@ score = 95

→ My_var = "Hello"
→ score = 95.

* Data types :-

A data type defines the kind of value a variable can hold and the operations that can be performed on it. Since python is a dynamically typed language, the data type of a variable is determined automatically at runtime based on the assigned value.

1. standard (Built-in) Data types

Python provides several built-in data types, categorized as follows:

1. Numeric Types
   - int → Integer numbers (e.g., 10, -25)
   - float → Decimal numbers (e.g., 3.14, -2.5)
   - complex → complex numbers (e.g., 3+4j)
2. Text type
   - str → string (sequence of characters, e.g., "Hello")
3. sequence Types
   - list → ordered, mutable collection (e.g., [1,2,3])
   - tuple → ordered, immutable collection (e.g., (1,2,3))
   - range → sequence of numbers (e.g., range(5))
4. set types
   - set → unordered, unique collection (e.g., {1,2,3})
   - frozenset → Immutable version of set
5. Mapping Type
   - dict → key-value pairs (eg: {"name": "Alice", "age":25})

6. Boolean Type
   - bool -> logical values True or False.

7. Binary Types
   - bytes, bytearray, memoryview used for handling binary data

*Input and output functions :-

Input and output are handled primarily using the built-in functions input() and print()

1. Input function (input())
   - The input() function is used to take input from the user as a string.
   - If numeric input is required, the string must be typecast int or float.

Syntax :-

```
variable = input ("prompt Message:")
```

Ex:-

```
name = input ("Enter your name:"))
age = int(input (" Enter your age:"))
print (f "Hello {name}, you are {age} years old.")
```

2. output function (print())
   - The print() function is used to display information to the console.
   - Supports multiple arguments, string formatting, and special parameters like sep and end.

syntax :-

```
print (value1, value2, ..., sep = ' ', end = '\n')
```

ex:-

```
x, y = 10, 20
print ("the sum is", x+y)
```