Input And Output:

- Till now we have execute examples by hard coding variable values.
- But we can pass the values at run time also

```
demo.py

age=16
print(age)

output
16
```

- But in real time based on requirement few values we should take at run time or dynamically.

input() function:

- input() is a predefined function.
- This function accept input from the keyboard.
- This function takes a value from keyboard and returns it as a string type
- Based on requirement we can convert from string to other types.

```
demo.py

name = input("Enter the name: ")
age = input(“Enter the age: ")

print("You entered name as: ", name)
print("You entered age as: ", age)

python demo.py

Output

Enter the name: Sachin
Enter the age: 40

You entered name as: Sachin
You entered age as: 40
```

Command line arguments:

- While running the program we can provide the arguments in command prompt.
- These provided arguments are called as Command line arguments.
- We need to use argv to work with command line arguments
- argv is variable and it is list type in python.

- argv is available sys module
- So, we need to import sys module to access argv.
- If we are trying to access command line arguments with out of range index, then we will get IndexError.
- **syntax:**

```
from sys import argv


>py demo.py 10 20

argv[0] => demo.py (Name of the program )
argv[1] => 10
argv[2] => 20
```

- argv is a list type.
- Command line arguments are string type
- So, by default every element presents in argv is string type.
- Based on requirement we can convert from string type to another type using converion methods: int(), bool(), list(), set(), tuple(), dict() etc.