

## Artificial Intelligence / Machine Learning Bootcamp

### Python Functions

## What is a Function in Python?

A function in Python is a block of reusable code that performs a specific task when called. It helps break programs into smaller, manageable pieces and avoids repeating code.

## Why use functions?

Organize code better

Avoid repetition (write once, use many times)

Make debugging easier

Enable reusability and modularity

---

### 1. `print()`

Used to display output.

```
print("Hello, World!")
```

**Explanation:** This prints the string "Hello, World!" on the screen.

---

### 2. `type()`

Returns the data type of a variable.

```
x = 5  
print(type(x)) # Output: <class 'int'>
```

**Explanation:** It shows that `x` is of type integer.

---

### 3. `len()`

Returns the number of items in an object like a string, list, or tuple.

```
name = "Python"  
print(len(name)) # Output: 6
```

**Explanation:** Counts the number of characters in the string "Python".

---

### 4. `input()`

Takes input from the user.

```
name = input("Enter your name: ")  
print("Hello,", name)
```

**Explanation:** It waits for the user to type something, then prints it with "Hello".

---

### 5. `int()` / `float()` / `str()`

Convert values into integer, float, or string types.

```
a = "5"
print(int(a)) # Converts string to integer: 5
print(float(a)) # Converts string to float: 5.0
print(str(5)) # Converts number to string: "5"
```

**Explanation:** Helps change data types for calculation or display.

---

## 6. `max()` / `min()`

Returns the maximum or minimum value from a list or tuple.

```
numbers = [4, 7, 2, 9]
print(max(numbers)) # 9
print(min(numbers)) # 2
```

**Explanation:** Useful to find highest or lowest numbers.

---

## 7. `sum()`

Adds all items in a list or tuple.

```
marks = [80, 90, 70]
print(sum(marks)) # Output: 240
```

**Explanation:** Calculates the total of all numbers.

---

## 8. `range()`

Generates a sequence of numbers.

```
for i in range(3):
    print(i)
```

**Explanation:** Prints numbers from 0 to 2 (3 is not included).

---

## 9. `list()` / `tuple()` / `set()`

Converts items into respective data structures.

```
a = "abc"
print(list(a)) # ['a', 'b', 'c']
print(tuple(a)) # ('a', 'b', 'c')
print(set(a)) # {'a', 'b', 'c'}
```

**Explanation:** Changes a string into different types like list, tuple, or set.

---

## 10. `sorted()`

Returns a sorted version of a list or tuple.

```
nums = [3, 1, 4]
print(sorted(nums)) # [1, 3, 4]
```

**Explanation:** Does not change the original list; it returns a new sorted one.

---

## 11. `reversed()`

Returns an iterator that accesses the elements in reverse order.

```
nums = [1, 2, 3]
print(list(reversed(nums))) # [3, 2, 1]
```

**Explanation:** Helps to reverse the order of items.

---

## 12. `enumerate()`

Returns both index and value in a loop.

```
fruits = ['apple', 'banana']
for index, fruit in enumerate(fruits):
    print(index, fruit)
```

**Explanation:** Useful when you want to know the position of each item.

---

### 13. `zip()`

Combines two or more iterables together.

```
names = ['A', 'B']
scores = [90, 80]
for name, score in zip(names, scores):
    print(name, score)
```

**Explanation:** Pairs items from multiple lists together.

---

### 14. `abs()`

Returns the absolute (positive) value.

```
print(abs(-5)) # Output: 5
```

**Explanation:** Removes the negative sign.

---

### 15. `round()`

Rounds a number to nearest integer or decimal places.

```
print(round(4.56)) # Output: 5
print(round(4.567, 2)) # Output: 4.57
```

**Explanation:** Useful for formatting decimal values.

---

### 16. `help()`

Shows help documentation for any function.

```
help(len)
```

**Explanation:** Opens a detailed guide for `len()`.

---

### 17. `id()`

Returns the memory location of an object.

```
a = 10
print(id(a))
```

**Explanation:** Useful to understand how Python stores variables.

---

### 18. `isinstance()`

Checks if a value is of a specific data type.

```
x = 10
print(isinstance(x, int)) # Output: True
```

**Explanation:** Helps in type-checking during coding.

---

#### Connect @

Mail (Sanjana): [sanjanadevibihana@gmail.com](mailto:sanjanadevibihana@gmail.com)

Contact: [+91-6283762268](tel:+91-6283762268)

**SANJANA DEVI**

AI/ML Student | Intern @ CodroidHub Private Limited | First-Year Engineering Student.

**SANJANA DEVI**