JAVA

Why we learn java

- it’s open source.
- Platform independent.
- Most demanding language.
- Easy to use and learn.
- More securable language.

History of java

- Released by **sun microsystem** in the year **1995.**
- Father of java **James Goosling.**
- Initially named as **“oak”** was later rename as **“java”.**
- There have been several releases(**Versions**) of java programming language.
- Stable version in use java8 or javaSE8.
- Latest version of java is java18 or javaSE18.

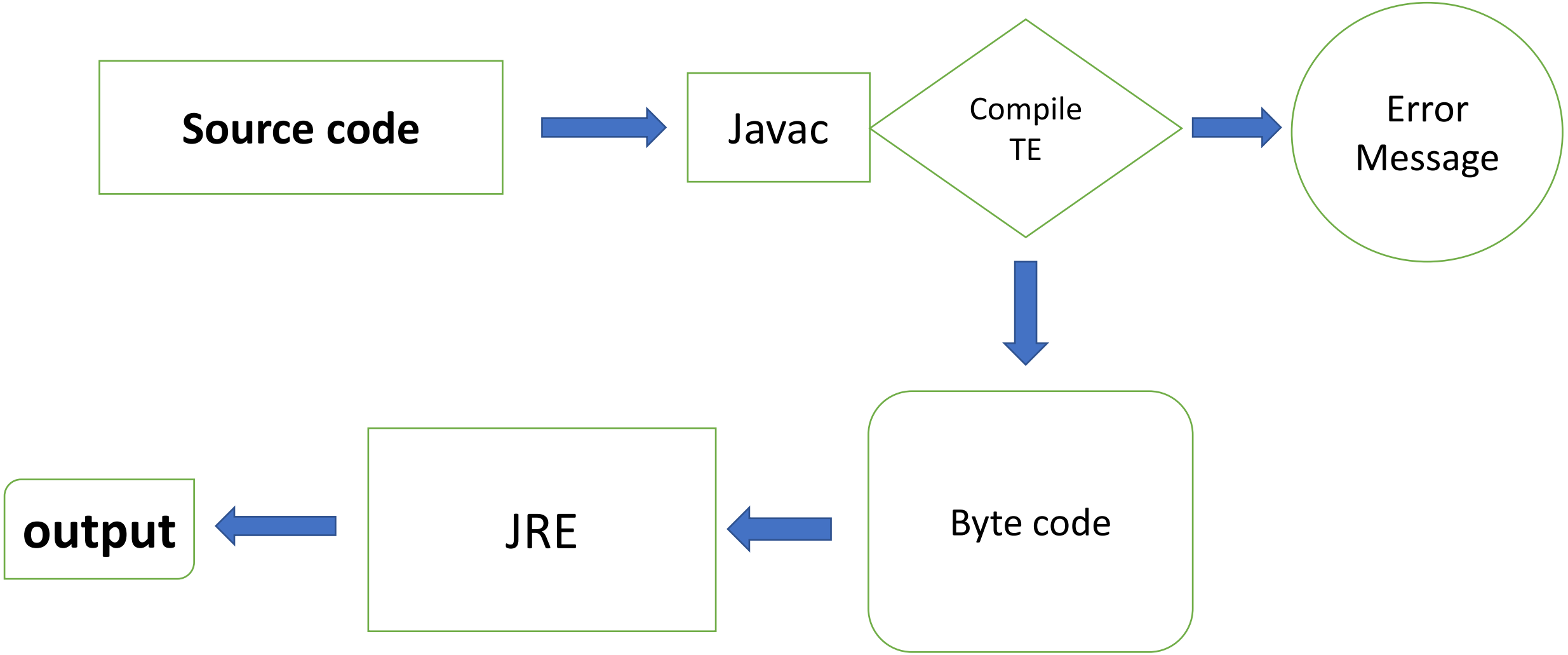
What is java?

Java is multipurpose high level , object oriented , platform independent programming language.

High level – human understanding language.

Oops – readability , reusability (abstraction, inheritance , encapsulation ,polymorphism).

Platform independent – once we create a java program , same program we can execute in all platforms.


Source code
Javac
Compile
TE
Error
Message
Byte code
JRE
output

Java life cycle

- The programmer creates the source code and saves program with a filename “.java” file extension.
- The “.java” file is provided to the compiler which checks the syntax level mistakes made by the programmer.
- If mistakes are found the compiler displays appropriate error message.
- If no mistakes are found the compiler generates “.class” file which contains java bytecode.
- The java bytecode is then executed by the JVM with the help of it’s subsystem in the JRE.
- The programmer performs basic output analysis.

Byte code

- Highly optimized version of java source code.
- It’s generated by the compiler and is presented in the “.class” file.
- The bytecode is not human readable
- The jvm can understand bytecode and it can be considered as java executable code

Java c

- **Javac** is the name of the java compiler.
- It takes the java source code in **“.java”** file as input and checks for syntax level mistakes.
- If programmer has made mistakes it displays corresponding error messages.
- If there are no mistakes the compiler generates **“.class”** file with the bytecode.

JVM

- This is the execution engine of java language.
- The jvm is responsible for all runtime activities in java language.
- The jvm executes bytecode line by line and generates output.
- It’s a virtual machine and does not have a physical existance.

JRE

- Java runtime environment
- The jvm uses several tools during execution like

a) Linkers
b) Verifiers
c) Memory managers
d) Loaders

- Event handlers……extc
- This collection of tools necessary for execution is packaged as the jre
- All activities in the jre is controlled by the jvm

Order of execution

- In java the vertical order of execution in top-bottom.
- In java the horizontal

Basic syntax:

Class: basic fundamental units of java programming.

Main method:entery point of execution

Ex:

```
Class classname{
Public static void main(string[] args)
}
```

Output statement:

Inbuild java statement.

It’s used to display content on output screen.

String values should be enclosed within(“….”)double quotes.

Char values enclosed in(‘….’) single quotes.

Numbers and Boolean values have to be display’s directly.

Keywords

Keywords are inbuild words having a special meaning

As a programmer we can use keywords but we cannot change the meaning.

Keywords are completely in lower case in a standard editor keyword will appear in a shade of blue color.

Java provides ttl no of 53 keywords.

Goto and constant are two keywords that are not used in java programming.

Primitives and variables

Primitives are used to create variables.

Int a=10;

Java language provide total of eight primitives.

Byte->1 byte

Short->2 byte

Int->4 byte

Long->8 byte

Float->4 byte

Double->8 byte

Char->2 byte

Boolean->1 bit

Variables:

Variable is a token or container used to store a value.

Data type . variable name=value;

Reinitialization:

Replacing the old value in a variable with a new value is called as reinitialization.

Ex;

Int a;

a=10;//initialization

a=20;//reintialization

Final variable

A variable created with final keyword is called as a final variable.

Ex:

Final int a=10;

If a variable is created with final keyword it cannot be re-initialized.

Ex:

Final int val=20;

Int val=30;

Operator:

Operators are symbol use to perform a certain task.

Type of operations, operators are classified into familitie’s.

1) Arithmetic operator
2) Relational operator
3) Logical operator
4) Bitwise operator(b.or,b.xor,b.leftshift,b.rightshift)
5) Unary operator(post oper)and(pre oper)

Control statement:

If we want to control the flow of execution of a program we will make use of control statements.

Branching statements:

If

If else

Else if leader

Nested if

Switch

Looping

For

While

Do while

Syntax:

If{

---whether the condition is true or false

}

If else:

If{

It will execute only the condition is true

}

Else{

It will execute only the condition is false

}

Syntax: else if leader

```
If(condition){
-----
}
Else if(condition){
---
}
Else if(condition){
---
}
Else{
---
}
```

Nested if

If(condition){

--

If(condition){

}

Else{

}

Else{

}

Switch

```
Switch(choice){
Case:1
Break
Case:2
Break
Case:3
Break
Default:s.o.p(“…….”)
}
```

METHODS

- Methods are set of instruction written by the programmer in order to perform some specific task . which is run’s when it’s called.

Advantage:

Code reusability.

Method can be declared within a class we declared method using().

Identifier

Identifier is a name given that a class name ,method name

As per the it standard while giving a class name every word first character must be in upper case.

As per it standard while giving a method name , first word all the character will be in lower case and from second word on words the first character must be in uppercase.

Taking input from the user

1)Import java.util.Scanner;

2)Scanner sc=new Scanner(System.in)

3)Int a=sc.nextInt();

Double b=sc.nextDouble();

Float c=sc.nextFloat();

Boolean d=sc.nextBoolean();

Char f=sc.nextCharAt();

Short g=sc.nextShort();

DEFINITION:

An array is a finite set of homogeneous values.

Finite-fixed in size

Set-group of values

Homogeneous-same data type

How to create arrays:

1) Using dimension
2) Using array initialization

Using dimension:

Datatype[] variable name=new datatype [size];

Int x=new int[5]

Using array initialization:

Datatype[] variable name=[value1,value2…….Vn];

Int x=[1,2,3,4,5]

Index value always begin at ‘0’.

Index value always end at size-1.

String

It’s predefined class which is available in the inbuild library folder.

String class contain some inbuild method’s.

Int length()

Responsible for returning the total no of characters for a given string.

Char charAt(int index)

Responsible for returning the character present at the given index value

Boolean equals(String s)

Responsible for compare the content of two string values.if two strings are same content,then it will return true else it will return false.

Object

An object is an entity which will have it’s own states and behavior.

State->refers to variables or data member.

Behavior->refers to methods or member function.

Class

An class is a java definition block or a java blue print ,which will contain different states and behavior.

State->refers to variables or data member.

Behavior->refers to methods or member function.

Static variable or static data member:

The variable which are declared outside a method using static keyword are known as static variable.

The variable which are declared outside a method without using static keyword are known as non-static variable.

local variable

The variables which are declared inside a method are known as local variable’s.

Static method and non static method:

The method which are declared using static keyword known as static method.

The method which are declared without using static keyword non static method.

Accessing static properties of one class into another class.

Syntax:

Classname.static member name;

During class loading process all the static member will be loading into

The class memory or class area which will have the same name as that of class name. This the reason we access static properties one class into another class by using the class name.

Accessing non static properties one class into another class:

Step 1:create a object

Syntax:

New class name();

Step2:assign reference variable to the object:

Syntax:

Class name reference variable=new class name();

CONSTRUCTOR

Constructor are a special method or member function which will have same name as that of the class name.

Used to initialize non-static variables.

Properties of constructor :

Constructor will not allow any modifiers(static,final,abstract)

Constructor will not allow any return type.

Constructor will execute automatically when we create an object of a class.

Syntax:

```
Class sample{
   public sample(){
-----
-----
}
}
```

Purpose of constructor

It’s used to initialize the non-static variable which are present inside the object .

Using constructor we can initialize the member of one object at a time.

Question:

Create a car class with the attributes name ,price, and color initialize these attributes with the help of constructor for 2 cars and display their details?

Constructor overloading

The process of writing multiple constructor within the same class but different in their argument known as constructor overloading.