Міністерство освіти і науки України
Національний технічний університет України
“Київський політехнічний інститут імені Ігоря Сікорського”
Факультет інформатики та обчислювальної техніки
Кафедра автоматики та управління в технічних системах

Лабораторна робота №2

Індексування, вибір, редагування набору даних

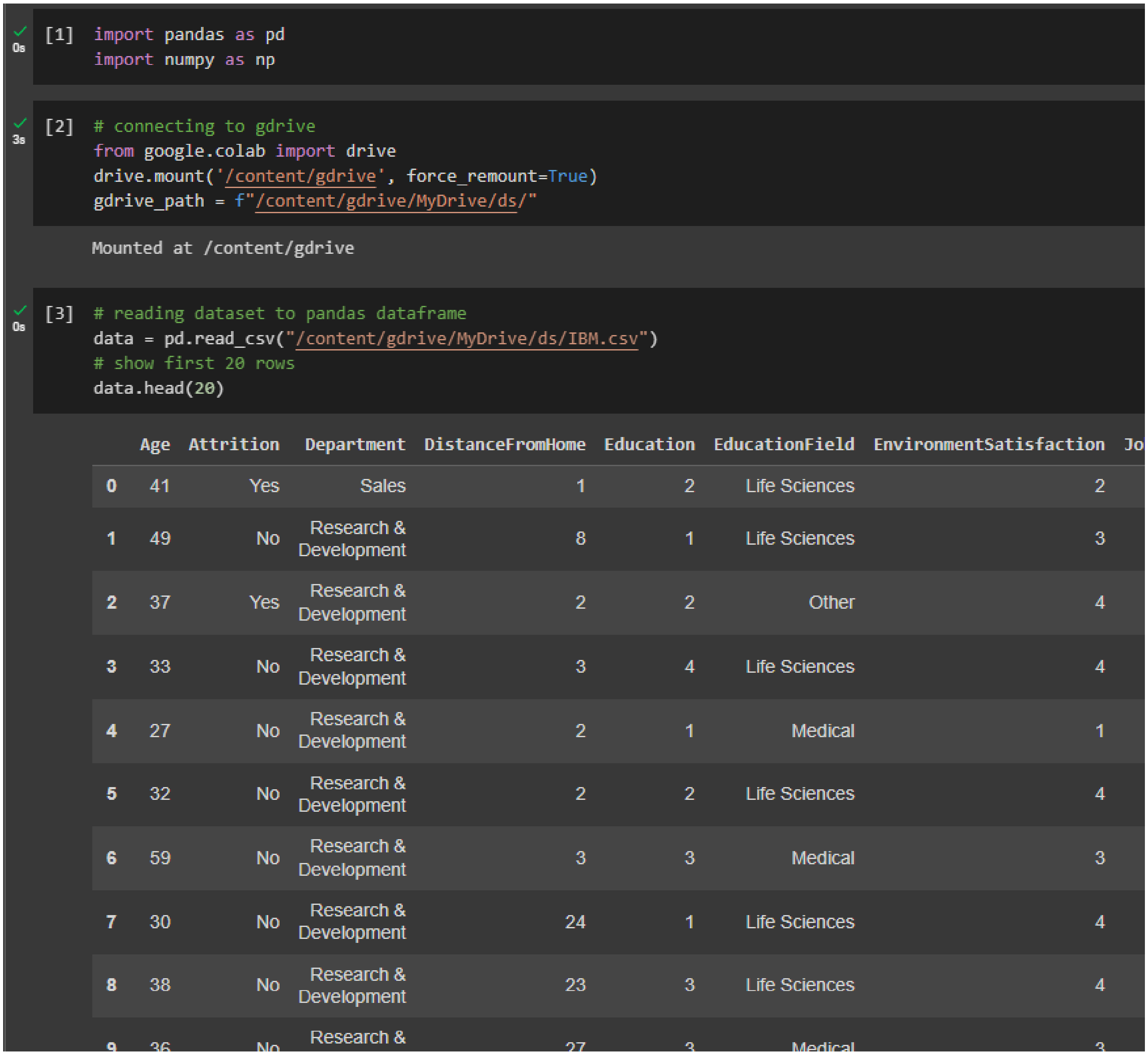
Виконала	
студентка групи ІТ-91:	Перевірив:
Луцай Катерина	Новотарський М. А.

Київ 2023

Мета: навчитися швидко та ефективно вибирати необхідні дані з набору даних з використанням структур даних та інструментів бібліотеки Pandas..

Варіант: 15 – dataset “IBM Attrition Dataset” (https://www.kaggle.com/yasserh/ibm-attrition-dataset)

Хід виконання роботи:



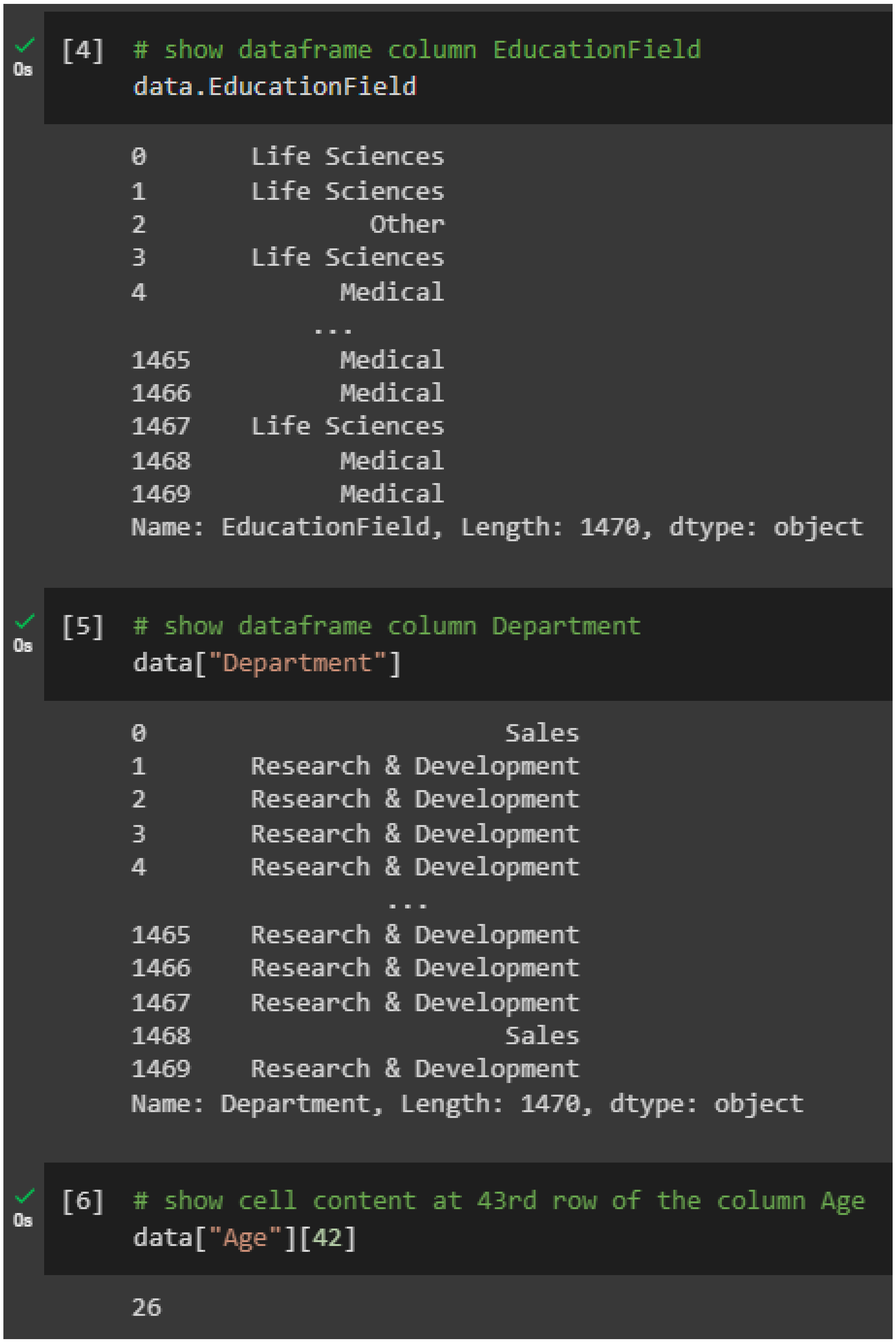
```
[1] import pandas as pd
    import numpy as np
```

```
[2] # connecting to gdrive
    from google.colab import drive
    drive.mount('/content/gdrive', force_remount=True)
    gdrive_path = f"/content/gdrive/MyDrive/ds/"

    Mounted at /content/gdrive
```

```
[3] # reading dataset to pandas dataframe
    data = pd.read_csv("/content/gdrive/MyDrive/ds/IBM.csv")
    # show first 20 rows
    data.head(20)
```

	Age	Attrition	Department	DistanceFromHome	Education	EducationField	EnvironmentSatisfaction	Jo
0	41	Yes	Sales	1	2	Life Sciences	2	
1	49	No	Research & Development	8	1	Life Sciences	3	
2	37	Yes	Research & Development	2	2	Other	4	
3	33	No	Research & Development	3	4	Life Sciences	4	
4	27	No	Research & Development	2	1	Medical	1	
5	32	No	Research & Development	2	2	Life Sciences	4	
6	59	No	Research & Development	3	3	Medical	3	
7	30	No	Research & Development	24	1	Life Sciences	4	
8	38	No	Research & Development	23	3	Life Sciences	4	
9	36	No	Research &	27	3	Medical	3	



```
[4]  # show dataframe column EducationField
     data.EducationField
```

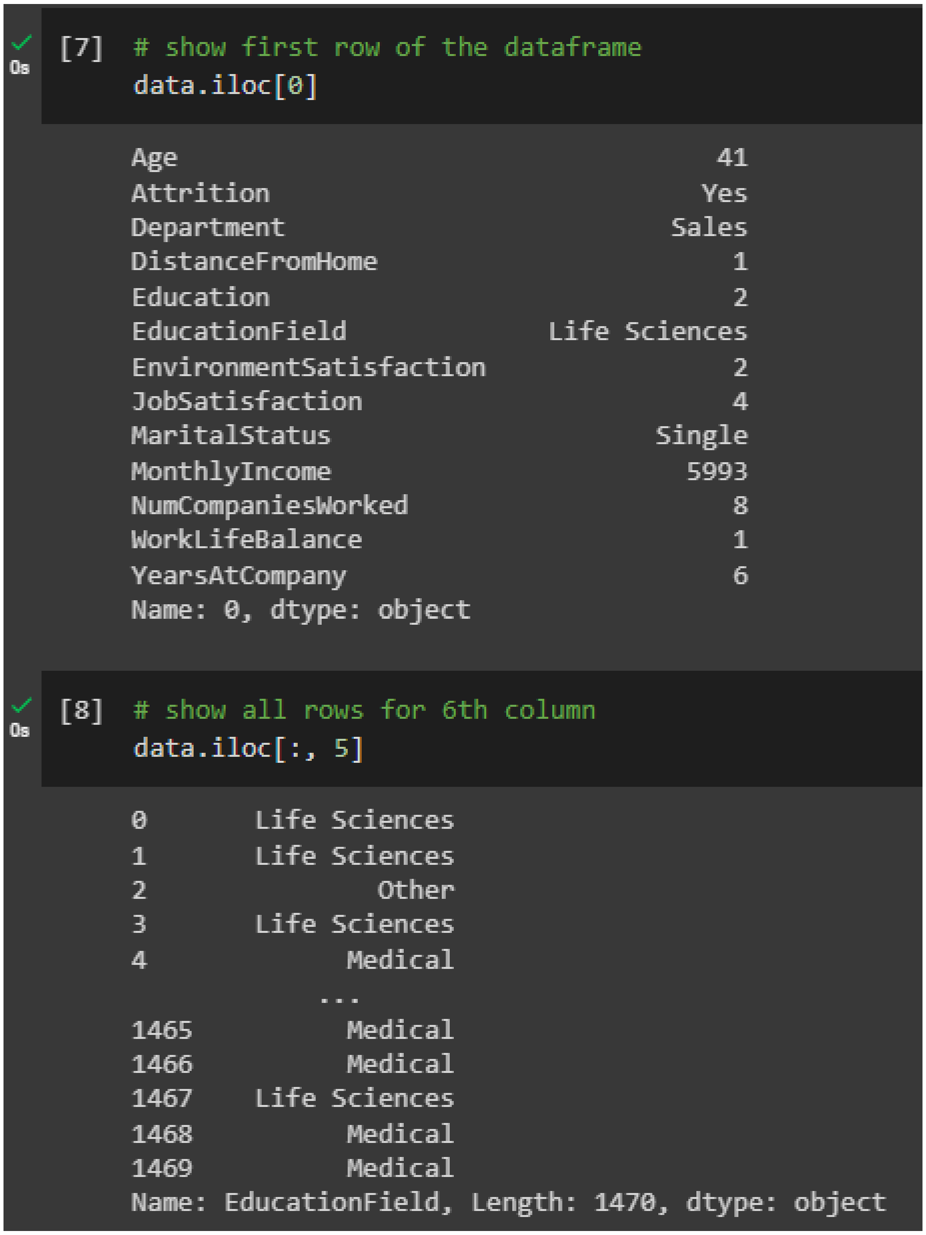
```
0       Life Sciences
1       Life Sciences
2               Other
3       Life Sciences
4             Medical
            ...
1465          Medical
1466          Medical
1467    Life Sciences
1468          Medical
1469          Medical
Name: EducationField, Length: 1470, dtype: object
```

```
[5]  # show dataframe column Department
     data["Department"]
```

```
0                        Sales
1       Research & Development
2       Research & Development
3       Research & Development
4       Research & Development
                 ...
1465    Research & Development
1466    Research & Development
1467    Research & Development
1468                     Sales
1469    Research & Development
Name: Department, Length: 1470, dtype: object
```

```
[6]  # show cell content at 43rd row of the column Age
     data["Age"][42]
```

```
26
```

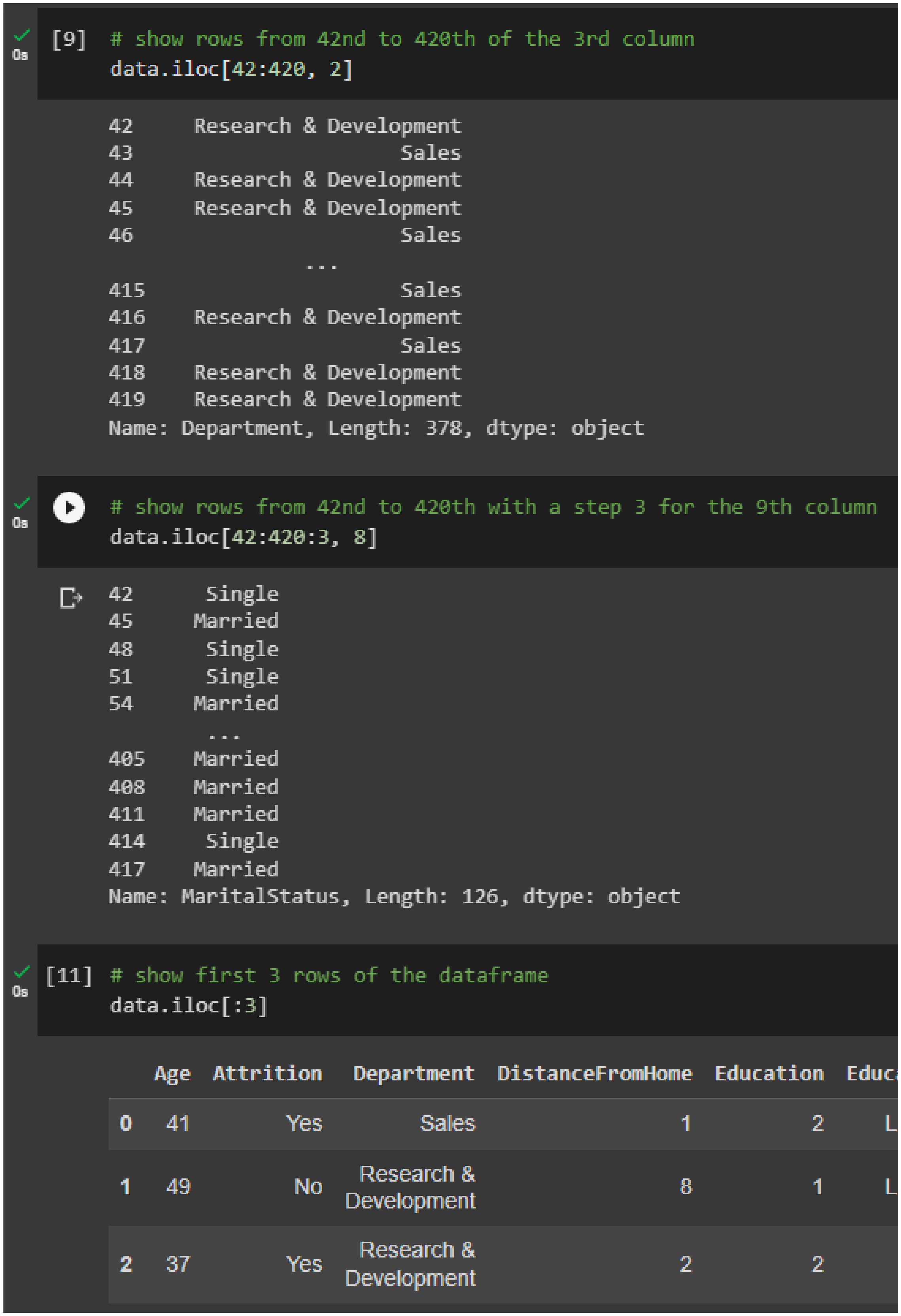


```
# show first row of the dataframe
data.iloc[0]
```

```
Age                                   41
Attrition                            Yes
Department                         Sales
DistanceFromHome                       1
Education                              2
EducationField             Life Sciences
EnvironmentSatisfaction                2
JobSatisfaction                        4
MaritalStatus                     Single
MonthlyIncome                       5993
NumCompaniesWorked                     8
WorkLifeBalance                        1
YearsAtCompany                         6
Name: 0, dtype: object
```

```
# show all rows for 6th column
data.iloc[:, 5]
```

```
0       Life Sciences
1       Life Sciences
2               Other
3       Life Sciences
4             Medical
            ...
1465          Medical
1466          Medical
1467    Life Sciences
1468          Medical
1469          Medical
Name: EducationField, Length: 1470, dtype: object
```



```
[9] # show rows from 42nd to 420th of the 3rd column
    data.iloc[42:420, 2]
```

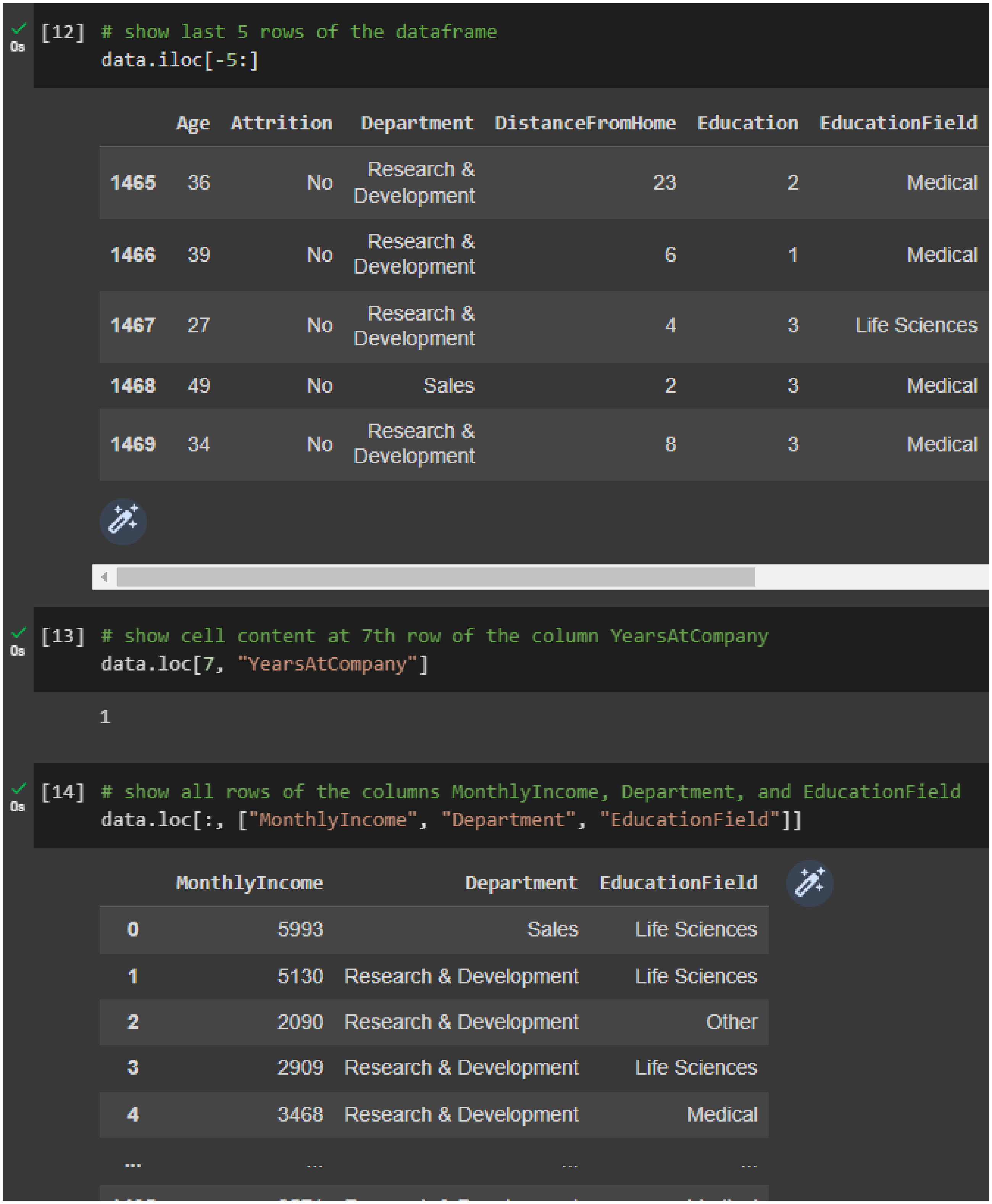
```
42      Research & Development
43                       Sales
44      Research & Development
45      Research & Development
46                       Sales
                 ...
415                      Sales
416     Research & Development
417                      Sales
418     Research & Development
419     Research & Development
Name: Department, Length: 378, dtype: object
```

```
# show rows from 42nd to 420th with a step 3 for the 9th column
data.iloc[42:420:3, 8]
```

```
42      Single
45     Married
48      Single
51      Single
54     Married
         ...
405    Married
408    Married
411    Married
414     Single
417    Married
Name: MaritalStatus, Length: 126, dtype: object
```

```
[11] # show first 3 rows of the dataframe
     data.iloc[:3]
```

	Age	Attrition	Department	DistanceFromHome	Education	Educa
0	41	Yes	Sales	1	2	L
1	49	No	Research & Development	8	1	L
2	37	Yes	Research & Development	2	2	



```
[12] # show last 5 rows of the dataframe
     data.iloc[-5:]
```

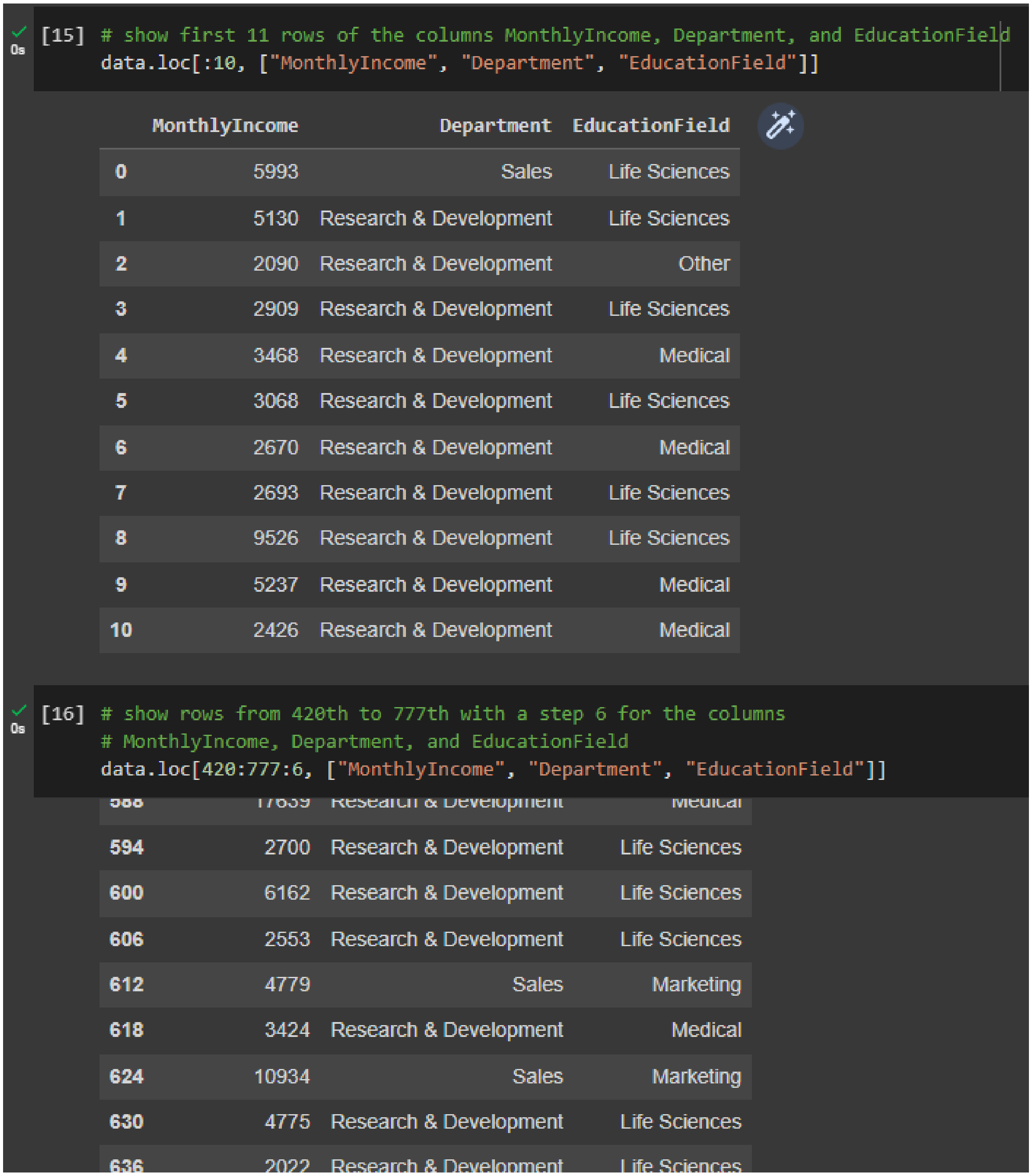
	Age	Attrition	Department	DistanceFromHome	Education	EducationField
1465	36	No	Research & Development	23	2	Medical
1466	39	No	Research & Development	6	1	Medical
1467	27	No	Research & Development	4	3	Life Sciences
1468	49	No	Sales	2	3	Medical
1469	34	No	Research & Development	8	3	Medical

```
[13] # show cell content at 7th row of the column YearsAtCompany
     data.loc[7, "YearsAtCompany"]
```

```
1
```

```
[14] # show all rows of the columns MonthlyIncome, Department, and EducationField
     data.loc[:, ["MonthlyIncome", "Department", "EducationField"]]
```

	MonthlyIncome	Department	EducationField
0	5993	Sales	Life Sciences
1	5130	Research & Development	Life Sciences
2	2090	Research & Development	Other
3	2909	Research & Development	Life Sciences
4	3468	Research & Development	Medical
...	...	...	...

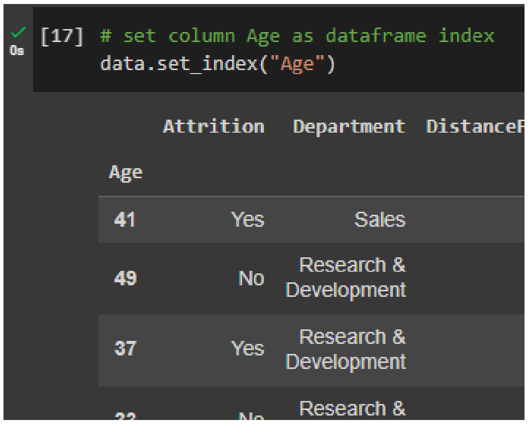


```
[15] # show first 11 rows of the columns MonthlyIncome, Department, and EducationField
     data.loc[:10, ["MonthlyIncome", "Department", "EducationField"]]
```

```
    MonthlyIncome              Department  EducationField
0            5993                   Sales   Life Sciences
1            5130  Research & Development   Life Sciences
2            2090  Research & Development           Other
3            2909  Research & Development   Life Sciences
4            3468  Research & Development         Medical
5            3068  Research & Development   Life Sciences
6            2670  Research & Development         Medical
7            2693  Research & Development   Life Sciences
8            9526  Research & Development   Life Sciences
9            5237  Research & Development         Medical
10           2426  Research & Development         Medical
```

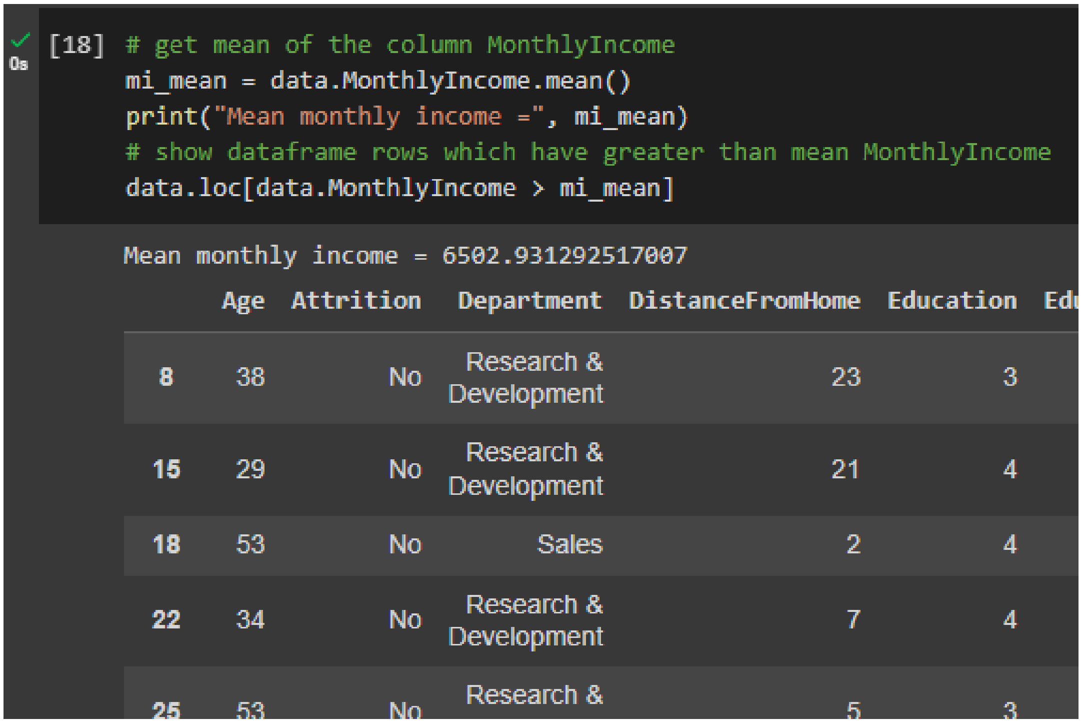
```
[16] # show rows from 420th to 777th with a step 6 for the columns
     # MonthlyIncome, Department, and EducationField
     data.loc[420:777:6, ["MonthlyIncome", "Department", "EducationField"]]
```

```
588    17639  Research & Development        Medical
594     2700  Research & Development  Life Sciences
600     6162  Research & Development  Life Sciences
606     2553  Research & Development  Life Sciences
612     4779                   Sales      Marketing
618     3424  Research & Development        Medical
624    10934                   Sales      Marketing
630     4775  Research & Development  Life Sciences
636     2022  Research & Development  Life Sciences
```



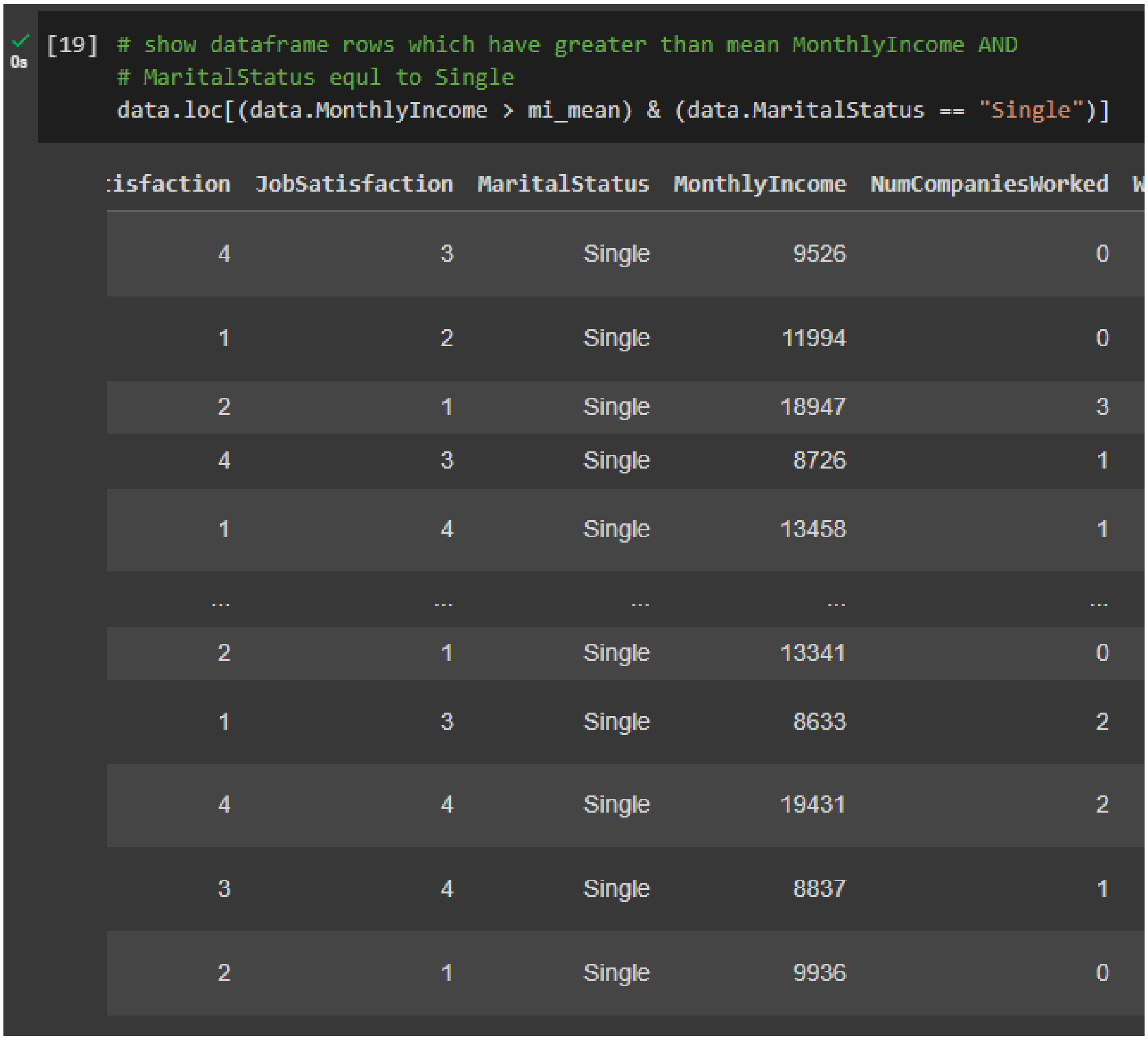
```
[17] # set column Age as dataframe index
     data.set_index("Age")
```

```
      Attrition  Department   DistanceF
Age
41          Yes       Sales
49           No  Research &
                Development
37          Yes  Research &
                Development
33           No  Research &
```



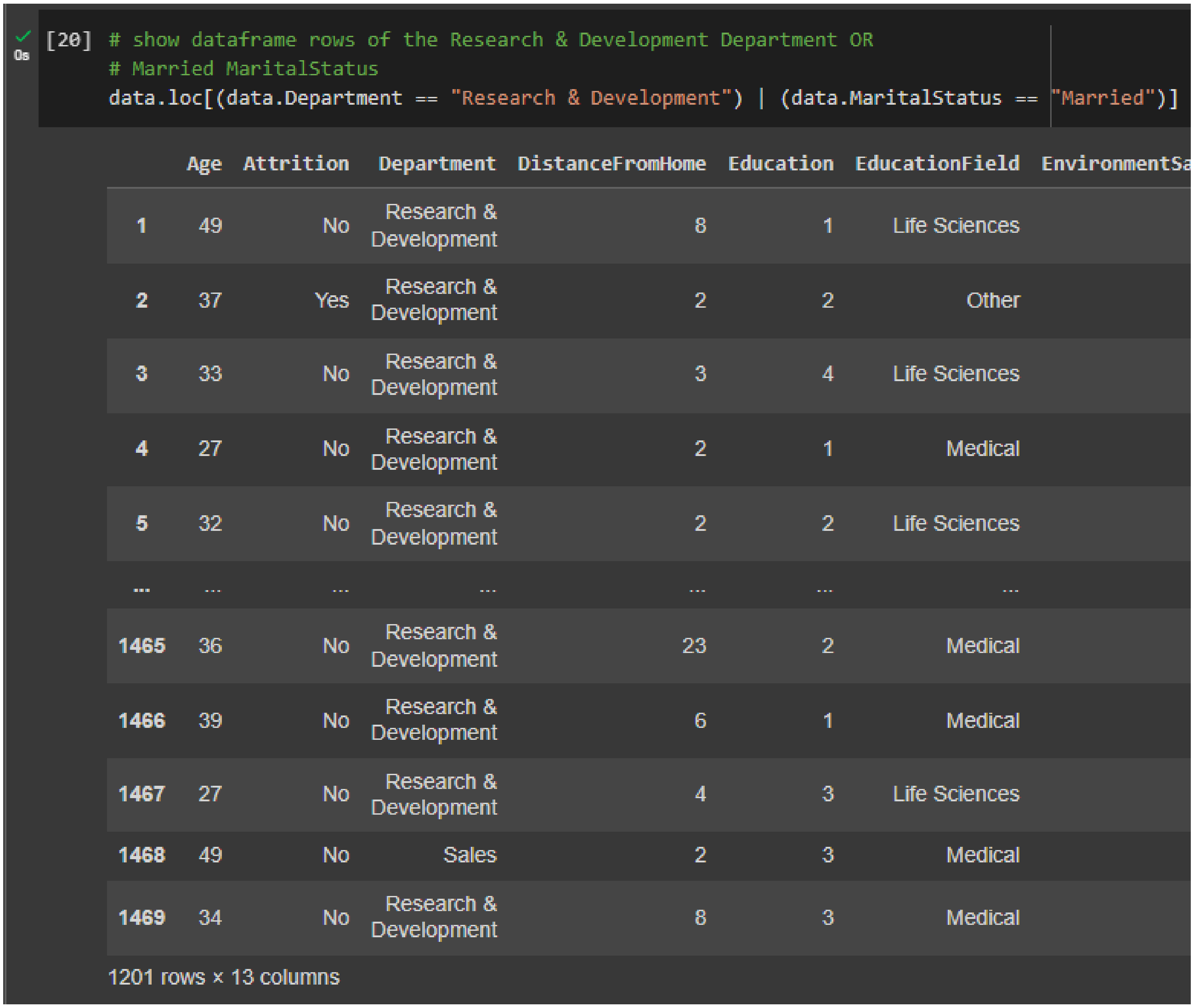
```
[18] # get mean of the column MonthlyIncome
     mi_mean = data.MonthlyIncome.mean()
     print("Mean monthly income =", mi_mean)
     # show dataframe rows which have greater than mean MonthlyIncome
     data.loc[data.MonthlyIncome > mi_mean]
```

```
Mean monthly income = 6502.931292517007
       Age  Attrition   Department   DistanceFromHome  Education  Edu
 8      38         No   Research &                 23          3
                        Development
15      29         No   Research &                 21          4
                        Development
18      53         No        Sales                  2          4
22      34         No   Research &                  7          4
                        Development
25      53         No   Research &                  5          3
```



```
[19] # show dataframe rows which have greater than mean MonthlyIncome AND
     # MaritalStatus equl to Single
     data.loc[(data.MonthlyIncome > mi_mean) & (data.MaritalStatus == "Single")]
```

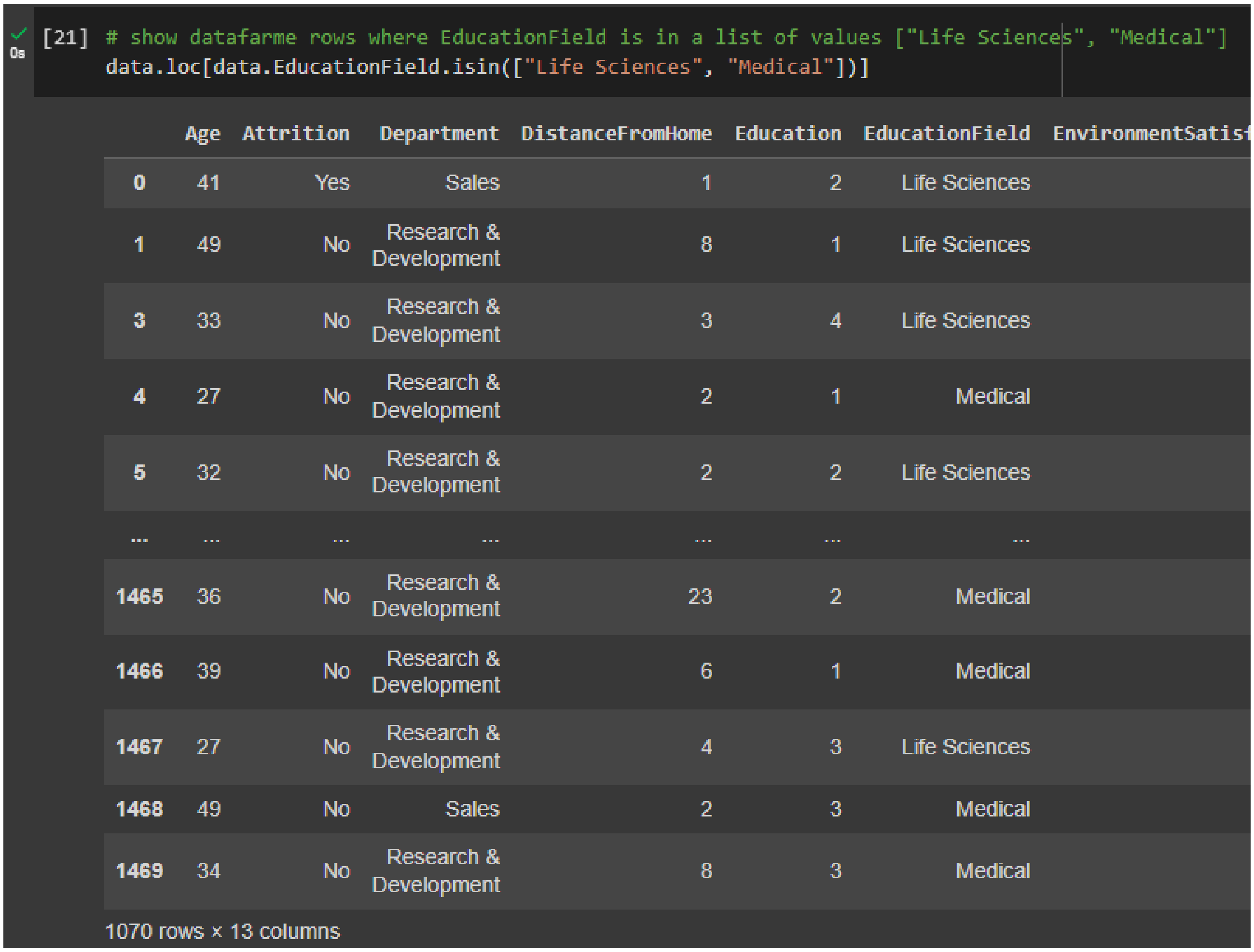
```
:isfaction  JobSatisfaction  MaritalStatus  MonthlyIncome  NumCompaniesWorked  W
         4                3         Single           9526                   0
         1                2         Single          11994                   0
         2                1         Single          18947                   3
         4                3         Single           8726                   1
         1                4         Single          13458                   1
       ...              ...            ...            ...                 ...
         2                1         Single          13341                   0
         1                3         Single           8633                   2
         4                4         Single          19431                   2
         3                4         Single           8837                   1
         2                1         Single           9936                   0
```



```
# show dataframe rows of the Research & Development Department OR
# Married MaritalStatus
data.loc[(data.Department == "Research & Development") | (data.MaritalStatus == "Married")]
```

	Age	Attrition	Department	DistanceFromHome	Education	EducationField	EnvironmentSa
1	49	No	Research & Development	8	1	Life Sciences	
2	37	Yes	Research & Development	2	2	Other	
3	33	No	Research & Development	3	4	Life Sciences	
4	27	No	Research & Development	2	1	Medical	
5	32	No	Research & Development	2	2	Life Sciences	
...	...	...	...	...	...	...	
1465	36	No	Research & Development	23	2	Medical	
1466	39	No	Research & Development	6	1	Medical	
1467	27	No	Research & Development	4	3	Life Sciences	
1468	49	No	Sales	2	3	Medical	
1469	34	No	Research & Development	8	3	Medical	

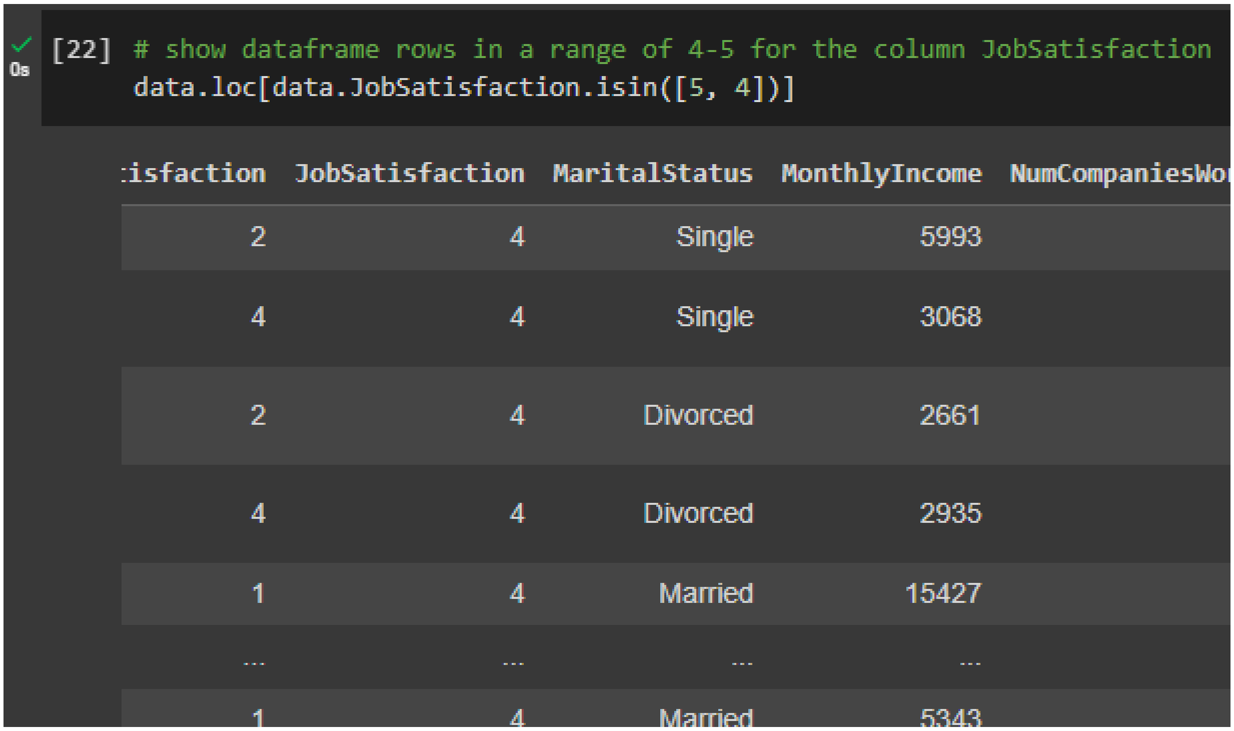
1201 rows × 13 columns



```
[21] # show dataframe rows where EducationField is in a list of values ["Life Sciences", "Medical"]
     data.loc[data.EducationField.isin(["Life Sciences", "Medical"])]
```

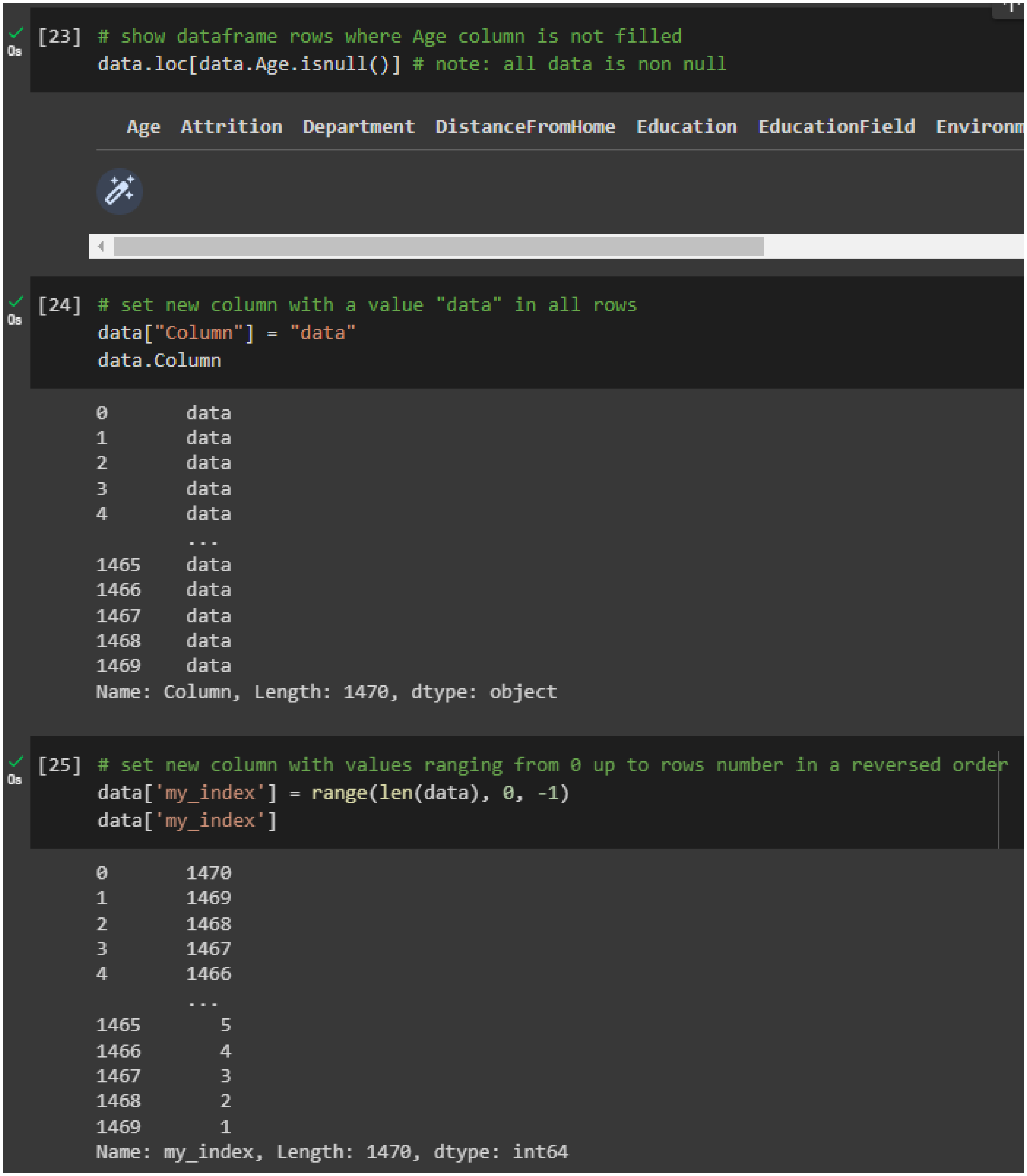
	Age	Attrition	Department	DistanceFromHome	Education	EducationField	EnvironmentSatisf
0	41	Yes	Sales	1	2	Life Sciences	
1	49	No	Research & Development	8	1	Life Sciences	
3	33	No	Research & Development	3	4	Life Sciences	
4	27	No	Research & Development	2	1	Medical	
5	32	No	Research & Development	2	2	Life Sciences	
...	...	...	...	...	...	...	
1465	36	No	Research & Development	23	2	Medical	
1466	39	No	Research & Development	6	1	Medical	
1467	27	No	Research & Development	4	3	Life Sciences	
1468	49	No	Sales	2	3	Medical	
1469	34	No	Research & Development	8	3	Medical	

1070 rows × 13 columns



```
[22] # show dataframe rows in a range of 4-5 for the column JobSatisfaction
     data.loc[data.JobSatisfaction.isin([5, 4])]
```

:isfaction	JobSatisfaction	MaritalStatus	MonthlyIncome	NumCompaniesWo
2	4	Single	5993	
4	4	Single	3068	
2	4	Divorced	2661	
4	4	Divorced	2935	
1	4	Married	15427	
...	...	...	...	
1	4	Married	5343	



```
[23] # show dataframe rows where Age column is not filled
     data.loc[data.Age.isnull()] # note: all data is non null
```

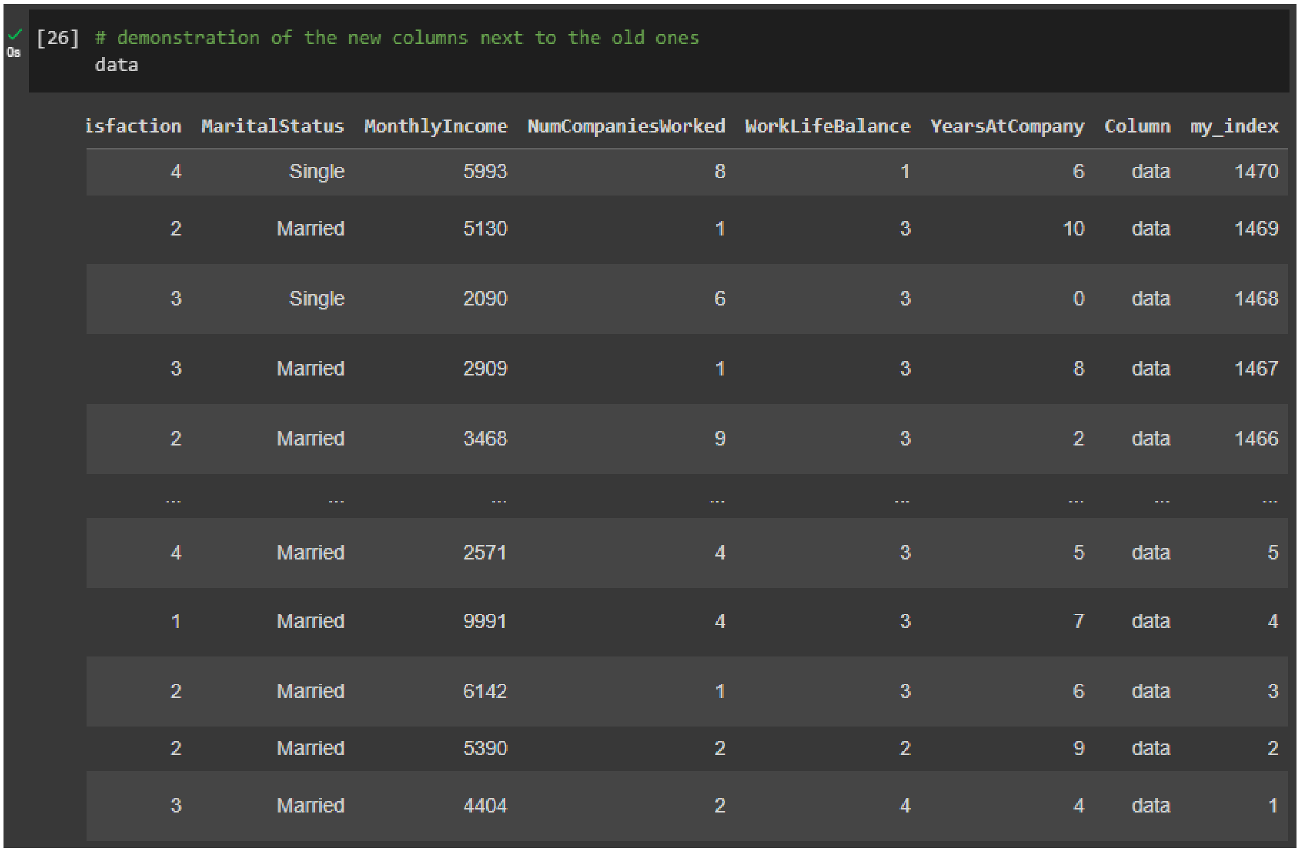
```
     Age  Attrition  Department  DistanceFromHome  Education  EducationField  Environm
```

```
[24] # set new column with a value "data" in all rows
     data["Column"] = "data"
     data.Column
```

```
0       data
1       data
2       data
3       data
4       data
        ...
1465    data
1466    data
1467    data
1468    data
1469    data
Name: Column, Length: 1470, dtype: object
```

```
[25] # set new column with values ranging from 0 up to rows number in a reversed order
     data['my_index'] = range(len(data), 0, -1)
     data['my_index']
```

```
0       1470
1       1469
2       1468
3       1467
4       1466
        ...
1465       5
1466       4
1467       3
1468       2
1469       1
Name: my_index, Length: 1470, dtype: int64
```



```
[26] # demonstration of the new columns next to the old ones
     data
```

isfaction	MaritalStatus	MonthlyIncome	NumCompaniesWorked	WorkLifeBalance	YearsAtCompany	Column	my_index
4	Single	5993	8	1	6	data	1470
2	Married	5130	1	3	10	data	1469
3	Single	2090	6	3	0	data	1468
3	Married	2909	1	3	8	data	1467
2	Married	3468	9	3	2	data	1466
...	...	...	...	...	...	...	...
4	Married	2571	4	3	5	data	5
1	Married	9991	4	3	7	data	4
2	Married	6142	1	3	6	data	3
2	Married	5390	2	2	9	data	2
3	Married	4404	2	4	4	data	1

Вихідний код у jupyter notebook: https://colab.research.google.com/drive/1SYHbXfiYp_b_Q9iZT0qkQKioYKf-njLw?usp=sharing

Висновки: було розглянуто основні методи мови Python для швидкого та ефективного вибору необхідних даних із набору даних з використанням структур даних та інструментів бібліотеки Pandas.