## 1.Components of Java Architecture

The Java architecture includes the three main components:

* Java Virtual Machine (JVM)
* Java Runtime Environment (JRE)
* Java Development Kit (JDK)

Java Virtual Machine

The main feature of Java is **WORA**. WORA stands for **Write Once Run Anywhere**. The feature states that we can write our code once and use it anywhere or on any operating system. Our Java program can run any of the platforms only because of the Java Virtual Machine. It is a Java platform component that gives us an environment to execute java programs. JVM's main task is to convert byte code into machine code.

### Java Runtime Environment

It provides an environment in which Java programs are executed. JRE takes our Java code, integrates it with the required libraries, and then starts the JVM to execute it.

### Java Development Kit

It is a software development environment used in the development of Java applications and applets. Java Development Kit holds JRE, a compiler, an interpreter or loader, and several development tools in it

# 2.Objects and Classes in Java

An entity that has state and behavior is known as an object e.g., chair, bike.

An object has three characteristics:

* **State:** represents the data (value) of an object.
* **Behavior:** represents the behavior (functionality) of an object such as deposit, withdraw, etc.
* **Identity:** An object identity is typically implemented via a unique ID.

A class is a group of objects which have common properties. It is a template or blueprint from which objects are created. It is a logical entity. It can't be physical.

A class in Java can contain:

* **Fields**
* **Methods**
* **Constructors**
* **Blocks**
* **Nested class and interface**

Syntax to declare a class:

**class** <class\_name>{

    field;

    method;

}

# 3. Access Modifiers in Java

The access modifiers in Java specifies the accessibility or scope of a field, method, constructor, or class. We can change the access level of fields, constructors, methods, and class by applying the access modifier on it.

There are four types of Java access modifiers:

1. **Private**: The access level of a private modifier is only within the class. It cannot be accessed from outside the class.
2. **Default**: The access level of a default modifier is only within the package. It cannot be accessed from outside the package. If you do not specify any access level, it will be the default.
3. **Protected**: The access level of a protected modifier is within the package and outside the package through child class. If you do not make the child class, it cannot be accessed from outside the package.
4. **Public**: The access level of a public modifier is everywhere. It can be accessed from within the class, outside the class, within the package and outside the package.

# 4.Java Garbage Collection

In java, garbage means unreferenced objects.

Garbage Collection is process of reclaiming the runtime unused memory automatically. In other words, it is a way to destroy the unused objects.

To do so, we were using free() function in C language and delete() in C++. But, in java it is performed automatically. So, java provides better memory management.

### Advantage of Garbage Collection

* It makes java **memory efficient** because garbage collector removes the unreferenced objects from heap memory.
* It is **automatically done** by the garbage collector(a part of JVM) so we don't need to make extra efforts.

### Decision-Making statements:

decision-making statements decide which statement to execute and when. Decision-making statements evaluate the Boolean expression and control the program flow depending upon

the result of the condition provided.

1) If Statement:

It evaluates a Boolean expression and enables the program to enter a block of code if the expression evaluates to true.

**Syntax:**

**if**(condition) {

statement 1; //executes when condition is true

}

### 2) if-else statement

The [if-else statement](https://www.javatpoint.com/java-if-else) is an extension to the if-statement, which uses another block of code, i.e., else block. The else block is executed if the condition of the if-block is evaluated as false.

**Syntax:**

**if**(condition) {

statement 1; //executes when condition is true

}

**else**{

statement 2; //executes when condition is false

}

### 3) if-else-if ladder:

 It is the chain of if-else statements that create a decision tree where the program may enter in the block of code where the condition is true. We can also define an else statement at the end of the chain.

**Syntax:**

**if**(condition 1) {

statement 1; //executes when condition 1 is true

}

**else** **if**(condition 2) {

statement 2; //executes when condition 2 is true

}

**else** {

statement 2; //executes when all the conditions are false

}

### 4. Nested if-statement

In nested if-statements, the if statement can contain a **if** or **if-else** statement inside another if or else-if statement.

**Syntax:**

**if**(condition 1) {

statement 1; //executes when condition 1 is true

**if**(condition 2) {

statement 2; //executes when condition 2 is true

}

**else**{

statement 2; //executes when condition 2 is false

}

}

### Switch Statement:

The switch statement contains multiple blocks of code called cases and a single case is executed based on the variable which is being switched.

**Syntax:**

**switch** (expression){

**case** value1:

    statement1;

**break**;

   .

    .

    .

**case** valueN:

     statementN;

**break**;

**default**:

**default** statement;

}