COMMAND LINE ARGUMENTS:

Command line arguments is another way to read user provided input.

We have to pass arguments from the command prompt at the time of executing python script.

HOW CAN YOU READ COMMAND LINE ARGUMENTS?

**2 ways:**

1. **sys** --🡪predefined module in python

This module contains one variable **argv-🡪This variable present inside sys module.**

**Argv ------🡪list type**

**Purpose:**

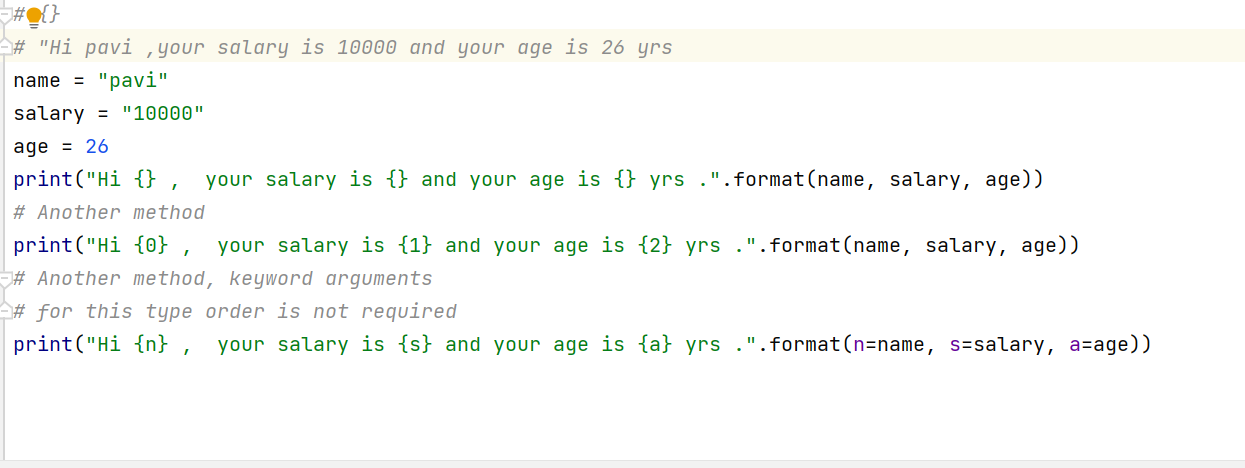
* Main objective of command line argument is we can customize behaviour of the application based on our requirements.

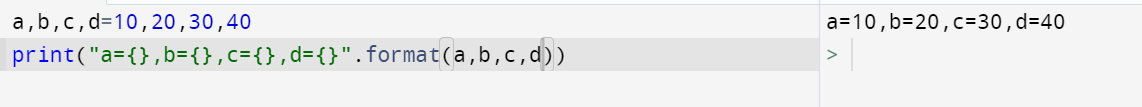
CONCLUSION:

1. Space is the separator between command line arguments, using “” to solve this problem.
2. If you are trying to access command line arguments without range index, we are going to get Index out of error.
3. If you want to add two numbers from command line arguments we should typecast our arguments, because **argv** default return type is string only.

**REPLACEMENT OPERATOR:**

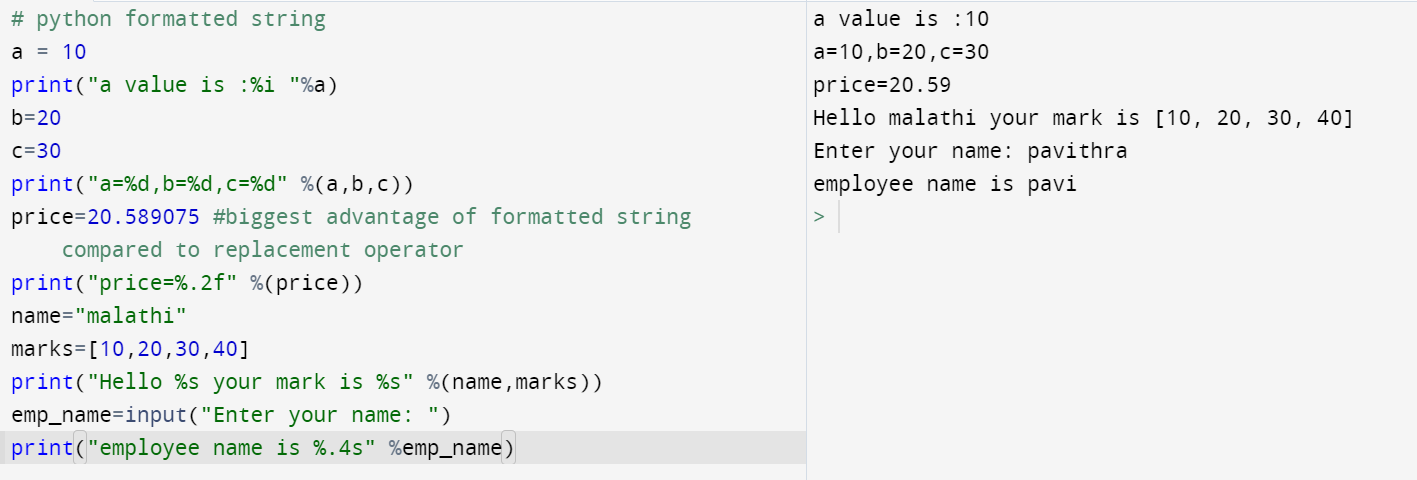
**{} -----🡪 Replacement operator**

****

****

**Print() with formatted string:**

* **% i ------🡪signed decimal value**
* **%d -----🡪signed decimal value**
* **%f ------🡪float value**
* **%s ------🡪string, any other objects like list, tuple etc….**
* **Formatted string is more powerful than string replacement operator.**
* **Customization is possible with formatted string.**
* **Syntax: print(“formatted string” %(variable name))**

****