

Day 4 – Variables, Data Types & Operators(Python Basics for AI Engineer)

==> Why Day 4 is EXTREMELY Important

Before AI, ML, or GenAI models:

👉 **We must know how to store data and operate on it**

Everything in AI depends on:

- Variables → store data
- Data Types → define data nature
- Operators → perform logic & math

✗ No variables → no data

✗ No data types → errors

✗ No operators → no computation

This is the **foundation of all programming**.

4.1. Variable

==> 2. What is a Variable?

Definition ==>

A **variable** is a name given to a memory location that stores data.

-- > Simple words:

Variable = container (box) that holds value

== > Real-Life Example

- Name = "Ramakant"
- Age = 22
- Salary = 50000

Just like:

- A bottle holds water
- A variable holds data

==> Syntax (How to Create Variable)

```
variable_name = value
```

==> Example

```
name = "Ramakant"
```

```
age = 22
```

```
height = 5.9
```

==> Rules for Defining Variables

--> Allowed

- Letters (a–z, A–Z)
- Numbers (after letters)
- Underscore (_)

-- > Not Allowed

- Starting with number
- Special symbols (@, #, \$)
- Python keywords

== > Invalid Examples

```
1name = "Amit"    # ✗
```

```
my-name = "Amit" # ✗  
for = 10        # ✗ keyword
```

== > Valid Examples

```
student_name = "Amit"  
student1 = 101  
_age = 22
```

== > Why Variables Are Needed

- Store user input
- Store model data
- Store predictions
- Store configuration values

-- > Every AI program uses **thousands of variables**

4.2. Data Types

=> What are Data Types?

--> Definition

Data types define the type of value stored in a variable.

Python needs to know:

- Is it number?
- Is it text?
- Is it collection?

--> Built-in Data Types in Python

1 Integer (int)

Whole numbers

```
x = 10
```

```
y = -5
```

Use case:

- Age
- Count
- Quantity

2 Float (float)

Decimal numbers

```
price = 99.99
```

```
accuracy = 0.89
```

Use case:

- ML accuracy
- Loss values
- Measurements

3 String (str)

Text data

```
name = "Python"
```

```
message = "AI Engineer"
```

Use case:

- NLP
- Chatbots
- User input

4 Boolean (bool)

True or False

```
is_active = True
```

```
is_logged_in = False
```

Use case:

- Conditions
- Model decisions

--> Check Data Type

```
x = 10
```

```
print(type(x))
```

Out : int

--> Collection Data Types

5 List (list)

Ordered & changeable

```
marks = [80, 85, 90]
```

Use case:

- Dataset rows
- ML features

6 Tuple (tuple)

Ordered & immutable

```
coordinates = (10, 20)
```

Use case:

- Fixed values
- Model configurations

7 Set (set)

Unordered & unique

```
unique_ids = {1, 2, 3}
```

Use case:

- Remove duplicates
- Feature uniqueness

8 Dictionary (dict)

Key-value pairs

```
student = {  
    "name": "Amit",  
    "age": 21,  
    "marks": 85  
}
```

Use case:

- JSON data
- API responses
- AI model inputs

7. Type Conversion (Very Common in AI)

```
age = "22"
```

```
age = int(age)
```

```
Out: 22
```

-- > Required when:

- Reading user input
- Loading CSV data

4.3 Operators

--> What are Operators?

Definition

Operators perform operations on variables and values.

Without operators:

- ✗ No calculation
- ✗ No comparison
- ✗ No logic

--> Arithmetic Operators

```
a = 10
b = 3
print(a + b) # Addition
13
print(a - b) # Subtraction
7
print(a * b) # Multiplication
30
print(a / b) # Division
3.33
print(a // b) # Floor division
3
print(a % b) # Modulus
1
print(a ** b) # Power
1000
```

--> AI Use:

- Loss calculation
- Feature scaling
- Optimization

--> Assignment Operators

```
x = 5
x += 2 # x = x + 2
```

Used in:

- Loops

- Counters
- Training steps

--> Comparison Operators

a = 10

b = 5

a > b

a == b

a != b

AI Use:

- Threshold checks
- Accuracy evaluation

--> Logical Operators

(age > 18) and (is_logged_in == True)

AI Use:

- Decision making
- Rule-based systems

--> Membership Operators

"a" in "data"

Used in:

- NLP
- Text search