Programming

Winter 24/25

Exercises

Number 02, Submission Deadline: October 30, 2024

1. String formatting.

Often, computational results are reported in form of text, where several pieces of information are composed into a single sentence, e.g.: "The sum of 4 + 10 + 28 is 42". Python provides a convenient way of constructing such strings through the use of *place holders*, as shown here by two examples:

```
a = 4
b = 10
c = 28
# first example
my_string = 'The sum of {} + {} + {} is {}'.format(a, b, c, a + b + c)
print(my_string)
# second example (notice the leading "f" in front ot the string!)
my_string2 = f'The sum of {a} + {b} + {c} is {a + b + c}'
print(my_string2)
```

Read the "Guide to the Newer Python String Format Techniques" at https:// realpython.com/python-formatted-output/ to inform yourself about the format() function and f-strings.

2. Halloween theme: The Magic Potion Shop!

There's a brand new magic potion shop in town, and this week, you are the apprentice to the Head Witch managing the shop!

She needs help in organising her inventory, which is composed of potions with unique names; every potion has of course a price and a certain availability, meaning how many items of that particular potion are currently present in the inventory.

During your short internship at the shop, you'll be asked to put your programming skills to test (the head Witch can't code...) to complete a short program that prints on the screen all of the details about the potions.

But there's an additional detail: since it's almost Halloween, the head Witch asked you to develop this program for her customers to use! In fact, she has announced a special giveaway for her customers, and they will use your program to try to get lucky and win some potions.

Ready to take over the role of Head Magician? Open the file MagicPotionShop2.py and complete the code!

An example of output is the following:

(9 P)

(2 P)

		luna@dhcp-10-68-226	200. /Denuml	aha	7.8
	p-10-68-226-88 ~/Downloads				
it's vour nam		s python Magici	-occonsno	h•hÀ	
	ion shop's name: Bielefeld	's Brand New Po	otion Sho	0	
many potion	is do you want to pick (cho	ose a number b	etween 1	and 5)? 3	
	.cks: [2, 3, 4]				
	**************************************		*******	*****	
	efeld's Brand New Potion S. by chief witch's brand new		ina		
	itly has your 3 lucky potic				
Index	Name	Quantity		t	
2 3	Invisibility Potion Strength Potion		50 35		
5 4	Zero Bugs Potion	4	1 50		

3. What type of programmer are you?

You are creating your character alter ego for a video game. The character's attributes (health, speed, eyesight, gold) depend on the choices you make. Your task is to write a Python program that:

(9 P)

- (a) Sets the initial values for health, speed, eyesight and gold.
- (b) Asks the player to choose one of three special items (coffee, the Python Docs, or ChatGPT) which will modify their attributes.
- (c) Based on the attributes and choices, the player either gets into a fight, has a rare encounter or avoids danger and finds a treasure.

Let's break the problem down into little pieces:

- (a) Character initialisation:
 - Start by setting your health, speed, eyesight and gold to some default values. For instance:

```
health = 100
speed = 50
eyesight = 50
gold = 20
```

(b) Item choice:

- The player will choose a special item between coffee, Python Docs and ChatGPT.
- Each item affects the character's attributes:
 - Coffee: +10 speed, -30 health
 - Python Docs: +10 health, -30 eyesight
 - ChatGPT: -10 health, +30 speed

- (c) Encounter: based on your new stats, you will have one of three encounters:
 - If speed is greater than 70, you fight a hacker;
 - If health is less or equal than 70, you meet Morpheus;
 - If neither of these conditions is true, you wander safely and find some extra gold.
- (d) Final outcome: you should eventually print the final attributes of the character.

Here you can find an example of output under all three circumstances:

• • • luna@dhcp-10-68-226-88:-/Downloads	7.#3
<pre>blace) luna@dhcp-10-68-226-88 ~/Downloads\$ (base) luna@dhcp-10-68-226-88 ~/Downloads\$ python ProgrammerType.py You took the red pill and walk into the matrix. Your initial stats are: Health: 100 Speed: 50 Evesight: 50 Gold: 20 You can choose a special item to navigate the matrix. Choose your item: 1. Coffee (+10 speed, -30 health) 2. Python Docs (+10 health, +20 speed) Enter 1, 2, or 3: 1 Four enterset. Four enterset.</pre>	
You walk deeper into the matrix You're becoming tired, but Morpheus appears and offers you another coffee! (and a fruit too). You gain 100 golo	d !
Final character stats: Health: 70 Speed: 70 Eyesight: 50 Gold: 220 Kbase) luna@dhcp-10-68-226-88 ~/Downloads\$	

Figure 1: Special item: coffee

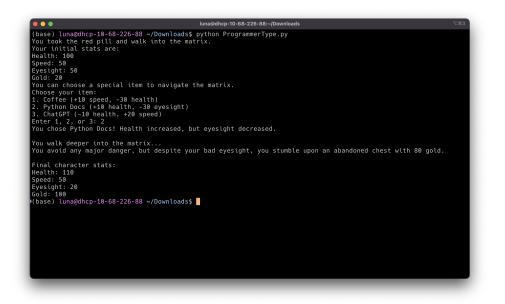


Figure 2: Special item: Python Docs

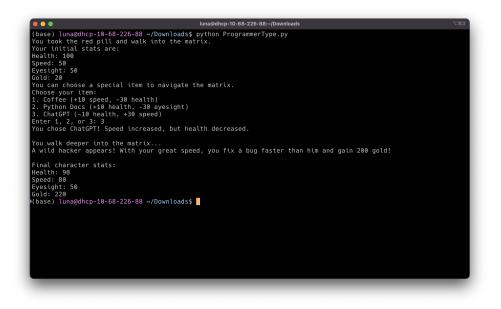


Figure 3: Special item: ChatGPT

You can find a skeleton of the program in the ProgrammerType2.py file. Hint: you should use if-{elif }*-else clauses

Bonus:

1. Feeling like testing yourself?

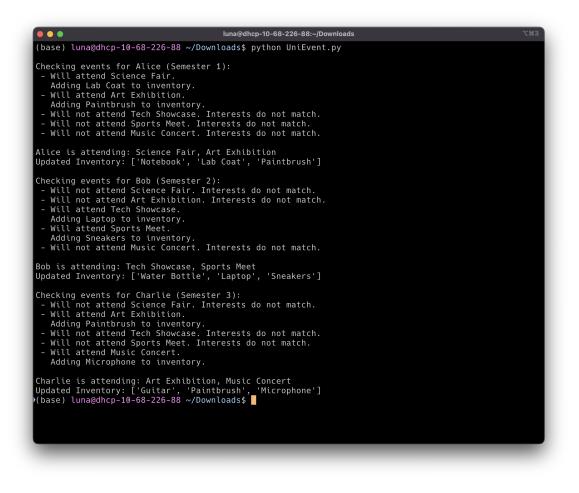
The file UniEvent2.py contains a difficult bonus exercise for you.

This time, you are a Uni Club Event Organiser! You are in charge of organising different events for a University club. The club has several members, each with their own interests. There are multiple events happening, and each event matches certain interests. Your task is to match members to events they would enjoy, update their attendance records, and manage their inventories (like equipment they need for events).

- **Members**: each member is represented as a dictionary. They have a name, semester, a set of interests, and an inventory (a list of items).
- **Events**: events are stored in a dictionary, where each event has a name and is associated with one or more interests, as well as any required items for that event.
- Event assignment: based on each member's interests, they will be matched to events that suit them. If the member doesn't already have the required item in their inventory, the item will be added.
- Final task: print the updated details of each member, including which events they are attending and their updated inventory.

You can find an example of output here:

(2 P)



Important: your submission should contain:

- One .py OR .ipynb file of your solutions to the mandatory exercises
- One **.py** OR **.ipynb** of your solutions to the bonus exercises (**only** if you chose to do them)