

Comments, Variables, Console

Solve the following exercises and upload your solutions to **Moodle** until the specified due date. Make sure to use the **exact filenames** that are specified for each individual exercise. Use the provided **unit tests** to check your scripts before submission (see the slides **Handing in Assignments on Moodle**). Unless explicitly stated otherwise, you can assume correct user input and correct arguments. You are *not allowed* to use any concepts and modules that have not yet been presented in the lecture.

Important Information!

Please try to *exactly match the output* given in the examples (naturally, the input can be different). We are running automated tests to aid in the correction and grading process, and deviations from the specified output lead to a significant organizational overhead, which we cannot handle in the majority of the cases due to the high number of submissions.

For example, if the exercise has an output of
Chairs: XYZ
(where XYZ is some user input), do not write
The number of chairs: XYZ
(additional **The number of** and lowercase **c**) or
Chairs:XYZ
(missing space after the colon).

Feel free to copy the output text from the assignment sheet, and then change it according to the exercise task.

Exercise 1 – Submission: a1_ex1.py

25 Points

Create four variables of data types `bool` (boolean), `int` (integer), `float` (floating point) and `str` (string). You can choose arbitrary variable names and values (to use the unit test, you of course need to use the same values as in the example below). Print the variables (to the console) but with the following additional rules:

- The integer must have a minimum print width of 7, must have leading zeros and should display its sign at the beginning of the line (irrespective whether it is positive or negative).
- The float must have a minimum print width of 8, and the number of decimals (precision) must be set to 4.
- The string must be printed two times next to each other, i.e., if the string is **A**, then **AA** must be printed.

Example output for `boolean = False`, `integer = -48`, `float = 1.5` and `string = pythonisgreat` (the `□` character below represents a space, you do not have to literally print this character):

```
False
-000048
□□□□□□1.5000
pythonisgreatpythonisgreat
```

Exercise 2 – Submission: a1_ex2.py**25 Points**

Read four numbers **a**, **b**, **c** and **d** from the console and convert them to integers. Afterwards, perform the following calculations and print the results (no string formatting required; see the example input and output below for how it must look like):

- The sum of **a**, **b** and **d**
- The product of all four numbers
- The sum of **a** and **c** times the sum of **b** and **d**
- The result of an integer division when dividing **a** by **c**
- The result of a regular division when dividing **a** by **b**
- The remainder of a division (modulo) when dividing **a** by **d**
- a^{-c}
- $\sqrt{d} = d^{\frac{1}{2}}$
- $\frac{c}{3} \cdot \left(a^{b+\frac{c}{2}} - 1\right) + d$

Example input and output:¹

```
a: 20
b: 19
c: 18
d: 17
Sum of a, b and d: 56
Product of all numbers: 116280
The sum of a and c times the sum of b and d: 1368
a divided by c (int): 1
a divided by b (float): 1.0526315789473684
Remainder of a divided by d: 3
a to the power of -c: 3.814697265625e-24
d to the power of 1/2 (square root): 4.123105625617661
Complex equation: 1.610612736e+37
```

¹Green colored text indicates user input from the console.

Exercise 3 – Submission: a1_ex3.py**25 Points**

Write a program that computes and prints several metrics of a cylindrical tank given a user-specified **radius** and **height** (see the example input and output below for how it must look like). Both numbers have to be converted to float and are assumed to be meters. Take 3.14159 as the value for π . The metrics to compute are the following:

- The surface area of the tank (float), which includes both the side and the top and bottom areas.
- The volume of the tank (float)
- The amount of water required to fill the tank, given that the tank needs to be filled up to 90% of its volume (float).
- The amount of paint needed to paint the exterior surface of the tank (float). For every square meter of surface, 0.65 liters of paint are required.
- The number of steel plates needed to construct the tank (int). Each steel plate covers 2square meters of surface. For example, a surface area of 85 square meters requires 43 plates.

All float results must be printed with 2 decimal places.

Example input and output:

```
Radius (in metres): 2.4
Height (in metres): 5.5
Surface area of the tank: 119.13
Volume of the tank: 99.53
Amount of water needed: 89.57
Litres of paint needed: 77.43
Number of plates needed: 60
```

Exercise 4 – Submission: a1_ex4.py**25 Points**

Write a program that can print a small order form for a furniture store (see the example input and output below for how it must look like). You have to read in three integer numbers, which will then be part of such an order:

- The number of ordered chairs. Each chair costs 49.99 Euros.
- The number of ordered tables. Each table costs 199.99 Euros.
- The number of ordered lamps. Each lamp costs 29.99 Euros.

Calculate the total cost for these three positions, and finally, compute the total cost of the entire order. The number of ordered items must have a minimum print width of 3 (4 for the line with the lamps, because it is a letter shorter), for the single price 6 and 10 for the item sum. The total sum should have a print width of 26, so that even for large sums (see example below) the formatting still works. You have to format only the output, not the user input. All float results must be printed with 2 decimal places.

Example input and output:

```
Input the number of ordered chairs: 123
Input the number of ordered tables: 654
Input the number of ordered lamps: 123
```

Order Form:

```
-----
Chairs: 123 x 49.99 = 6148.77
Tables: 654 x 199.99 = 130793.46
Lamps: 123 x 29.99 = 3688.77
-----
Total: 140631.00
=====
```