জাভাতে **ডেটা টাইপ (Data Type)** নির্দেশ করে একটি ভেরিয়েবল বা এক্সপ্রেশন কী ধরনের ডেটা স্টোর বা রিটার্ন করতে পারে। জাভাতে ডেটা টাইপ দুই প্রকার:

1. **প্রিমিটিভ ডেটা টাইপ (Primitive Data Types)**
2. **নন-প্রিমিটিভ ডেটা টাইপ (Non-Primitive Data Types)**

নিচে এই ডেটা টাইপগুলো বিস্তারিতভাবে বাংলায় ব্যাখ্যা করা হলো:

**১. প্রিমিটিভ ডেটা টাইপ (Primitive Data Types)**

প্রিমিটিভ ডেটা টাইপ হল জাভার বিল্ট-ইন ডেটা টাইপ, যা সরাসরি ডেটা স্টোর করে। এগুলি মেমোরিতে কম জায়গা নেয় এবং দ্রুত এক্সিকিউট হয়। জাভাতে ৮টি প্রিমিটিভ ডেটা টাইপ রয়েছে:

| **ডেটা টাইপ** | **সাইজ** | **ডেসক্রিপশন** | **উদাহরণ** |
| --- | --- | --- | --- |
| byte | 1 byte | ছোট পূর্ণ সংখ্যা (-128 থেকে 127) | byte b = 100; |
| short | 2 bytes | মাঝারি পূর্ণ সংখ্যা | short s = 5000; |
| int | 4 bytes | পূর্ণ সংখ্যা (সবচেয়ে বেশি ব্যবহৃত) | int i = 100000; |
| long | 8 bytes | বড় পূর্ণ সংখ্যা | long l = 100000L; |
| float | 4 bytes | দশমিক সংখ্যা (সিঙ্গেল প্রিসিশন) | float f = 3.14f; |
| double | 8 bytes | দশমিক সংখ্যা (ডাবল প্রিসিশন) | double d = 3.14159; |
| char | 2 bytes | একটি ইউনিকোড ক্যারেক্টার | char c = 'A'; |
| boolean | 1 bit | true বা false ভ্যালু | boolean b = true; |

**উদাহরণ:**

java

Copy

int age = 25; // পূর্ণ সংখ্যা

double salary = 50000.75; // দশমিক সংখ্যা

char grade = 'A'; // ক্যারেক্টার

boolean isJavaFun = true; // বুলিয়ান

**২. নন-প্রিমিটিভ ডেটা টাইপ (Non-Primitive Data Types)**

নন-প্রিমিটিভ ডেটা টাইপগুলি রেফারেন্স টাইপ, অর্থাৎ এগুলি অবজেক্টকে রেফার করে। এগুলির ডিফল্ট ভ্যালু null হয়। নন-প্রিমিটিভ ডেটা টাইপের উদাহরণ:

1. **স্ট্রিং (String)**: টেক্সট ডেটা স্টোর করার জন্য।

java

Copy

String name = "Java";

1. **অ্যারে (Array)**: একই টাইপের একাধিক ভ্যালু স্টোর করার জন্য।

java

Copy

int[] numbers = {1, 2, 3, 4, 5};

1. **ক্লাস (Class)**: ইউজার-ডিফাইন্ড ডেটা টাইপ।

java

Copy

class Student {

String name;

int age;

}

1. **ইন্টারফেস (Interface)**: ক্লাসের জন্য কন্ট্র্যাক্ট ডিফাইন করে।

java

Copy

interface Animal {

void sound();

}

**প্রিমিটিভ vs নন-প্রিমিটিভ ডেটা টাইপ:**

| **বৈশিষ্ট্য** | **প্রিমিটিভ ডেটা টাইপ** | **নন-প্রিমিটিভ ডেটা টাইপ** |
| --- | --- | --- |
| **ডিফিনিশন** | বিল্ট-ইন ডেটা টাইপ | ইউজার-ডিফাইন্ড ডেটা টাইপ |
| **মেমোরি ব্যবহার** | কম মেমোরি ব্যবহার করে | বেশি মেমোরি ব্যবহার করে |
| **ডিফল্ট ভ্যালু** | ডিফল্ট ভ্যালু থাকে (যেমন: int এর জন্য 0) | ডিফল্ট ভ্যালু null |
| **স্টোরেজ** | সরাসরি ডেটা স্টোর করে | অবজেক্টকে রেফার করে |
| **উদাহরণ** | int, char, boolean | String, Array, Class |

**ডেটা টাইপের ব্যবহার:**

ডেটা টাইপ ব্যবহার করে ভেরিয়েবল ডিক্লার করা হয় এবং প্রোগ্রামে ডেটা ম্যানিপুলেট করা হয়। নিচে একটি উদাহরণ দেওয়া হলো:

java

Copy

public class Main {

public static void main(String[] args) {

int number = 10; // প্রিমিটিভ ডেটা টাইপ

String message = "Hello"; // নন-প্রিমিটিভ ডেটা টাইপ

System.out.println(number); // আউটপুট: 10

System.out.println(message); // আউটপুট: Hello

}

}

**ডেটা টাইপ কনভার্শন (Type Conversion):**

জাভাতে এক ডেটা টাইপ থেকে অন্য ডেটা টাইপে কনভার্শন করা যায়। এটি দুইভাবে করা যায়:

1. **ইমপ্লিসিট টাইপ কাস্টিং (Implicit Type Casting)**: ছোট ডেটা টাইপ থেকে বড় ডেটা টাইপে অটোমেটিক কনভার্শন।

java

Copy

int a = 10;

double b = a; // ইমপ্লিসিট কাস্টিং (int থেকে double)

1. **এক্সপ্লিসিট টাইপ কাস্টিং (Explicit Type Casting)**: বড় ডেটা টাইপ থেকে ছোট ডেটা টাইপে ম্যানুয়াল কনভার্শন।

java

Copy

double a = 10.5;

int b = (int) a; // এক্সপ্লিসিট কাস্টিং (double থেকে int)

**ডেটা টাইপের গুরুত্ব:**

1. ডেটা টাইপ নির্দেশ করে একটি ভেরিয়েবল কী ধরনের ডেটা স্টোর করতে পারে।
2. এটি মেমোরি ম্যানেজমেন্টে সাহায্য করে।
3. ডেটা টাইপ ব্যবহার করে প্রোগ্রামের এফিসিয়েন্সি বাড়ানো যায়।

জাভার ডেটা টাইপ শেখা খুবই গুরুত্বপূর্ণ, কারণ এটি প্রোগ্রামিং এর বেসিক কনসেপ্ট। নিয়মিত অনুশীলন করলে ডেটা টাইপের ব্যবহার সহজেই আয়ত্ত করা যায়। 😊