**NOTES- INTRO TO PROGRAMMING-CSE 110**

**April 22nd, 2025**

More info:

* The Basics of Writing a Program!!
  + "PRINT” it means that the words will be displayed on the screen when the program is run

Click here: <https://byui-cse.github.io/cse110-course/lesson01/prepare.html>

Class 2nd week- April 29th, 2027

‘’- characters

“”- Strings

With 3 “”” se pone en otro paragraph y también asi \n

**Ints-** can only hold whole numbers (i.e., no decimals)

**float**- decimal/exponential

**input-** returns a **string** by default- no suma ni nada, se queda asi como lo pusiste

The **%** operator is called the modulo operator, and it returns the remainder of a division operation.

IMPORTANT example

dolls = 5  
  
print("There are " + dolls + " dolls")

* "There are " is a **string**
* dolls is an **integer** (5)
* " dolls" is a **string**

You're trying to **add (concatenate)** a string, an integer, and another string. But in Python:

❗ You **can't concatenate a string and an integer directly** — this will raise a TypeError.

CORRECT ONE

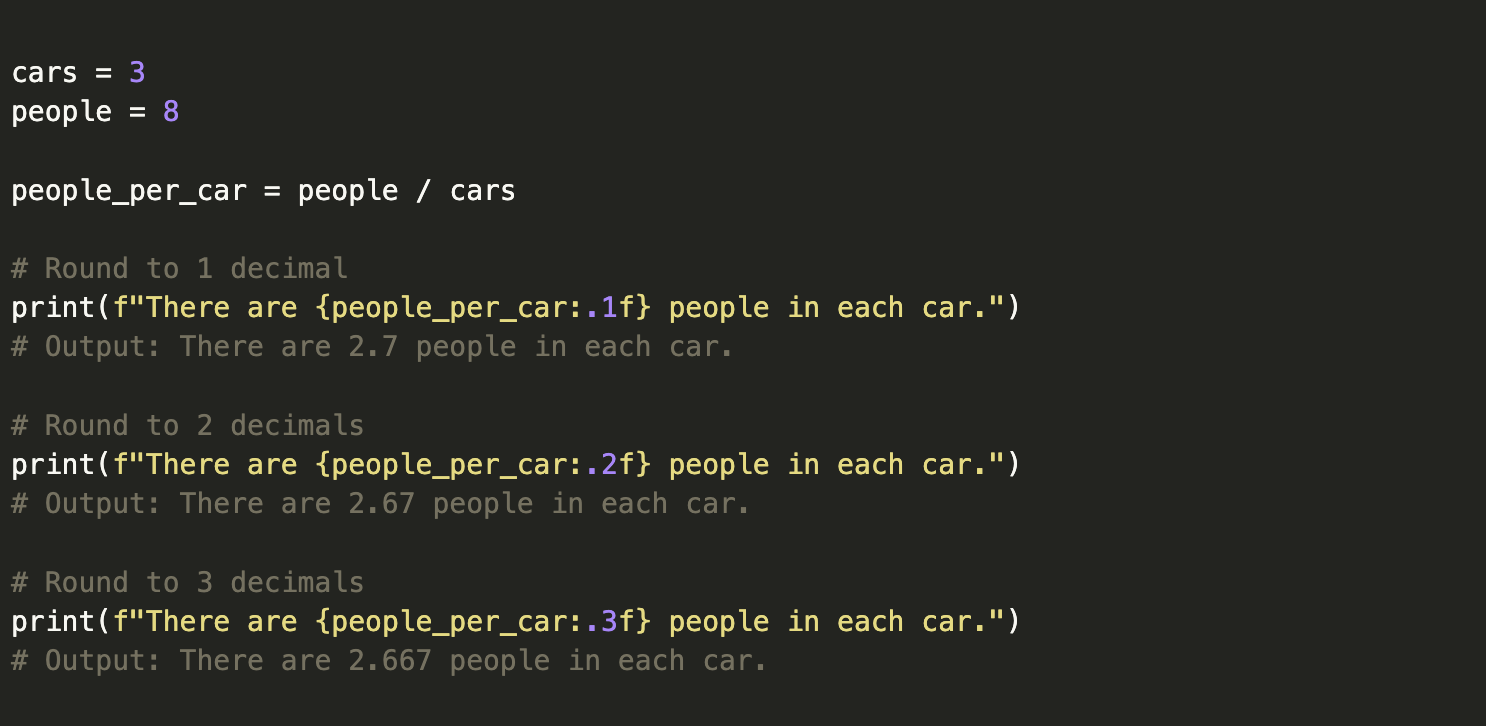
dolls = 5

print("There are " + str(dolls) + " dolls")

**Defining the number of decimals to display**

In a format string, you define the precision, or number of decimals to display, by putting a **:.2f** after the variable name (changing the 2 to whichever amount you'd like).

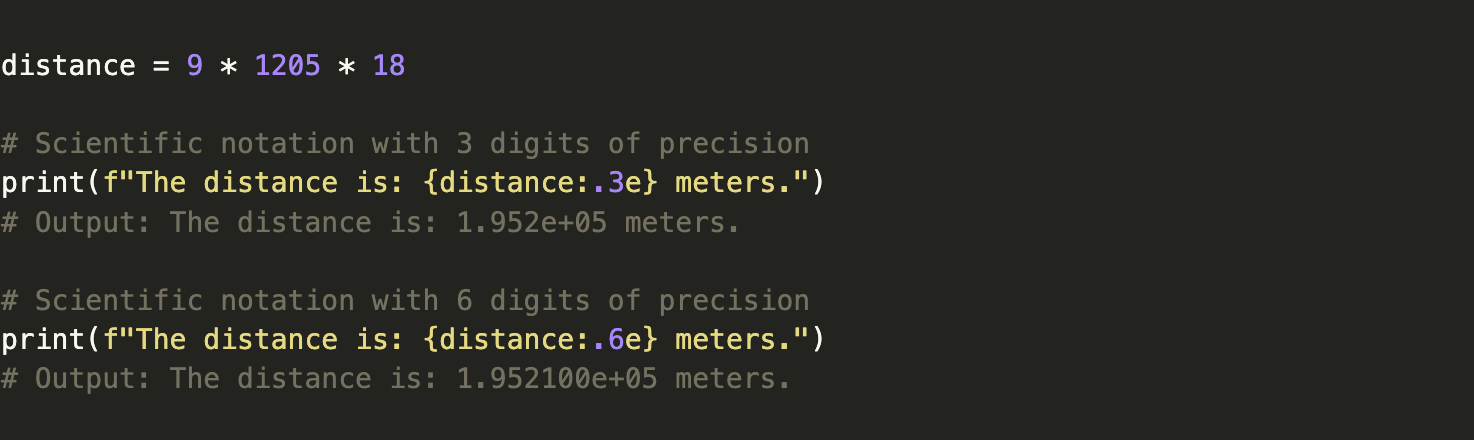
* The colon (**:**) after the variable name indicates that you are going to specify how to format it.
* The period (**.**) indicates that you are setting the precision or number of decimal places.
* The number (in this example **2**) indicates that you would like that number of decimal places to be displayed
* The **f** indicates that you want fixed-point notation.



**Scientific Notation**

You can tell the computer to display the number in scientific notation, or "exponent" notation by using **:.3e** after your variable, where **3** defines the precision and **e** indicates that you are using exponent notation.

The following shows this in action:



**Thousands Grouping**

When you write numbers in code, you don't use commas to separate the groupings of digits (in other words, you don't write: 10,000,000, just 10000000). Recently, Python added a notation that lets you type using underscores such as 10\_000\_000.

In any case, when you want to display large numbers to the user, you may want to display it with commas or underscores. This is done by using either **:,** or **:\_** after the variable name.

The following shows this in action:

A screenshot of a computer program

AI-generated content may be incorrect.

A screenshot of a computer program

AI-generated content may be incorrect.

You can see the list of [Mathematical functions](https://docs.python.org/3/library/math.html)

A few that might be of interest to you are the following:

* **math.ceil(value)**—Rounds **value** up to the next whole number, the "ceiling."
* **math.floor(value)**—Rounds **value** down to the next whole number.
* **math.exp(value)**—Raises e to the power of **value**.
* **math.sin(value)**—Computes the trigonometry sine function of **value** in radians.

IF STATEMENTS

<https://video.byui.edu/media/t/1_kqzill12>

LIST INDEXES: <https://byui-cse.github.io/cse110-course/lesson10/prepare.html>

WORKING WITH FILES: <https://byui-cse.github.io/cse110-course/lesson11/prepare.html>