Variable & Data types:

A variable is like a container that stores data. It has a name (identifier) and holds a vlaue.

Rules for variables:

- Must start with a letter or underscore (_), not a number.
- Can contain letters, numbers, and underscores.
- Are case-sensitive (Age ≠ age).
- No reserved keywords (for, class, etc.) can be used.

In [ ]:
```
x = 10          # integer
name = "Barsa"  # string
pi = 3.14       # float
is_active = True # boolean
```

In [2]:
```
x=10
print(x)

name="Barsa"
print(name)

pi=3.14
print(pi)

is_active=True
print(is_active)
```

```
10
Barsa
3.14
True
```

Data Types:

A data type tells Python what kind of value is stored in a variable.

1.Numeric Types

- int → Whole numbers (e.g., 10, -5)
- float → Decimal numbers (e.g., 3.14, -0.5)
- complex → Complex numbers (e.g., 2 + 3j)

2.Text Type

- str → Strings (e.g., "Hello", 'Python')
- Boolean Type
- bool → True or False

3.Sequence Types

- list → Ordered, changeable (e.g., [1, 2, 3])
- tuple → Ordered, unchangeable (e.g., (1, 2, 3))
- range → Sequence of numbers (e.g., range(5))

4.Set Types

- set → Unordered, unique values (e.g., {1, 2, 3})
- frozenset → Immutable set

5.Mapping Type

- dict → Key-value pairs (e.g., {"name": "Barsa", "age": 21})

6.None Type

- None → Represents no value (e.g., x = None)

In [5]:
```
x = 100          # int
y = 12.5         # float
z = "Python"     # str
a = [1, 2, 3]    # list
b = (4, 5, 6)    # tuple
c = {"age": 20}  # dict
d = True         # bool
e = None         # NoneType
```

In [6]:
```
# Checking type of variables:

print(type(x))  # Output: <class 'int'>
print(type(y))  # Output: <class 'float'>
print(type(z))  # Output: <class 'str'>
print(type(a))  # Output: <class 'list'>
print(type(b))  # Output: <class 'tuple'>
print(type(c))  # Output: <class 'dict'>
print(type(d))  # Output: <class 'bool'>
print(type(e))  # Output: <class 'NoneType'>
```

```
<class 'int'>
<class 'float'>
<class 'str'>
<class 'list'>
<class 'tuple'>
<class 'dict'>
<class 'bool'>
<class 'NoneType'>
```

In [ ]: