

# Python Exercise Week 3: conditional statements, range, loops

---

(modified from Mayo Röttger and Andrea Schrader)

Programs should be written in Python according to the respective exercise description. The program must be correct in terms of syntax and semantic. If there exists a minimal solution using only a single pre-defined function, this function is not allowed. The weekly exercise should be uploaded on ILIAS as a single Jupyter Notebook (.ipynb). The assignment will be provided after the lecture on Wednesday and must be uploaded by Tuesday 8:00. Make sure that answers to questions are contained within the Jupyter Notebook that is uploaded on ILIAS. Use commented code when possible

---

## Exercise 1:

**1a.** Create a function that given 2 user-defined values (`threshold_1` and `threshold_2`), returns all the species that are between the two values. The function should work regardless of whether `threshold_1 > threshold_2` there are multiple ways to solve this!

Start by setting these list variables where the first list contains the name of the species and the second its size:

```
species_names = ["Elephant", "Zebra", "Lion", "Giraffe", "Rhinoceros"]
population_sizes = [500, 1200, 250, 800, 80]
```

Remember to use string formatting when possible and break down the code in simple functions. Hint: the index of `species_names` for an animal, is the same as its index in `population_size`!

Test it with:

```
threshold_1 = 300
threshold_2 = 1000
```

and

```
threshold_1 = 2000
threshold_2 = 600
```

Example input:

```
threshold_1 = 100  
thredhold_2 = 400
```

Expected Output:

```
The species between 100 and 400 individuals are:  
Lion  
Rhinoceros
```

**1b.** Run the function defined above but get the name of the animals in every 500 interval (0-500, 500-1000, 1000-1500, 1500-2000). Print a different message if no species is within that *range*

---

## Exercise 2:

Given the DNA sequence

```
plasmid_DNA =  
'ATGCGCTAGCTAGGCTATTATGATGAAGGATAATGACGCATGCACCCATGACTGGACTTGCAGGGCAATACGG  
CAGCGAC'
```

**2a.** Count the occurrences of Cs and Gs in the sequence (GC content) using a for-loop (without using the `str.count` function).

Expected output:

```
The sequence contains 41 Cs and Gs out of 80 nucleotides.  
Resulting in an GC ratio of 51,25%
```

**2b.** Write the function using a while loop.

**2c.** Using the for loop, count the occurrences of the substring **AT**

---

## Exercise 3:

**3a.** Given 2 list of DNA sequences **DNA\_list\_1** and **DNA\_list\_2**. Write a function that compares each sequence from **DNA\_list\_2** to each sequence in **DNA\_list\_1**. If a sequence from **DNA\_list\_2** is found in a sequence from **DNA\_list\_1** print a message.

The sequence ... is found in ... where the dots stand for the sequence

**3b** If the sequence is not found, look for its reverse complement using the function from Exercise 3 from Week 2. If it's found then, print it on screen.

```
seq_list_1 = ["ATGCGT", "CGTAGG", "TTGCAA"]  
seq_list_2 = ["CG", "CAA", "TAG"]
```

---

#### Exercise 4:

**4a.** Simulate growth dynamics of bacteria over a month. Starting from a population of 10, they double in size every day until they reach 1000 individuals. For each day, print how many bacteria are present.

Expected output:

```
At day 1 of the exponential growth phase, there are 10 bacteria  
At day 2 of the exponential growth phase, there are 20 bacteria  
...
```

**4b.** At that point their growth is limited by the amount of resources and stays stationary for 7 days. After the stationary phase, the depletion of nutrients causes the bacterial community to die out, they lose 10% of the individual every day with a minimum of 10 dead individuals per day. For each day, print how many bacteria are present.

Expected output:

```
At day 1 of the death phase, there are 1000 bacteria  
At day 2 of the death phase, there are 900 bacteria  
...
```

**4c.** How long did it take for the bacteria to go through all the phases? change your code to keep track of how many days have passed and print it at the end.

Expected output:

```
The bacteria went through exponential, stationary and death phase in N  
days
```

#### Exercise 5:

**5a.** Model predator-prey interactions between rabbits and foxes. Starting from the year 2000, the rabbit population grows by 30% each year. Calculate the population over 10 years.

Starting from the initial population: `rabbits = 500`

Expected output: `In 2000, there are 500 rabbits`

**5b.** After 10 years of growing without predator, the authorities decide to add foxes to limit the rabbit growth explosion. To this end, we add the foxes to the equation. Each year, after growing the rabbit population, remove 10 rabbits for each fox due to predation. Calculate the population over 10 the following years (2010-2020).

foxes = 50

**5c.** Let's model the predator population, starting from year 2010. Each year that the rabbit population is higher than the previous one, the foxes benefit and double their population due to ease of access to food and migration. Double the fox population after reducing the number of rabbits. If the rabbit population is lower than the year before, reduce the fox population by 60%. For the next 10 years, print the number of rabbits and foxes (2010-2020).

**\*\*Expected output: In 2000, there are 500 rabbits and 0 foxes \*\***

**5d.** What happens in 2050? Do you think this system is sustainable? Try and change some of the values. What happens if foxes eat 9 rabbit each? what happens if foxes eat 8 each?