

V2_using_pandas

November 1, 2025

1 Activity: Using Pandas

1.1 Introduction

In this activity you will practice using some of the basic functionality associated with Pandas. This activity will cover the following topics:

- Creating a `DataFrame`
- Displaying `DataFrame` information
- Accessing column data
- Getting ranges of column data
- Creating Series from dictionaries
- Using the `iloc()` and `loc()` methods
- Getting data from multiple columns
- Getting rows

Question 1 Create a DataFrame called `df` from the given CSV file `student_data.csv` then using the `df` DataFrame, assign the `Name` column to a Series called `names`.

```
[8]: import pandas as pd
# Your code here

df = pd.read_csv('student_data.csv')

names = df['Name']

print(names)

#df.head()

df.tail()
```

```
0      Jennifer Jackson
1      Michael Johnson
2          Robert Lee
3      Linda Harris
4      Michael Moore
       ..
70      William Johnson
71          Linda Smith
72      Jennifer Lee
73      Michael Miller
74      Emily Johnson
Name: Name, Length: 75, dtype: object
```

```
[8]:      Name  Age  Math Grade  English Grade
 70  William Johnson    17        99            75
 71      Linda Smith    18        92            83
 72      Jennifer Lee   17        75            88
 73  Michael Miller    12        86            84
 74  Emily Johnson    13        94            100
```

```
[9]: # Question 1 Grading Checks
```

```
assert isinstance(df, pd.DataFrame), "Did you create a DataFrame called df?"
assert isinstance(names, pd.Series), "Did you assign the Name column in a
→variable called names?"
```

Question 2 Using the df DataFrame, assign the Age and Math Grade columns to a DataFrame called age_math.

```
[11]: # Your code here
```

```
age_math = df[['Age', 'Math Grade']]
age_math.head()
```

```
[11]:    Age  Math Grade
 0     14        84
 1     14        92
 2     18        87
 3     13        90
 4     18        88
```

```
[12]: # Question 2 Grading Checks
```

```
assert isinstance(age_math, pd.DataFrame), "Did you assign the Age and Math
→Grade columns to a variable called age_math?"
```

Question 3 Using the .loc() method, assign the Age and Math Grade columns for the first 30 rows of df to a variable called first_thirty_loc.

```
[13]: # Your code here
```

```
first_thirty_loc = age_math.loc[0:29]
```

```
[14]: # Question 3 Grading Checks
```

```
assert isinstance(first_thirty_loc, pd.DataFrame), "Did you correctly assign
→the first 30 rows to a variable called first_thirty_loc?"
```

Question 4 Get the even numbered rows from the Name and English Grade columns and assign the result to a variable called `even_rows_english`.

[18]: # Your code here

```
# even_rows_english = df.loc[df.index % 2 == 0, ["Name", "English Grade"]]

even_rows_english = df.loc[0::2, ["Name", "English Grade"]]

#even_rows_english = df[['Name', English Grade']]
```

[19]: # Question 4 Grading Checks

```
assert isinstance(even_rows_english, pd.DataFrame), "Did you correctly assign  
→even numbered rows to a variable called even_rows_english? Hint: the first  
→row is index 0."
```