PROJECT

Gradebook

You are a student and you are trying to organize your subjects and grades using Python. Let’s explore what we’ve learned about lists to organize your subjects and scores.

Tasks

10/10 Complete

Mark the tasks as complete by checking them off

Create Some Lists:

1.

Create a list called `subjects` and fill it with the classes you are taking:

- `"physics"`
- `"calculus"`
- `"poetry"`
- `"history"`

Hint

To create a new list with values, you can use the syntax:

```
my_list = [val_1, val_2, val_3, val_4]
```

2.

Create a list called `grades` and fill it with your scores:

- `98`
- `97`
- `85`
- `88`

Hint

To create a new list with values, you can use the syntax:

```
my_list = [val_1, val_2, val_3, val_4]
```

3.

Manually (without any methods) create a two-dimensional list to combine `subjects` and `grades`. Use the table below as a reference to associated values.

Name	Test Score
`"physics"`	`98`
`"calculus"`	`97`
`"poetry"`	`85`
`"history"`	`88`

Assign the value into a variable called `gradebook`.

Hint

Remember the key components of a two-dimensional list:

1. A two-dimensional list begins and ends with square brackets ([and]). This is our "container" list that wraps all of our inner sublists.
2. Any number of sublists within the "container" list. These are our inner lists.
3. Each item is separated by a comma (,) both in the inner and outer lists.

```
#Outermost "container" list
example_2d_list = [
  #Innermost sublists
  ["First Sublist"],
  ["Second Sublist"],
  ["Third Sublist"]
]
```

4.

Print `gradebook`.

Does it look how you expected it would?

Hint

The output should look like this:

```
[['physics', 98], ['calculus', 97], ['poetry', 85], ['history', 88]]
```

Add More Subjects:

5.

Your grade for your computer science class just came in! You got a perfect score, `100`!

Use the `.append()` method to add a list with the values of `"computer science"` and an associated grade value of `100` to our two-dimensional list of `gradebook`.

Hint

You can add a value to the end of a list with this syntax:

```
list.append(new_value)
```

`new_value` can represent any value, even a list!

6.

Your grade for `"visual arts"` just came in! You got a 93!

Use `append` to add `["visual arts", 93]` to `gradebook`.

Hint

To add a value like this to the end of a two-dimensional list use:

```
list.append([string_value, number_value])
```

Where `string_value` represents our class name and `number_value` represents our grade

Modify The Gradebook:

7.

Our instructor just told us they made a mistake grading and are rewarding an extra 5 points for our visual arts class.

Access the index of the grade for your visual arts class and modify it to be 5 points greater.

Hint

When accessing a two-dimensional list, determine the index for both the inner and outer list.

For our outer index, since we just used `.append()` for our visual arts class, it will be at the end of the outer list (that wraps all of our classes).

To access the last element use the `-1` index.

```
# ["visual arts", 93]
gradebook[-1]
```

Now, we need to access the value of 93. In a two-dimensional list, we use another set of brackets.

```
gradebook[-1][X]
```

What would the value of `X` be to access the value `93` in the sublist `["visual arts", 93]`?

8.

You decided to switch from a numerical grade value to a Pass/Fail option for your poetry class.

Find the grade value in your `gradebook` for your poetry class and use the `.remove()` method to delete it.

Hint

Your grade for poetry is an `85` and the value exists at `gradebook[2]`. Use the `.remove()` method on this sublist and provide the value you want to remove.

```
sublist.remove(value)
```

9.
Use the `.append()` method to then add a new `"Pass"` value to the sublist where your poetry class is located.
Hint
Your grade for poetry should exist on the sublist `gradebook[2]`. Call append on this sublist with the value of `"Pass"`

```
sublist.append(value)
```

One Big Gradebook!

10.
You also have your grades from last semester, stored in `last_semester_gradebook`.

Create a new variable `full_gradebook` that combines both `last_semester_gradebook` and `gradebook` using `+` to have one complete grade book.

Print `full_gradebook` to see our completed list.
Hint
We can combine any two lists using `+`. In our case, we want to combine `last_semester_gradebook` and `gradebook`

```
full_gradebook = X + Y
```

What would `X` and `Y` need to be to complete the combination of our two books into full_gradebook?

script.py

```
last_semester_gradebook = [["politics", 80], ["latin", 96], ["dance", 97], ["architecture", 65]]

# Your code below:
subjects = ["physics", "calculus", "poetry", "history"]
grades = [98, 97, 85, 88]
gradebook = [["physics", 98], ["calculus", 97], ["poetry", 85], ["history", 88]]
#print(gradebook)

gradebook.append(["computer science", 100])
#print(gradebook)

gradebook.append(["visual arts", 93])
#print(gradebook)
```

```
gradebook[-1][-1] = 98
#print(gradebook)

gradebook[2].remove(85)
#print(gradebook)

gradebook[2].append("Pass")
#print(gradebook)

full_gradebook = last_semester_gradebook + gradebook
print(full_gradebook)
```