

## Experiment 16

### Aim:

To develop a Pandas program to split the dataset into groups based on school code. Also check the type of GroupBy object

**Code:**

```
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Codes Version control Current File
groupby.py x
1 import pandas as pd
2
3 df = pd.read_csv('school.csv')
4
5 grouped = df.groupby('school_code')
6
7 print(type(grouped))
8
9 print(grouped.groups)
10
11 for name, group in grouped:
12     print(f"Group name: {name}")
13     print(group)
14
```

**Input:**

[illegible]

# Output:

Run groupby x

↶

↷

↺

↻

⌵

🗑


"C:\Users\maano\_0waenfu\OneDrive\College\Query Processing\Codes\.venv\Scripts\python.exe" "C:\Us  
<class 'pandas.core.groupby.generic.DataFrameGroupBy'>  
{'s001': [0, 3], 's002': [1, 4], 's003': [2], 's004': [5]}  
Group name: s001  
school\_code class name date\_of\_birth age height weight address  
0 s001 V Alberto Franco 15-05-2002 12 173 35 street1  
3 s001 VI Eesha Hinton 25-09-1998 13 167 30 street1  
Group name: s002  
school\_code class name date\_of\_birth age height weight address  
1 s002 V Gino Mcneill 17-05-2002 12 192 32 stree2  
4 s002 V Gino Mcneill 11-05-2002 14 151 31 street2  
Group name: s003  
school\_code class name date\_of\_birth age height weight address  
2 s003 VI Ryan Parkes 16-02-1999 13 186 33 street3  
Group name: s004  
school\_code class name date\_of\_birth age height weight address  
5 s004 V David Parkes 15-09-1997 12 159 32 street4  
  
Process finished with exit code 0

## Experiment 17

### Aim:

To develop a Pandas program to split the dataset into groups based on school code and get mean, min, and max value of age for each school

## Code:



The screenshot shows a code editor with a dark theme. The top menu bar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. Below the menu bar, there are tabs for 'Codes' and 'Version control'. The main editor area displays a Python script in a file named 'mean\_meadiana\_mode.py'. The script uses pandas to read a CSV file, group data by 'school\_code', and calculate the mean, minimum, and maximum values for the 'age' column. The script is as follows:

```
1 import pandas as pd
2
3 df = pd.read_csv('school.csv')
4
5 grouped = df.groupby('school_code')['age'].agg(['mean', 'min', 'max'])
6
7 print(grouped)
```

The bottom status bar shows the current file is 'mean\_meadiana\_mode.py', the time is 7:15, and the encoding is UTF-8.

**Input:**

school - Excel

File Home Insert Draw Page Layout Formulas Data Review View Help Acrobat Tell me what you want to do

Paste Clipboard Font Alignment Number Conditional Formatting Styles Cell Styles Cells Editing Add-ins Create and Share Adobe PDF

P21

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
	school_code	class	name	date_of_bir	age	height	weight	address										
1	s001	V	Alberto Franco	15-05-2002	12	173	35	street1										
2	s002	V	Gino Mureill	17-05-2002	12	192	32	street2										
3	s003	VI	Ryan Parkes	16-02-1999	13	186	33	street3										
4	s001	VI	Esha Hinton	25-09-1998	13	167	30	street1										
5	s002	V	Gino Mcneill	11-05-2002	14	151	31	street2										
6	s004	V	David Parkes	15-09-1997	12	159	32	street4										
7																		
8																		
9																		
10																		
11																		
12																		
13																		
14																		
15																		
16																		
17																		
18																		
19																		
20																		
21																		
22																		
23																		
24																		
25																		
26																		
27																		
28																		

school

Ready Accessibility: Unavailable

# Output:

Run mean\_meadiana\_mode

↑  
↓  
⇌  
⇓  
📄  
🗑️

```
"C:\Users\maano_0waenfu\OneDrive\College\Query Processing\Codes\.venv\Scripts\python.exe" "C:\Us

      mean  min  max
school_code
s001      12.5  12   13
s002      13.0  12   14
s003      13.0  13   13
s004      12.0  12   12


Process finished with exit code 0
```

## Experiment 18

### Aim:

To develop a Pandas program to split the dataset into groups based on school code and class.

## Code:



The screenshot shows a code editor with a dark theme. The top menu bar includes File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, and Help. Below the menu bar, there's a toolbar with icons for Codes, Version control, Current File, Run, Debug, and other tools. The main editor area displays a Python script named 'groupby 2.py'. The script imports pandas as pd, reads a CSV file 'school.csv', and groups the data by 'school\_code' and 'class'. It then iterates over the groups and prints the group details.

```
1 import pandas as pd
2
3 df = pd.read_csv('school.csv')
4
5 grouped = df.groupby(['school_code', 'class'])
6
7 for (school_code, class_number), group in grouped:
8     print(f"Group (school_code: {school_code}, class: {class_number}):")
9     print(group)
10    print()
11
```

The bottom status bar shows the current file is 'groupby 2.py', the cursor is at line 11:1, and the encoding is CRLF. The Python version is 3.12 (Codes).

**Input:**

Screenshot of Microsoft Excel showing a spreadsheet titled "school - Excel". The ribbon includes File, Home, Insert, Draw, Page Layout, Formulas, Data, Review, View, Help, Acrobat, and Tell me what you want to do.

The spreadsheet contains data for school records:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
	school_code	class	name	date_of_bir_ago		height	weight	address										
1	s001	V	Alberto Franco	15-05-2002	12	173	35	street1										
2	s002	V	Gino Muneill	17-05-2002	12	192	32	street2										
3	s003	VI	Ryan Parkes	16-02-1999	13	186	33	street3										
4	s001	VI	Esha Hinton	25-09-1998	13	167	30	street3										
5	s002	V	Gino Mcneill	11-05-2002	14	151	31	street2										
6	s004	V	David Parkes	15-09-1997	12	159	32	street4										
7																		
8																		
9																		
10																		
11																		
12																		
13																		
14																		
15																		
16																		
17																		
18																		
19																		
20																		
21																		
22																		
23																		
24																		
25																		
26																		
27																		
28																		

The status bar at the bottom shows "Ready" and "Accessibilities Unavailable".

## Output:

```
Run groupby 2 x
```

```
Group (school_code: s001, class: V):
  school_code class      name date_of_birth  age  height  weight  address
0      s001      V  Alberto Franco   15-05-2002   12    173    35  street1

Group (school_code: s001, class: VI):
  school_code class      name date_of_birth  age  height  weight  address
3      s001      VI  Eesha Hinton   25-09-1998   13    167    30  street1

Group (school_code: s002, class: V):
  school_code class      name date_of_birth  age  height  weight  address
1      s002      V   Gino McNeill   17-05-2002   12    192    32  stree2
4      s002      V   Gino McNeill   11-05-2002   14    151    31  street2

Group (school_code: s003, class: VI):
  school_code class      name date_of_birth  age  height  weight  address
2      s003      VI   Ryan Parkes   16-02-1999   13    186    33  street3

Group (school_code: s004, class: V):
  school_code class      name date_of_birth  age  height  weight  address
5      s004      V   David Parkes   15-09-1997   12    159    32  street4
```

# Experiment 19

## Aim:

To develop a Pandas program to split the dataset into groups based on school code and class.

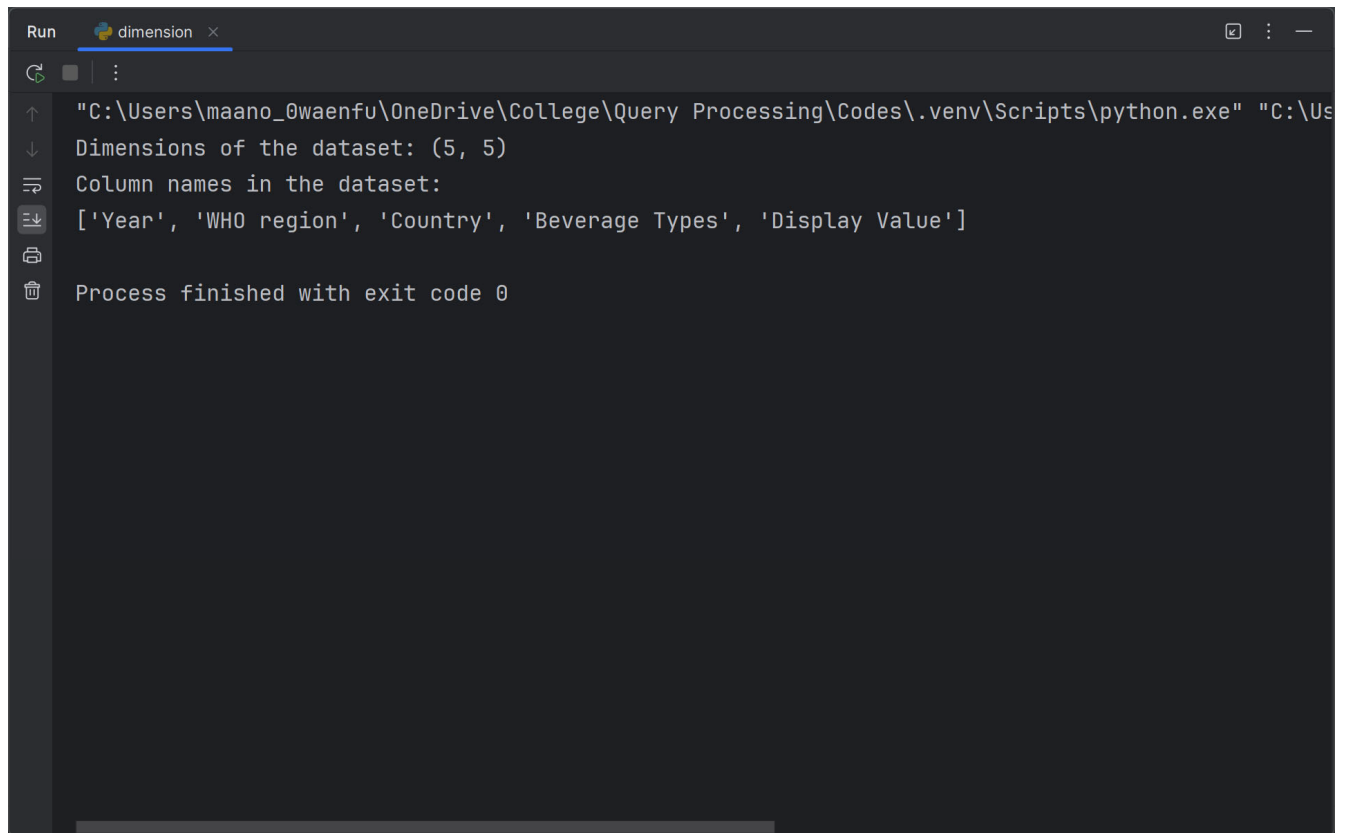
## Code:

```
File Edit View Navigate Code Refactor Run Tools VCS Window Help
Codes Version control Current File
dimension.py
1 import pandas as pd
2
3 df = pd.read_csv('wine_consumption.csv')
4
5 print("Dimensions of the dataset:", df.shape)
6
7 print("Column names in the dataset:")
8 print(df.columns.tolist())
9
```

## Input:

Year	WHO region	Country	Beverage Types	Display Value
1986	Western Pacific	Vietnam	Wine	0
1986	Americas	Uruguay	Other	0.5
1985	Africa	Cte d'Ivoire	Wine	1.62
1986	Americas	Colombia	Beer	4.27
1987	Americas	Saint Kitts and Nevis	Beer	1.98

## Output:



```
Run dimension x
"C:\Users\maano_0waenfu\OneDrive\College\Query Processing\Codes\.venv\Scripts\python.exe" "C:\Us
Dimensions of the dataset: (5, 5)
Column names in the dataset:
['Year', 'WHO region', 'Country', 'Beverage Types', 'Display Value']
Process finished with exit code 0
```

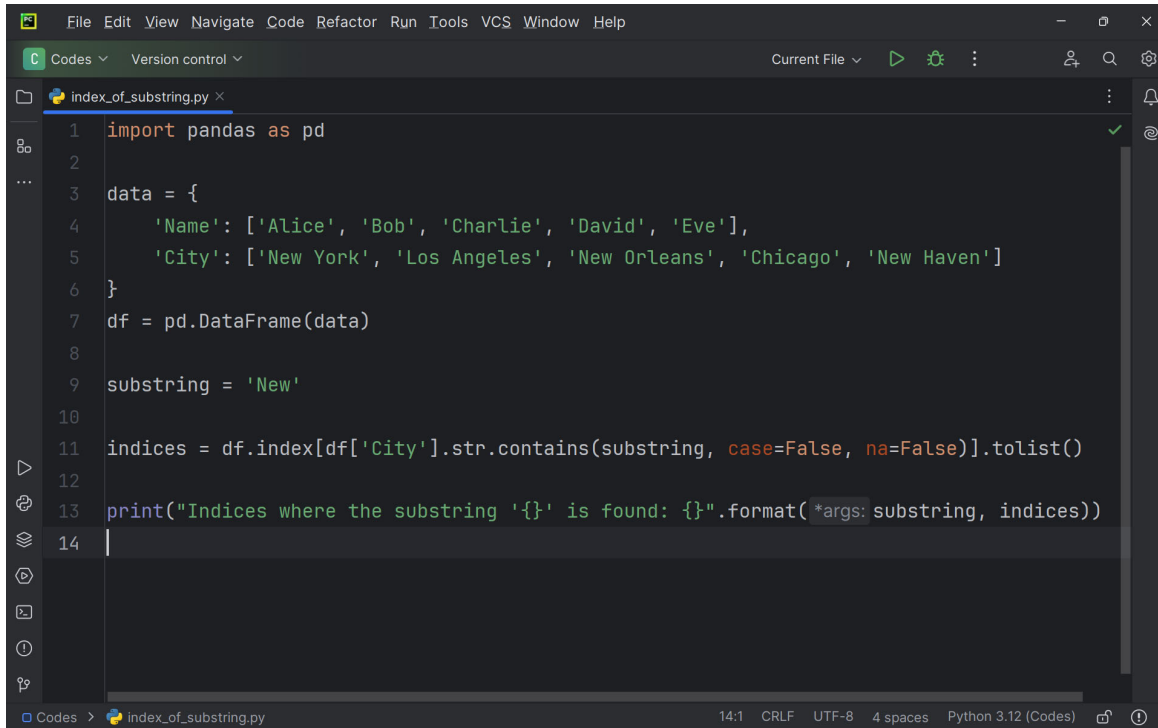


## Experiment 20

### Aim:

To develop a Pandas program to split the dataset into groups based on school code and class.

### Code:

A screenshot of a code editor window titled 'index\_of\_substring.py'. The editor shows a Python script that uses Pandas to find indices of a substring in a DataFrame. The script is as follows:

```
1 import pandas as pd
2
3 data = {
4     'Name': ['Alice', 'Bob', 'Charlie', 'David', 'Eve'],
5     'City': ['New York', 'Los Angeles', 'New Orleans', 'Chicago', 'New Haven']
6 }
7 df = pd.DataFrame(data)
8
9 substring = 'New'
10
11 indices = df.index[df['City'].str.contains(substring, case=False, na=False)].tolist()
12
13 print("Indices where the substring '{}' is found: {}".format(*args: substring, indices))
14
```

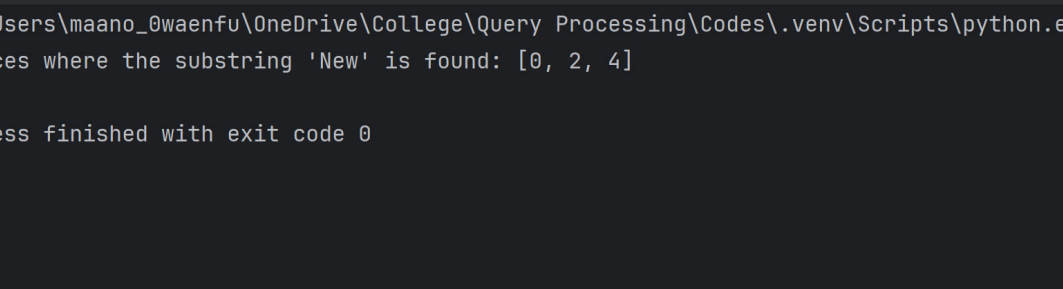
The editor interface includes a menu bar (File, Edit, View, Navigate, Code, Refactor, Run, Tools, VCS, Window, Help), a toolbar with icons for running, debugging, and other actions, and a status bar at the bottom showing file encoding (UTF-8), line length (14:1), and Python version (3.12).

### Input:

```
data = {
    'Name': ['Alice', 'Bob', 'Charlie', 'David', 'Eve'],
    'City': ['New York', 'Los Angeles', 'New Orleans', 'Chicago', 'New Haven']
}
df = pd.DataFrame(data)

substring = 'New'
```

## Output:



```
Run index_of_substring x
"C:\Users\maano_0waenfu\OneDrive\College\Query Processing\Codes\.venv\Scripts\python.exe" "C:\Us
Indices where the substring 'New' is found: [0, 2, 4]

Process finished with exit code 0
```