Exercise session 10

Introduction to Python. Built-in data types. Variables, lists, tuples, dictionaries, sets. Control structures. Functions. Docstrings.

Advanced Programming - SISSA, UniTS, 2024-2025

Pasquale Claudio Africa

03 Dec 2024

Exercise 1: numerical types, strings

1. What is 5 to the power of 5?

2. What is the remainder from dividing 73 by 6?

3. How many times does the whole number 3 go into 123? What is the remainder of dividing 123 by 3?

4. Split the following string into a list by splitting on the `#` character:

   ```
   s = "apple#banana#cherry#orange"
   ```

5. Use string methods to extract the website domain from an email, e.g., from the string `"pafrica@fakemail.com"`, you should extract `"fakemail"`.

6. Given the following variables:

   ```
   thing = "light"
   speed = 299792458 # m/s
   ```

   Use f-strings to print `"The speed of light is 2.997925e+08 m/s."`

Exercise 1: lists, dictionaries, tuples

7. Given this nested list, use indexing to grab the word `"AdvProg"`:

```
l = [10, [3, 4], [5, [100, 200, ["AdvProg"]], 23, 11], 1, 7]
```

8. Given this nested dictionary grab the word `"AdvProg"`:

```
d = {
    "outer": [
        1,
        2,
        3,
        {"inner": ["this", "is", "inception", {"inner_inner": [1, 2, 3, "AdvProg"]}]},
    ]
}
```

9. Why does the following cell return an error?

```
t = (1, 2, 3, 4, 5)
t[-1] = 6
```

Exercise 1: `if-else`

10. Given the variable `language` which contains a string, use `if-elif-else` to write a program that prints

   - `"I love snakes!"` if `language` is `"Python"` (any kind of capitalization).
   - `"Are you a pirate?"` if `language` is `"C++"` (any kind of capitalization).
   - `"What is `language`?"` if `language` is anything else.

Exercise 2: functions, comprehension

1. Create a function `website()` that grabs the website domain from a URL string. For example, if your function is passed `"www.google.com"`, it should return `"google"`.

2. Create a function `divisible(a, b)` that accepts two integers (`a` and `b`) and returns `True` if `a` is divisble by `b` without a remainder. For example, `divisible(10, 3)` should return `False`, while `divisible(6, 3)` should return `True`.

3. Use list comprehension to square every number in the following list of numbers: `l = [1, 2, 3, 4, 5, 6, 7, 8, 9]`.

4. For the following list of names, write a list comprehension that creates a list of *only* words that start with a capital letter. **Hint**: `str.isupper()`.

```
names = ['Steve Irwin', 'koala', 'kangaroo', 'Australia', 'Sydney', 'desert']
```

Exercise 2: comprehension, exceptions

5. For the following list of `keys` and `vals` use dictionary comprehension to create a dictionary of the form `{'key-0': 0, 'key-1': 1, etc}`. **Hint**: `zip()` can help you combine two lists into on object to be used for comprehension/looping).

```python
keys = [f"key-{k}" for k in range(10)]
vals = range(10)
```

6. Write a `try` / `except` to catch the error generated from the following code and print "I caught you!". Make sure you catch the specific error being caused, this is typically better practice than just catching all errors!

```python
5 / 0
```

Exercise 2: generators, loops, comprehension

7. This question is a little harder. Create a generator function called `listgen(n)` that yields numbers from 0 to n, in batches of lists of maximum 10 numbers at a time. For example, your function should behave as follows:

```python
g = listgen(100)
next(g)
[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]
next(g)
[10, 11, 12, 13, 14, 15, 16, 17, 18, 19]
next(g)
[20, 21, 22, 23, 24, 25, 26, 27, 28, 29]
```

8. Given this 3x4 matrix implemented as a list of lists, write a list comprehension that creates its transpose:

```python
matrix = [[1, 2, 3, 4], [5, 6, 7, 8], [9, 10, 11, 12]]
```

Exercise 2: functions

9. Create a function `lucky_sum(*args)` that takes all the integers a user enters and returns their sum. However, if one of the values is 13 then it does not count towards the sum, nor do any values to its right. For example, your function should behave as follows:

```
lucky_sum(1, 2, 3, 4)
10

lucky_sum(1, 13, 3, 4)
1

lucky_sum(13)
0
```

This example is inspired by the related CodingBat challenge.

Exercise 2: loops, comprehension

10. See below code. Print only the EEG signal and events separately from this nested list using (1) for loop, (2) list comprehension.

```python
two_channel_eeg_signal1 = [8, 9]
event1 = 1

two_channel_eeg_signal2 = [3, 3]
event2 = 2

two_channel_eeg_signal3 = [2, 3]
event3 = 2

nested_list = []
nested_list.append([two_channel_eeg_signal1, event1])
nested_list.append([two_channel_eeg_signal2, event2])
nested_list.append([two_channel_eeg_signal3, event3])
print("EEG signal: ", some_nested_list)
```

Exercise 3: sets, functions

Let A and B be sets. The set $(A \setminus B) \cup (B \setminus A)$ is called the symmetric difference of the two sets.

1. Write a function that performs this operation.

2. Use docstrings to document the function and check the documentation invoking the `help()` function.

3. Compare your results to the output of the command:

```
A.symmetric_difference(B)
```