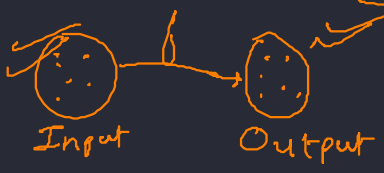
Functions in Python

In math, a function is a relationship that maps inputs to outputs. Similarly, in Python or in any programming language we have functions. It is a self-contained block of code that takes inputs and produces outputs.



Why Use Functions?

1. ABSTRACTION: Functions hide complexity. You don't need to know how len() works internally, you just use it. When you create functions, you're building a vocabulary for your code with descriptive names instead of confusing statement sequences.

2. ENCAPSULATION: Functions create their own namespace. Variables inside a function stay inside and no accidental interference occurs with the rest of your code.

3. Modularity: You organize your code in functions or modules and use it whenever required.

4. REUSABILITY: Without functions, you will copy-paste code everywhere. But with functions, change once and benefit everywhere.
This is the DRY principle: Don't Repeat Yourself!

5. MAINTAINABILITY: Functions make code easier to understand, debug, and enhance. You can test each piece independently and document clearly.

6. TESTABILITY: Functions with clear inputs and outputs are easy to test. You can verify each function works correctly before combining them.