**Give me some idea on what is a Data type. What are Primitive Data types and Non-Primitive Data types. Explain with some examples**.

**ANS:**

A data type is a classification that specifies which type of value a variable can hold in a programming language. It defines the operations that can be performed on the data, as well as the way it is stored in memory.

**1)Primitive Data Types:**

Primitive data types are the most basic types of data that are directly supported by a programming language. They are not made up of other data types and are typically mapped directly to machine-level instructions. These data types serve as the foundation for storing values in programming.

**some common primitive data types with examples:**

**1)Integer (int):**

Represents whole numbers.

Example: x = 10;

**2)Floating-Point (float):**

Represents numbers with fractional parts (decimals).

Example: y = 3.14

**3)Character (char):**

Represents a single character or symbol.

Example: char letter = 'A';

**4)Boolean (bool):**

Represents two possible values: true or false.

Example: isStudent = true;

Example: name = "Alice"

**5)Byte:**

A small unit of storage, usually 8 bits, capable of storing values from 0 to 255.

Example: byte age = 25;

**2. Non-Primitive Data Types**

Non-primitive data types are more complex and usually represent collections or more structured data. These types are built from primitive types and may hold a reference to the data stored in memory.

**Common Non-Primitive Data Types:**

1. **Arrays**
   * An array is a collection of elements, all of which are of the same type. Arrays can store multiple values, accessed by an index.
   * Example: **int[] numbers = {1, 2, 3, 4, 5};**
2. **Strings** (in languages where strings are not primitive types)
   * A string is an object that stores a sequence of characters, and can have more functionality (e.g., methods for manipulating text).
   * Example: **String name = "Alice";**
3. **Classes/Objects**
   * A class is a blueprint for creating objects. Objects are instances of classes and can contain both data and methods.
   * Example:

**java**

**Copy code**

**class Car {**

**String model;**

**int year;**

**Car(String model, int year) {**

**this.model = model;**

**this.year = year;**

**}**

**}**

**Car myCar = new Car("Toyota", 2020);**

1. **Lists/Collections (like ArrayList, LinkedList)**
   * These are data structures that store multiple values or objects and can be resized or modified dynamically.
   * Example (in Java): **ArrayList<String> names = new ArrayList<String>();**
2. **Dictionaries/Maps**
   * A map (or dictionary) is a collection of key-value pairs, where each key maps to a specific value.
   * Example (in Python): **dict person = {"name": "John", "age": 30};**
3. **Sets**
   * A set is a collection that stores unique values (no duplicates).
   * Example: **Set<String> uniqueNames = new HashSet<String>();**