Java Installation

* Download JDK version 21.0 and install it.
* After that, Download and install IDE (Recommended Intelli J Community Edition).
* Setup paths.

Git Basics

* Git is a version Control System.
* It is used to maintain and manage the code development on our local machine.

GitHub

* GitHub is cloud - based platform that lets you to manage, maintain and collaborate remotely.
* To create and update from our local system to remote repository, we have some git commands like Init, add, push, pull etc…

**Init**: used to initialize .git in your local repository

**Add**: it a command used to add files to stage, which files will get into tracked changes.

**Commit**: it is used to save your staged changes.

**Push**: it is used to push your local changes or repositories to remote repositories.

**Pull**: it is used to fetch changes from remote repository (GitHub).

**Merge**: it is used to combine changes from one branch into another.

**Merge conflicts**:

-> If there are no conflicts, Git just combines the changes.

-> If there are conflicts, Git will stop and ask you to resolve them manually.

**Clone**: it is used to clone the remote repository into your local system.

**Branch:** it is separate line of development, where your changes will not affect the main project.

**Data Types:**

* Data type is used to specify which type of data should be stored.

There are 2 Types of data types.

1. Primitive
2. Non – Primitive

**1.Primitive data type**

It is a predefined type that stores values directly in memory

There are 8 data types:

| **Data Type** | **Size** | **Description** |
| --- | --- | --- |
| byte | 1 byte | Small integer (-128 to 127) |
| short | 2 bytes | Medium integer (-32,768 to 32,767) |
| int | 4 bytes | Standard integer (-2^31 to 2^31-1) |
| long | 8 bytes | Large integer |
| float | 4 bytes | Decimal number (less precision) |
| double | 8 bytes | Decimal number (more precision) |
| char | 2 bytes | Single character (Unicode) |
| boolean | 1 bit | Logical values: true or false |

2) **Non-primitive Data type**

It a data type which is storing reference address of memory, which data is stored in that, rather than storing direct values.

Ex: String

All custom objects will undergo Non-primitive type.

**Variables**

Variables are block of memory where we can store data in it.

Syntax:

DataType variableName = value;

Ex: int a = 20;

* Variable re-declaration is not possible, variable Re-Intialization is Possible.
* Removing old data and updating with new data is known as Variable Re-initialization.

There are 3 types of Variables

1. Local Variables
2. Static Variables
3. Instance Variables (Non – Static)
4. **Local Variables**

* The variables which are declared within any method, block or constructor.
* It can be accessible within that block.
* It should be initialized before using it.

Ex:

Public static void greet(){

String msge = “Hello”;

System.out.println(msge);

}

1. **Static Variables**

* The variables which are declared inside class and outside any method or block with “ static ” keyword is called static variables.
* Initialized only once when class is loaded.
* It can be accessible anywhere within the class.
* For each object of the class, it is same.

Ex:

class Demo{

static int a = 30;

public static void main(String[] args){

System.out.println(a);

}

}

1. **Non-static Variables**

* The variables which are declared inside class and outside any method without declaring static keyword is known as Non-static variables.
* Each object of the class, get its own copy.
* It contains default values like 0, 0.0, false, null.

Ex:

class Demo{

int age;

public static void main(String[] args){

System.out.println(“age : “+age);

}

}

**Operators**

* Operators are used to perform some operations on data or values.
* There are 5 types of operators.

1. Arithmetic Operators
2. Relational Operators.
3. Logical Operators.
4. Unary Operators.
5. Ternary Operators.
6. **Arithmetic Operators**

* The operators which are used to perform mathematical operations, such as

Addition( + ) , Subtraction ( - ), Division( / ), Multiplication( \* ), Modulus ( % ).

Ex:

int x = 10;

int y = 20;

int add = x + y;

int mul = x \* y;

int sub = x – y ;

int div = x / y;

int mod = x % y;

System.out.println( add ); // 30

System.out.println( mul ); // 200

System.out.println( sub ); // -10

System.out.println( div ); // 0

System.out.println( mod ); // 10