Java Full Stack

5 phases

Phase 1

Agile

Git : few topics self learning – few topics covered by trainer.

Core Java

Basic Programming

OOPs

Exception handling

Multithreading overview

File handling

Collection Framework and Data structure

Maven : self learning and few topics covered by trainer.

Phase 2

Phase 3

Phase 4

Phase 5

Capstone Project : final project

Day 1 : 16-05-2022

GIT: Git is open source sub version control system tool which help to record the program or application execution.

SVN :

GIT : it is a type of distributed sub version control system which help to share the data between more than one team or developer or programmers.

In Git will provide repository : repository mean like a folder which hold any type of files or program or folder or application.

Git provide local as well as remote repository( git hub or git lab or aws and azure etc).

Git commands

Git --version

git init (this command is use to make the local folder as local repository) this command you have to execute only once.

git status (this command is use to find the current status of local repository)

we have to add the file from local folder to staging area.

git add filename ( This command is use to add the file from local folder to staging area)

or

git add . (this command is use to add all files and folder present in local directory)

now we have to move the file or folder from staging area to local repository

git commit –m “commit message”

staging are : this area provided by git which is use to store the set of files and folder before commit.

Remote repository can be git hub or git lab or AWS or Azure etc.

git branch -M main (this command is use to rename the branch)

git remote add origin URL (link local repository with remote repository)

git remote add origin https://github.com/Kaleakash/test112233.git

(this command is use to connect local repository to remote repository)

We will genera the Token which help to push the local repository code to remote repository.

git push –u origin main (this command is use to push the code from local repository to remote repository)

git clone url (this command is use to clone any public remote repository into local machine)

it is use to download fresh copy in local machine

git pull : it is use to get new update in existing repository

git branch : git branch is like a pointer which hold more than one commit details.

git branch (this command is use to check the branch name available in local machine)

git branch branchName (this command is use to create the branch)

to switch to the branch

git branch –D branchName This command is use to delete the branch

git merge banchName This command is use to merge the branch code into current branch.

Day 2: 17-05-2022

Java : Java is platform independent and pure object oriented programming language.

1990

1991 they introduce one language ie Oak. In Nov 1995 it rename to Java.

Java developed by James gosling and team. It was belong to sun micro system and part of Oracle.

Java Version 1.0…………………1.7 Java 8 ……………………18.

Java

J2SE J2EE J2ME

JavaSE JavaEE JavaME

JSE JEE JME

Java Standard Java Enterprise Java Miro

Edition Edition Edition

Core Java web application chips programing

Standalone

Or

Desktop

Application

Phase 1 Phase 2

OOPs : Object Oriented Programming system

object : object is any real world entity.

Ex : Person, Place, Bank, Animal, Car

Properties or state -🡪 have 🡪

Person

Behaviour 🡪 do/does

Have -🡪

Car

Behaviour

Object is a concept.

class : class is known as blue print of object or template object or using class we can describe the object or class is user-defined data type which is to create the object.

In Java if we want to display any simple message we have to take the help of class.

Syntax of class

class className {

fields or variable

methods or functions

}

Class name must be follow some rules

In Java class name must be follow Pascal naming rules.

1. If class contains one word it must start with upper case.
2. If class container more than one word each word first letter upper case.
3. Please write meaningful name for class while developing the application.

class Test {

public static void main(String args[]) {

System.out.println(“Welcome to Java”);

}

}

Once installed the Java you can see jdk and jre

Java development kit

Java run time environment

Write the program in Notepad

class Test {

public static void main(String args[]) {

System.out.println("Welcome to Java....");

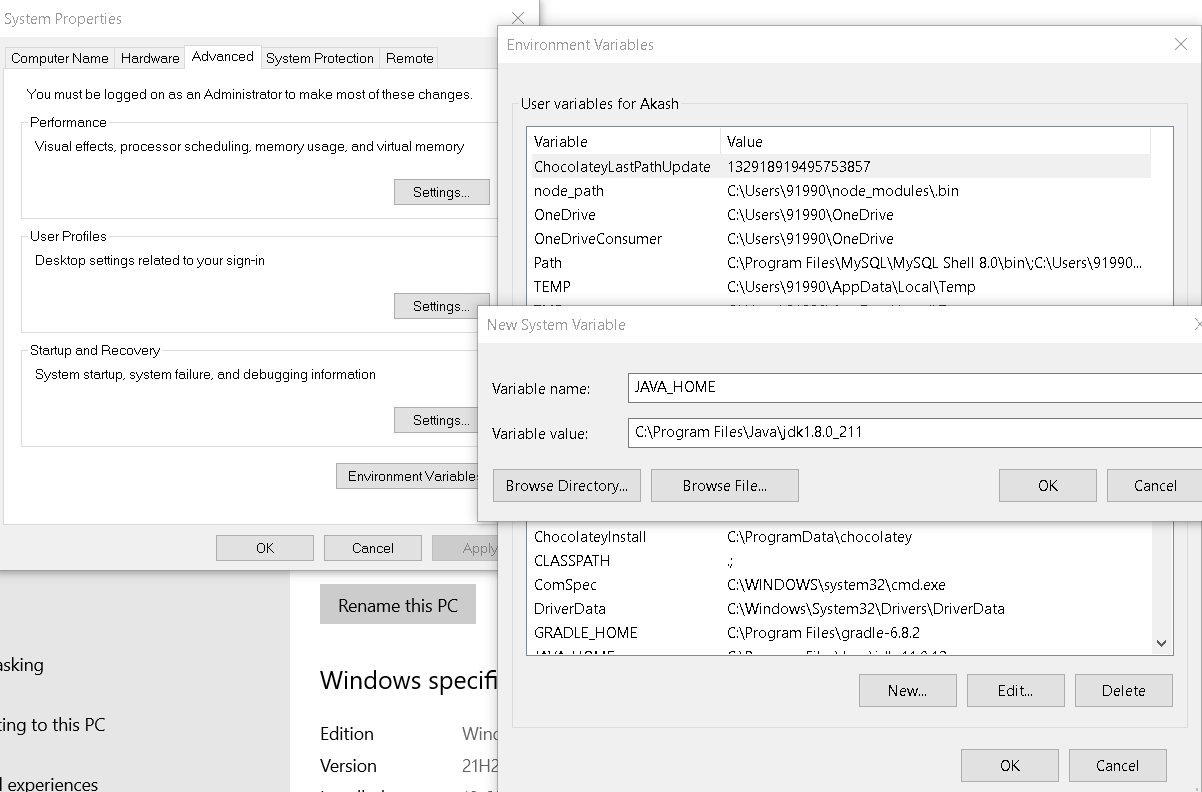
}

}

Save the program with Test.java.

Copy the Java home

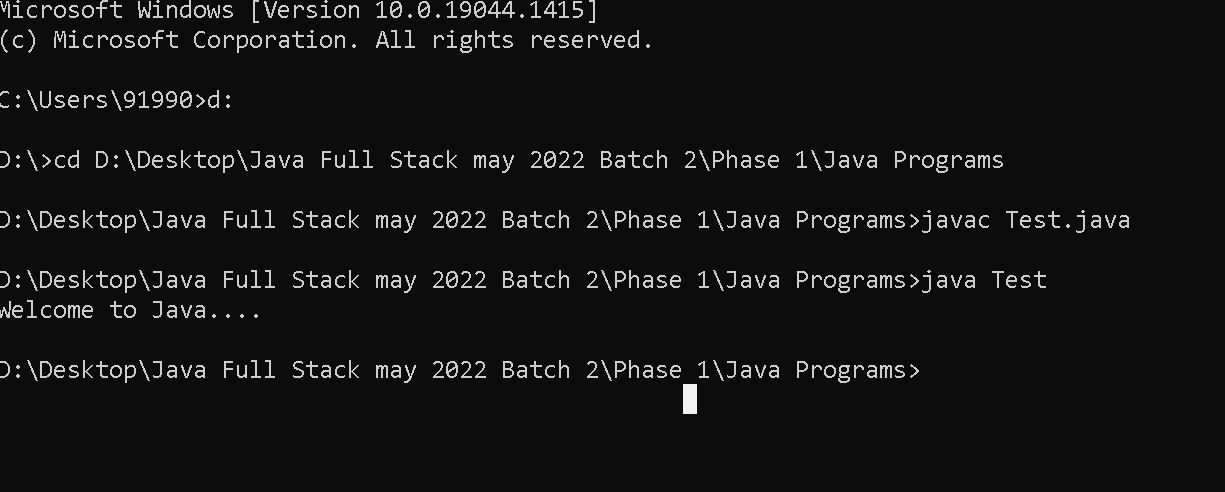
C:\Program Files\Java\jdk1.8.0\_211



javac

java

Open the command prompt



This program is use to display the simple message

class Test {

public static void main(String args[]) {

System.out.println("Welcome to Java....");

System.out.println("Welcome to Java....");

System.out.println("Welcome to Java....");

}

}

Data types : data type is a type of data which tells which type of data it will hold.

In Java data types mainly divided into two types.

1. Primitive data type : these data types is use to store only value
2. Non primitive data type or reference data types : these data type is use to store value as well as reference of another data types.

Primitive data types

1. byte 1 byte
2. short 2 byte
3. int 4 byte
4. long : without decimal 8 byte
5. float 4 byte
6. double : with decimal 8 byte
7. char : single character 2 byte
8. boolean : true or false 1 bit

Simple Data types example

class Test {

public static void main(String args[]) {

int a=10;

double b = 10.20;

char c = '\*';

boolean res = true;

System.out.println(a);

System.out.println("VAlue of a is "+a);

System.out.println("VAlue of b is "+b);

System.out.println("VAlue of c is "+c);

System.out.println("VAlue of res is "+res);

}

}

Type casting : converting from one data type to another data type is known as type casting

Two types of type casting

1. implicit type casting : automatically convert it
2. explicit type casting : we have to convert it

int family

---------------🡪 implicit --------------🡪

byte short int long

🡨-------------explicit ---------------------------------

**Type casting first example**

class Test {

public static void main(String args[]) {

byte a=10;

short b=a; // implicit type casting done

System.out.println(a);

System.out.println(b);

short c = 10;

//byte d =(type)c;

byte d = (byte)c; // explicit type casting

System.out.println(c);

System.out.println(d);

}

}

Type casting int to float and vice-versa

Implicit

Int 🡨-----------------------🡪 float

Explicit

In Java every decimal number by default double consider.

Type casting second example (int and float data types)

class Test {

public static void main(String args[]) {

int a=10;

float b=a; // implicit type casting

System.out.println(a);

System.out.println(b);

//float d =10.10f; // explicit type casting

float d = (float)10.10; // explicit type casting

int e = (int)d; // explicit type casting we will loose the decimal point number

System.out.println(d);

System.out.println(e);

}

}

Operator : operator is use to do the operations

1. arithmetic operator : +, -, \*, /, %
2. conditional operator : >, >=, <, <=, ==, !=
3. assignment operator : =
4. increment and decrement operator : ++, --
5. logical operator : &&, !!

class Test {

public static void main(String args[]) {

int a=10;

int b=5;

int sum = a+b;

int sub = a-b;

int mul = a\*b;

int div = a/b;

int mod = a%b; // remainider

System.out.println("Sum "+sum);

System.out.println("Sub "+sub);

System.out.println("Mul"+mul);

System.out.println("Div "+div);

System.out.println("Mod "+mod);

}

}

Operator 2nd example

class Test {

public static void main(String args[]) {

int a=10;

int b=5;

int c =20;

int sum = a+b;

int sub = a-b;

int mul = a\*b;

int div = a/b;

int mod = a%b; // remainider

System.out.println("Sum "+sum);

System.out.println("Sub "+sub);

System.out.println("Mul"+mul);

System.out.println("Div "+div);

System.out.println("Mod "+mod);

boolean res1 = a > b;

boolean res2 = a > b && a > c; // both condition must be true then result is true

boolean res3 = a > b || a > c; // any one condition must be true then result is true

System.out.println(res1);

System.out.println(res2);

System.out.println(res3);

}

}

If statement :

1. simple if statement
2. if else
3. if else if
4. switch statement

simple if statement

class Test {

public static void main(String args[]) {

int a=10;

int b=50;

if(a>b) {

System.out.println("Yes.....");

}

System.out.println("Finish");

}

}

class Test {

public static void main(String args[]) {

int a=10;

int b=50;

if(a>b) {

System.out.println("Yes.....");

System.out.println("a is largest");

}else {

System.out.println("No.....");

System.out.println("b is largest");

}

System.out.println("Finish");

}

}

Switch statement syntax

In switch statement user can take the decision which block or set of code want to execute.

Syntax

switch(variableName) {

case label1 : block1;

break;

case label 2: block2;

break;

case label 3: block3;

break;

case label 4: block4;

break;

default : wrong choice

break;

}

switch, case, break and default are keywords.

Switch statement example

class Test {

public static void main(String args[]) {

int choice=10;

switch(choice) {

case 1: System.out.println("block1");

break;

case 2: System.out.println("block2");

break;

case 3: System.out.println("block3");

break;

default : System.out.println("Wrong choice");

break;

}

System.out.println("finish");

}

}

In Java We can take the value through keyboard lot of ways.

One of the way is using Scanner class.

Scanner is a pre-defined class which provide set of methods which help to scan the value through keyboards.

We have to create Scanner class object

Syntax

Scanner obj = new Scanner(System.in);

Scanner is a pre-defined class part of util package. Package is a collection of classes and interfaces.

So whenever we create Scanner class object we have to import util package.

obj.nextInt() scan the int value

obj.nextFloat() scan the float value

obj.nextDouble() scan the double value

Scanner class example

//import java.util.Scanner;

import java.util.\*;

class Test {

public static void main(String args[]) {

Scanner obj = new Scanner(System.in);

System.out.println("Enter the value of a ");

int a = obj.nextInt(); // it is use to scan the int value

System.out.println("Enter the value of b ");

int b = obj.nextInt(); // it is use to scan the int value

int sum = a+b;

System.out.println("Sum of two number is "+sum);

}

}

looping : it is use to execute the set statement again and again till the condition become false.

While loop

Do while loop

//import java.util.Scanner;

import java.util.\*;

class Test {

public static void main(String args[]) {

// while loop

/\*

int i=0,n=10; // initiailzation

while(i <= n) { // condition true // entry loop

System.out.println("The value of i is "+i);

i++; //i=i+1

}

\*/

// do while loop

int i=0,n=10;

do {

System.out.println("The value of i is "+i);

i++;

}while(i>=n); // exit loop consider

System.out.println("Finish");

}

}

For loop