 True

2 False

name age

0 Sai 20

1 Rishi 21

2 Thara 22

name age

0 Yuvi 20

1 Rithi 21

2 Sanjay 22

name\_x age name\_y

0 Yuvi 20 Sai

1 Rithi 21 Rishi

2 Sanjay 22 Thara

name\_x age name\_y

0 Sai 20 Yuvi

1 Rishi 21 Rithi

2 Thara 22 Sanjay

team points assists rebounds

0 A 90 12 56

1 B 56 34 43

2 C 78 23 55

3 D 88 44 12

RESHAPED DATA FRAME

team variable value

0 A points 90

1 B points 56

2 C points 78

3 D points 88

4 A assists 12

5 B assists 34

6 C assists 23

7 D assists 44

8 A rebounds 56

9 B rebounds 43

10 C rebounds 55

11 D rebounds 12

0 1

0 11 12

1 3 4

0 1

0 5 6

1 7 8

0 1

0 11 12

1 3 4

DF1:

name age

0 sai 20

1 sub 21

2 poo 22

DF2:

name age

0 yuvi 15

1 rithika 16

2 sanjay 17

Left Merge:

name age

0 sai 20

1 sub 21

2 poo 22

Rgiht Merge:

name age

0 yuvi 15

1 rithika 16

2 sanjay 17

**DATA WRANGLING-2**

**DESCRIPTION:**

1. Using replace functions to replace NaN values by average value.
2. Filtering and dropping the rows and columns respectively.
3. Using concat function to concatenate two dataframes.

**PROGRAM:**

**import** pandas **as** pd

*#Assign data*

data**=**{'Name':['Sai','Yuvasri','Poovannan',

'Sabharish','Sangamithra','Rithika','Rubasri'],

'Age':[20,18,19,20,18,17,19],

'Gender':['M','F','M','M','F','F','F'],

'Marks':[90,76,'NaN',74,65,'NaN',71]}

*#convert into datafram*

df**=**pd**.**DataFrame(data)

*#display data*

print("Original DataFrame:\n",df)

*#convert average*

c**=**avg**=**0

**for** ele **in** df['Marks']:

**if** str(ele)**.**isnumeric():

c**+=**1

avg**+=**ele

avg**/=**c

*#Replace missing values*

df**=**df**.**replace(to\_replace**=**"NaN",value**=**avg)

*#Display data*

print("\nReplacing NaN with average marks:\n",df)

df**=**df[df['Marks']**>=**75]

*#Remove age row*

df**=**df**.**drop(['Age'],axis**=**1)

*#Display data*

print("\nFiltering and Dropping func:\n",df)

one **=** pd**.**DataFrame({

'Name':['Sai','Sub','Poo','Yuvasri','Rithika'],

'subject\_id':['sub1','sub2','sub4','sub6','sub5'],

'Marks\_scored':[98,95,96,92,88]},

index**=**[1,2,3,4,5])

two**=**pd**.**DataFrame({

'Name':['Sanjay','Sangamithra','Rubasri','Nikesh','Somanath'],

'subject\_id':['sub2','sub4','sub3','sub6','sub5'],

'Marks\_scored':[86,95,87,99,91]},

index**=**[1,2,3,4,5])

print("\nOriginal DataFrame 1:\n",one)

print("\nOriginal DataFrame 2:\n",two)

print(pd**.**concat([one,two]))

**OUTPUT:**

Original DataFrame:

Name Age Gender Marks

0 Sai 20 M 90

1 Yuvasri 18 F 76

2 Poovannan 19 M NaN

3 Sabharish 20 M 74

4 Sangamithra 18 F 65

5 Rithika 17 F NaN

6 Rubasri 19 F 71

Replacing NaN with average marks:

Name Age Gender Marks

0 Sai 20 M 90.000000

1 Yuvasri 18 F 76.000000

2 Poovannan 19 M 20.066667

3 Sabharish 20 M 74.000000

4 Sangamithra 18 F 65.000000

5 Rithika 17 F 20.066667

6 Rubasri 19 F 71.000000

Filtering and Dropping func:

Name Gender Marks

0 Sai M 90.0

1 Yuvasri F 76.0

Original DataFrame 1:

Name subject\_id Marks\_scored

1 Sai sub1 98

2 Sub sub2 95

3 Poo sub4 96

4 Yuvasri sub6 92

5 Rithika sub5 88

Original DataFrame 2:

Name subject\_id Marks\_scored

1 Sanjay sub2 86

2 Sangamithra sub4 95

3 Rubasri sub3 87

4 Nikesh sub6 99

5 Somanath sub5 91

Name subject\_id Marks\_scored

1 Sai sub1 98

2 Sub sub2 95

3 Poo sub4 96

4 Yuvasri sub6 92

5 Rithika sub5 88

1 Sanjay sub2 86

2 Sangamithra sub4 95

3 Rubasri sub3 87

4 Nikesh sub6 99

5 Somanath sub5 91

**DATA WRANGLING-3**

**DESCRIPTION:**

1. Using groupby() function to group the dataset.
2. Using duplicated function to remove duplicated rows in the dataframe.
3. Using merge() function to merge two dataframe.

**PROGRAM:**

**import** pandas **as** pd

car\_selling\_data **=**{'Brand':['Maruti','Maruti','Maruti','Maruti','Hyundai','Hyundai','Toyota','Mahindra','Mahindra','Ford',

'Toyota','Ford'],

'Year':[2010, 2011, 2009, 2013, 2010, 2011, 2011, 2010, 2013, 2010, 2010, 2011],

'Sold':[6, 7, 9, 8, 3, 5,2, 8, 7, 2, 4, 2]}

*#Creating DataFrame for provided Data*

df **=** pd**.** DataFrame(car\_selling\_data)

print("\nOriginal DataFrame:\n",df)

*#Group the data when year= 2020*

grouped **=** df**.**groupby('Year')

print("\nGroup by year 2010:\n",grouped**.**get\_group(2010))

*#initializing Data*

student\_data **=** {'Name': ['Amit', 'Praveen', 'Jagadesh','Rahul', 'Vishal', 'Suraj','Rishab', 'Sathish',

'Amit','Rahul', 'Praveen', 'Amit'],

'Roll\_no': [23,54,29,36,23,38,12,45,34,36,54,23],

'Email': ['xxxx@gmail.com', 'xxxxxx@gmail.com','xxxxxx@gmail.com','xx@gmail.com',

'xxxx@gmail.com', 'xxxxx@gmail.com','xxxxx@gmail.com', 'xxxxx@gmail.com',

'xxxxx@gmail.com', 'xxxxxx@gmail.com','xxxxxxxxxx@gmail.com',

'xxxxxxxxxx@gmail.com']}

*#Creating dataframe*

df**=**pd**.**DataFrame(student\_data)

print("\nOriginal DataFrame:\n",df)

*#Here df.duplicated() list duplicate entrines in rollno.*

*#so that (NOT) is placed in order to get non duplicate values.*

non\_duplicate **=** df[**~**df**.**duplicated('Roll\_no')]

*#printing non duplicate values.*

print("\nRemoved duplicated rows:\n",non\_duplicate)

*#creating datafram*

details**=**pd**.**DataFrame({'ID':[101, 102, 103, 104, 105, 106, 107, 108, 109, 110],

'NAME':['Arun', 'Praveen', 'Harish','Pooja', 'Rahul', 'Naresh','Saurabh','Anush','Dinesh',

'Mohit'],

'BRANCH':['Mech', 'Mech', 'CSE', 'CSE', 'CSE', 'EEE','EEE', 'ECE', 'ECE', 'IT']})

print("\nOriginal DataFrame 1:\n", details)

*#creating datafram*

fees\_status**=**pd**.**DataFrame ({'ID': [101, 102, 103, 104, 105, 106, 107, 108, 109, 110],

'PENDING': ['5000', '250', 'NIL', '9000', '15000', 'NIL', '4500', '1800', '250', 'NIL']})

print("\nOriginal DataFrame:\n", fees\_status)

*#mergind dataframe*

print("\nMerged Dataframe\n",pd**.**merge(details,fees\_status,on**=**'ID'))

**OUTPUT:**

Original DataFrame:

Brand Year Sold

0 Maruti 2010 6

1 Maruti 2011 7

2 Maruti 2009 9

3 Maruti 2013 8

4 Hyundai 2010 3

5 Hyundai 2011 5

6 Toyota 2011 2

7 Mahindra 2010 8

8 Mahindra 2013 7

9 Ford 2010 2

10 Toyota 2010 4

11 Ford 2011 2

Group by year 2010:

Brand Year Sold

0 Maruti 2010 6

4 Hyundai 2010 3

7 Mahindra 2010 8

9 Ford 2010 2

10 Toyota 2010 4

Original DataFrame:

Name Roll\_no Email

0 Amit 23 xxxx@gmail.com

1 Praveen 54 xxxxxx@gmail.com

2 Jagadesh 29 xxxxxx@gmail.com

3 Rahul 36 xx@gmail.com

4 Vishal 23 xxxx@gmail.com

5 Suraj 38 xxxxx@gmail.com

6 Rishab 12 xxxxx@gmail.com

7 Sathish 45 xxxxx@gmail.com

8 Amit 34 xxxxx@gmail.com

9 Rahul 36 xxxxxx@gmail.com

10 Praveen 54 xxxxxxxxxx@gmail.com

11 Amit 23 xxxxxxxxxx@gmail.com

Removed duplicated rows:

Name Roll\_no Email

0 Amit 23 xxxx@gmail.com

1 Praveen 54 xxxxxx@gmail.com

2 Jagadesh 29 xxxxxx@gmail.com

3 Rahul 36 xx@gmail.com

5 Suraj 38 xxxxx@gmail.com

6 Rishab 12 xxxxx@gmail.com

7 Sathish 45 xxxxx@gmail.com

8 Amit 34 xxxxx@gmail.com

Original DataFrame 1:

ID NAME BRANCH

0 101 Arun Mech

1 102 Praveen Mech

2 103 Harish CSE

3 104 Pooja CSE

4 105 Rahul CSE

5 106 Naresh EEE

6 107 Saurabh EEE

7 108 Anush ECE

8 109 Dinesh ECE

9 110 Mohit IT

Original DataFrame:

ID PENDING

0 101 5000

1 102 250

2 103 NIL

3 104 9000

4 105 15000

5 106 NIL

6 107 4500

7 108 1800

8 109 250

9 110 NIL

Merged Dataframe

ID NAME BRANCH PENDING

0 101 Arun Mech 5000

1 102 Praveen Mech 250

2 103 Harish CSE NIL

3 104 Pooja CSE 9000

4 105 Rahul CSE 15000

5 106 Naresh EEE NIL

6 107 Saurabh EEE 4500

7 108 Anush ECE 1800

8 109 Dinesh ECE 250

9 110 Mohit IT NIL

**Result:**

Thus data wrangling is done using various functions.