

Lab 3: input, type conversion & f-strings

Refresher quiz

Fill in the expected results for these expressions

$23 // 3 =$

$23 \% 3 =$

$3 / 2 + 5 =$

$12 / (3 + 1) =$

 Check



1. CIRCLE

Write code to ask the user for the radius of a circle.

The program should display

- The area of the circle $A = \pi r^2$
- The circumference of the circle $C = 2\pi r$.


Display the output to 3 significant digits in suitable sentences using f-strings. `import math` to get an accurate value for pi.

2. TWO DIGIT NUMBER USING OPERATORS (`//` and `%`)

Ask the user for a two-digit number. Display the number of tens and the number of units.

A sample run is:

```
What is the number: 58
58 is made up of 5 tens and 8 units.
```

 Extend your code to ask for three-digit numbers instead and display the number of hundreds, tens, and units.

3. SANDWICH BOARD

Write code to ask the user for the title of a new sandwich (e.g. "The MTU Breakfast Roll") and the cost of the sandwich. Print these details to the screen, suitably formatted.

The shop is currently giving a 20% reduction. Print out a sign saying

" The MTU Breakfast Roll Reduced today by 20% from €xx.xx to €xx.xx"

Note you should place the name of the sandwich into the input() statement using an f-string so whatever the user gives the program as the name of the sandwich will appear in the prompt. Use a named constant for the discount of 20%.

4. AVERAGE GRADE

Ask the user to enter the results for three subjects English, Maths, and Science.

Calculate the average mark they got and display the average as a percentage using an f-string.

Note f-string format specifiers have a datatype for percentages, use the % symbol for the datatype in your format specifier, see how this changes the code.

5. TABLE OF DATA USING F-STRINGS

Create 9 variables to store the following data:

Name	Age	Grade
Minnie Mouse	23	58.5
Daisy Duck	34	67.8

Donald Duck	56	45.5
-------------	----	------

Calling these name1, name2, name3, age1, etc is perfectly fine.

Use f-strings to display the data as a table as follows:

Name	Age	Grade
Minnie Mouse	23	58.50
Daisy Duck	34	67.80
Donald Duck	56	45.50

6. SHIPPING THINGAMAJIGS

A shipping company can fit 1000 thingamajigs into a crate.

Write code to ask the user for the number of thingamajigs.

Display the number of filled crates and the number of leftover thingamajigs as demonstrating in the following sample run, noting the use of :, to display large numbers with commas in the multiples of thousands.

```
How many thingamajigs? 5847
5,847 thingamajigs can be packed into 5, with 847 thingamajigs left unpacked.
```

7. SHOPPING

A person goes to the shop each morning to buy bread and milk. Ask the user how much are a loaf of bread and a litre of milk.

As this is a regular customer they are given a 10% discount. Ask them how many of each item they require and print out the bill.

Sample Output

```
-----
Enter the cost of a litre of milk: 1.50
Enter the cost of a loaf of bread: 2.15
```

```
Enter the no. of litres bought: 3
Enter the no. of loaves bought: 2
```

Your Bill

```
-----
Milk  3      1.50  4.50
Bread 2      2.15  3.30
-----
Total                8.80
Final Cost           7.92
```

8. GROUP EXERCISE 🤸 (please wait for me to initiate this exercise)

Please complete this exercise while working in groups of 2 or 3. Now you have the ability to write python code, perform mathematical operations, get user input and print using f-string notation. In this exercise I want you to use everything you have learned so far to design an exercise for your classmates.

As a group create an exercise that could be used to practise skills in lab, you should also design a grading rubric for your exercise. Write down the exercise and assign 15 marks to the individual tasks in the exercise (considering code functionality and code readability).

Together you will write code to complete the exercise and make sure you are using the grading rubric to help inform your solution.

Use the Inbox feature on Canvas to send me the exercise along with its solution and the grading rubric, including the names of the team members who designed the exercise.

9. FINISHED?

An important part of learning programming is learning the stylistic conventions of the language you are using. The coding conventions of Python are detailed in the PEP-8 Style Guide. PEP stands for Python Enhancement Proposal. See a link to the guide here:

<https://peps.python.org/pep-0008/> ➡ [\(https://peps.python.org/pep-0008/\)](https://peps.python.org/pep-0008/)

In particular note the following sections

- **String Quotes** ➡ [_ \(https://peps.python.org/pep-0008/#string-quotes\)](https://peps.python.org/pep-0008/#string-quotes)
- **Comments** ➡ [_ \(https://peps.python.org/pep-0008/#comments\)](https://peps.python.org/pep-0008/#comments)
- **Function and Variable Names** ➡ [_ \(https://peps.python.org/pep-0008/#function-and-variable-names\)](https://peps.python.org/pep-0008/#function-and-variable-names)