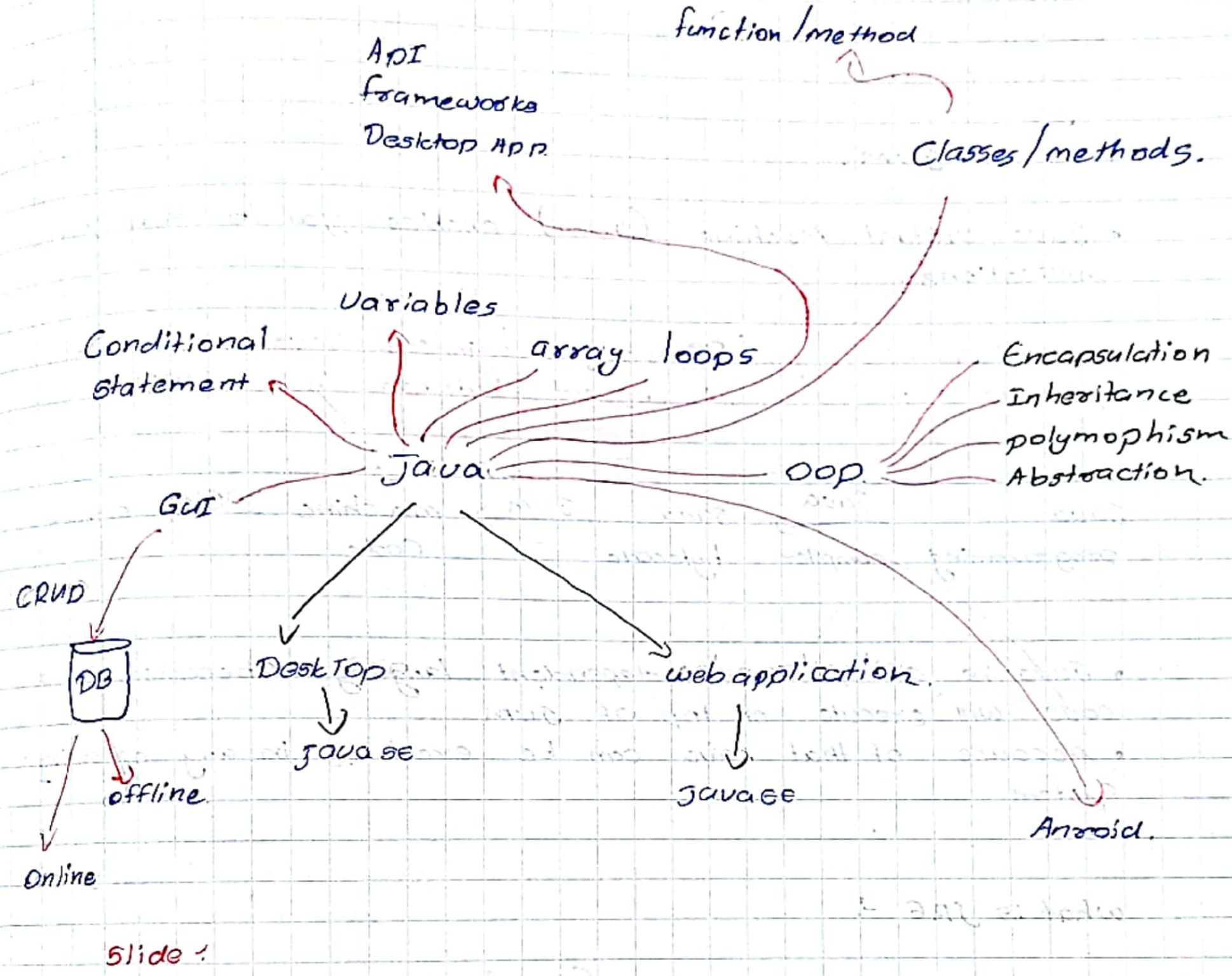
08/09/2022.

Java programming.



Slide :

- Java is a hihgh-level, class bassad objected oriented progamming language

- used to develop desktop ~~lang~~, mobile and web application, bigdata processing, embedded system etc.

- Oractal owned java.

Note :

- Java is a highlevel, class basedsed oop language.

- By using java variouse type of applications are capable of developing.

Java → Desktop Apps / stand alone app.

Jsp → web

Native → mobile applications.
anroide

What is JVM.

- Java virtual machine (JVM) enables you to run java applications.
- Java compiler will convert high level java code in to bytecode then JVM will convert bytecode in to machine code

Java programming → (Java compiler) → Java bytecode → (JVM) → machine code → (CPU) → output

- Java is a platform independent language because Java code will execute on top of JVM.
- Because of that Java can be executed in any operating System

what is JRE :-

- Java runtime Environment [JRE] it's a software package which provide class libraries, JVM and other necessary Components to run the java application.

what is JDK

- Java Development kit (JDK) it is a software developmon kit which contain JRE, compiler, Java Doc etc
- JDK will convert high level program to machine code in Java Development.
- To execute java application in any computer JDK is an essential thing.

Java variables.

- A variable is a memory allocation of Random access memory which is capable of holding relevent data in it.
- Declearing variables will create memory allocation in RAM.
- Each variable should have a unique variable name which works as an identifier to identify them seperally.

Java Data types.

- Data types will decied what type of data which works with ~~work~~ declared variables.
- In java all variables should declared before us it used.
- In Java following data types are available,

```
int     - works with integer data.
String  = works with string values /words.
double  = works with numericale data with floating points.
bool    = Stores 1, 0 or "True" or "false".
```

ex

```
int = 0;
String = name;
double = avg;
```

Java operations

- Operation is a syst Symbol which we are using in programming to meet certain requirements
- Depending on the Contribution of these operations it can be categerize as follow.

1. Arithmetic Operators.
   Ex : (x), (+), (-), (/)
2. Assignment operators
   Ex = (=), (!=)

3 Relational operators.

Ex = (==), (!==)

4 Logical operators

Ex = &&, ||, !|

Java basic input and Output.

Java output.

In Java there are 3 inbuilt functions can be used to display content on Java applications.

1.

System.out.print();

This method will display content and remain the curser on the same line.

2.

System.out.println();

This method will display content and move the curser to the next line.

3.

System.out.printf();

This method is similar with printf method in C programming.

Ex

code. :-

```
public static void main (String[] args) {

    System.out.print ("Hello world");
}
```

printing variables in java.

ex

code. :-

```
public static void main (String[] args) {

    double number = 10.6;

    System.out.println[5];
    System.out.println[number];

    System.out.println ("I am" + "awesome");
    System.out.println ("number = " + number);
}
```

Q1

Create a java application to declare 2 integer variables and initialise them with values get the answer for the basic arithmetic operations and display.

code

→

```
Public static void main (String [] args) {

    int num1 = 30;
    int num2 = 60;

    System.out.println("Sum = " + num1 + num2);

    System.out.println("Subtraction = "+num1 - num2);

    System.out.println("multiplication = " + num1 * num2);

    System.out.println("Division = " + num1 /num2);
```


15/9/2022.

Java input.

- In java user inputs has been taken in a different way
- When taking user inputs following methods should follow
    1. Import the class library file
    2. Create the class object using import.
    3. Through the created object get user inputs.

- above class library is known as

```
import java.util.Scanner;
```

- When taking multiple inputs one Class object is required inside the class file.

- When expecting the user input values to the java application Scanner object should create as follow

```
Scanner objscan = new Scanner (System.in);
```

- In above object creation object name is the only variable that can be change.

Ex :-

```
import java.utill.Scanner ;
class input {
    public static void main(String [] args) {
    Scanner input = new Scanner (System.in);
    //Getting float input
    System.out.println ("Enter a float value : ");
    float f = input.nextFloat ();
    System.out.println ("value is = " + f );

    //Getting double input
    System.out.println ("Enter the double variable : ");
    double fd = input.nextDouble ();
    System.out.println ("value is = " +d );
    //Getting String input
    System.out.println ("Enter the word : ");
    String s = input.next();
    System.out.println ("word is = " +s );
```

Q1

Create a java application to enter you name and display a greeting message.

Code

→

```
import java.util.Scanner;

class input {

    public static void main(String [] args) {

    Scanner objname = new Scanner(System.in);

    System.out.println("Enter you name :");

    String name =objname.next();

    System.out.println("wecome "+name);

    }
```

Q2

Create a java application to get two int user inputs and display multiplication.

Code :

~~clas~~

```
import java.util.Scanner;

class input {

    public static void main[String [] args ] {

    Scanner objnum = new Scanner(System.in ];

    System.out.println[" Enter the 1st number "];

    int num1 = objnum.nextInt();

    System.out.println[" Enter the 2nd number "];

    int num2 =objnum.nextInt();

    System.out.print[" Answer is = " (num1 x num2)];

}
```

Q3 Create a java application to check whether user input value odd or even

Code :-

```
import java.util.Scanner ;

class input {

    Scanner.out.println("Enter a number : ");

    int num1 =
    Public static void main(String[] args ) {
    Scanner objoddeven = new Scanner (System.in);
    System.out.println("Enter a number : ");
    int num1 = objoddeven.nextInt();
    if (num1 % 2 == 0 )
    {
        System.out.println("It's an even number ");
    }
    else
      {
        System.out.println("It's an odd number ");
      }

    }
}
```

Java loops

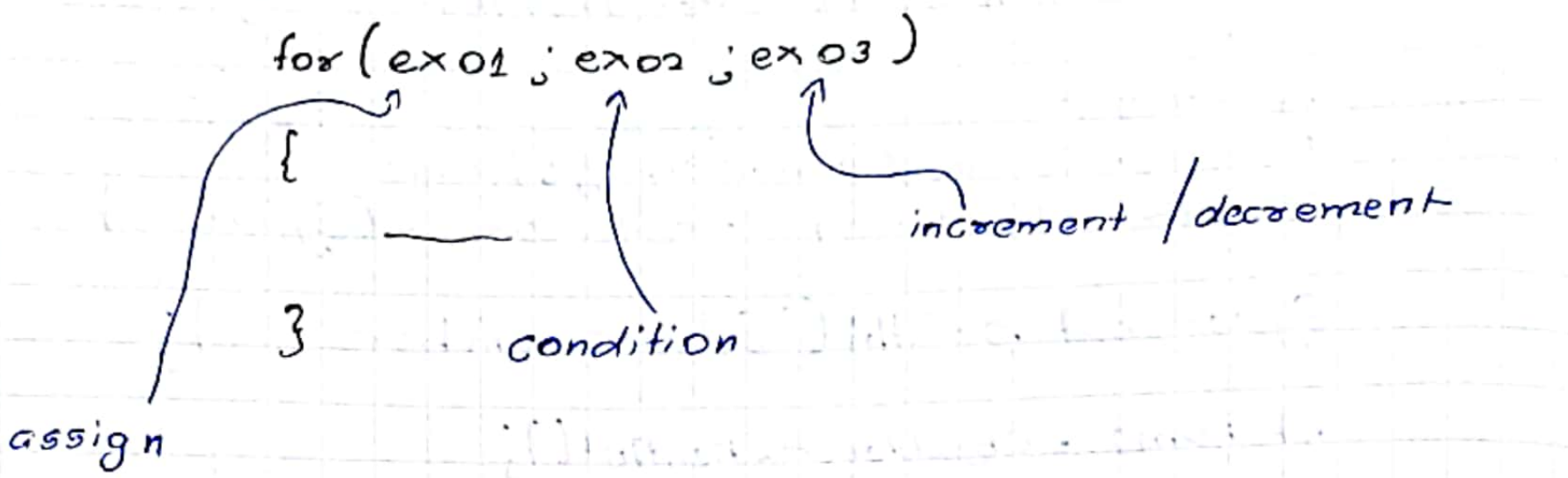
- Loops can be use in iterative programs in java programmin -g g
- Loops are capable of repeating the given code untill the condition is true
- In java programming for Loop, while loop and do while loop are commonly used.

for Loop

- foor loop consists with 3 expression and untill the condition will true loop will iterate

Syntax :-

~~forb~~



while loop.

- This loop again can be used of for iterative program until the condition is toue.

Syntax

```
while (Condition)
  {
  body
  Increment /decrement
  }.
```

dowhile

- This loop ~~again~~ will execute at leaset one iteration even the condition is false.
- Reason for that is the condition fa of dowhile loop is Checked at the end.

Syntax :-

```
do
{
    body
    Increment /decrement
} while (condition) ;
```

Q_4

Display the 1st 10 numbers using for, while, and dowhile loop.

Code. -:

```
Class loops {

    public static void main(System [] args) {

    // for loop

     for(int a=0; a<=10 ; a++)

      {
          System.out.println ("\t \n" +a );
      }

// while
   int b=0;
   while (b<=10)
   {
       System.out.println ("\t \n" + b);

       b++;
   }

// do while
   int c=0;
   do
```

Q. ~~Display the~~

```
        {
            System.out.println("\t \n" + c);
            c++
        } while (c<=10);
    }
}
```

Q4

Display following numbers pattern by using for, while and do while

100, 90, 80, 70, 60 ... 0

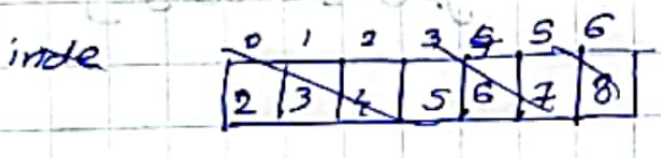
Code :-

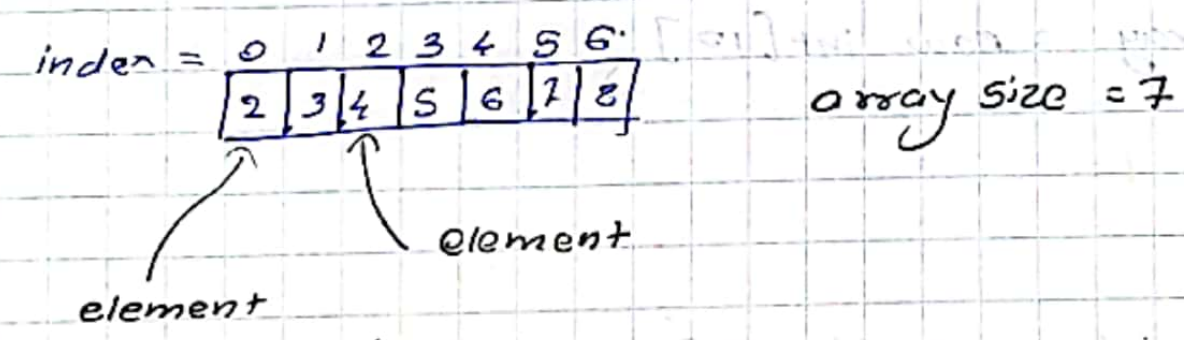
```
Class loop {

    public static void main(System.[] args) {

    // for loop

    ~~for(int a=0; a<=10; a-~~

    for(int a=100; a>=0; a=a-10)

     {
            System.out.println("\t \n" + a);
     }

    //while
     int b=0;
    while(b>

    int b = 100;

      while(b>=0)
      {
         System.out.println("\t \n" + b);
          b=b-10;
      }
```

```
// do while
  int c = 100;
  do
  {
     System.out.println("\t \n" + c);
     c = c - 10;

  } while(c >= 0);
```

Java array

- Arrays are capable of holding multiple erements in same data type.
- When declaring an array valid name and data type should provide
- giving the size of the array is not mandatory at the point of the declaration.





16/09/2022

Java Empty array declaration / free array.

- when declaring a free array 1st the data type should mention with [] s and it should follow with valid array name.

Syntax.

Data type [ ] valid arrayname ;

Ex :

String [ ] myarray ;

Declaring an array with fixed size.

- In java arrays can be created with a fixed size
- which mean values cannot be inserted more than the defined size to the array.

Syntax

- data type [ ] arrayname = new datatype [size] ;

Ex :

int [ ] array = new int [10] ;

Q,

Declare an int array size of 10 and take 10 user inputs to the array using a loop and display the values inside the array

I) find the maximum value insize the array
   I) using a forloop.

II) Display the above array values in reverse order

III) find out how many odd number and even numbers Contain inside the array

III Declare another int array size of 10 ~~modify~~ multiply the above array elements by 10 on assigne them to the new array and display the new array.

IV Sort the above Created new array in assending order

Code :-

```
package com.mycompluter.array;

import java.util.Scanner;

public class array {

    Public static void main (String [] args ) {

        Scanner objarray = new Scanner (System.in);

        int [] numbers = new int [10];


        //inputs

        for(int i=0 ; i<10 ; i++)
        {
            System.out.println ("Enter the element " +i " = " );
            ~~number~~ fio
            numbers [i] = objarray.nextInt();

        }

        //output

        for (int x=0 ; x<10 ; x++)
        {
            System.out.println ("\t \n" + Array numbers [x] );

        }
```

```
// max

int max = 0 ;

for(int z =0 ; z <=10; z++ )
{
    if (max < numbers[z])
     {
        max = number[z] ;
     }
}
System.out. println("The maximum number is

// reverse oder.

~~for(int =10 ; int~~

~~for(int =a~~

// reverse order

for(int a =9 ; a>0 ; a--)

{
  System.out.println ("\t \n" + number[a]) ;

}

// odd even count.

int even =0 , odd = 0 ;


for(int b =0 ; b <10 ; b++)

{
    if(numbers[b] % 2 == 0 )
     {
        even ++;
     }
     else
        odd++ ;
```

Eg51

```
System.out.println("Even number Count =" + even);

System.out.println("odd number Count =" + odd);
```

//new array

~~int [] numbersIle~~

```
int numberslle [] = new int [10];

~~for (a ++0 =~~
~~int a~~
for(a=0; a<10; a++)
{
    numberIle = numbers[10]X10;
}

~~for(d=~~

int numbr[] = new & int [10]

for (int a=0; a<10; a++)
{
    numbr[a] = numbers[a] * 10;
}

for( int e=0; e<10; e++)
{
    System.out.println("\t \n" + numbr[e]);
}
```

```
//sorting.

int size =10, passes =0;
int temp =0; size
for(int g =0; g<10; g++)
{
    for(int f =0; f<size-1; f++)
    {
       if(numbr[f]<numbr[f+1]
          {
          int temp.

          temp = numbr[f];
          number[f] = number[f+1]
          numbr[f+1] = temp;
       }
    }
    passes = passes + 1;
}
    passes = passes + 1;

for(int H =0; H<10; H++)
{
    System.out.println("\t \n" + numbr[H]);
}.
```

22/09/2022.

Q.

Create a java application to get the size of an int array from the user. and based on that Create an int array. get the user input to the array from ~~an~~ the user and find the maximum value which contain ~~insiz~~ inside the array.

Code :

```
package com.mycompany.chathu

import java.utill.Scanner ;

public class chathu {

    public static void main (String [] args) {

        Scanner input = new Scanner (System.In in)

        int size =0 ;

        System.out.println (" Enter the size of the array : ");
        size = input.nextIn();

        int array [] = new int [size];

        for (int i=0 ; i<size ; i++)
        {
            System.out.println (" Enter the elements : ");
            array[i] = input.nextInt();

        }

        for (int x = 0 ; ~~x<10~~ ; x<size ; x++).
        {
            System.out.println (" \t \n " +array[x]);

        }
```

```
//max.
int max = 0;
for(int z = 0; z<size; z++)
{
    if(max < array[z])
    {
        max = array[z];
    }
}
System.out.println("max is = " + max);
```

Java classes.

- Java class is a external file which contain in you java project.

- The class file is a blueprint for the objects
- By using a single class file n number of objects can be created, which helps to do the code reusability

Creating a java class.

- Java classes can be created with the key word class
- The boundary of the class will define by using curly brackets
- A class can be contain 2 members called
    1 fields / variables
    2 methods / functions

Structure :-

```
Class Classname
{
    //fields
    // methods
}
```

Ex

```
Class myclass {

    // variables can be declared → Global variable.

    public void mymethod()
        {
            //variable → local variable
        }
}
```

29/9/2022.

Java Encapsulation

- Encapsulation refers to hide data inside a single class but make available the value of the variable using methods implementation
- Data hiding is a way of restricting the access of our data members by hiding the implementation details.

Getters and Setters

- Get method will also a public methods which return the value of the private variable.
- Set method is again a public method which set some values to private variables.
- Set method does not return any value out of the function and most of the time it's a parameter method.

Ex. :-

```
private int sum;

public void setsum(int a)
{
   sum = a;
}

public void int getsum()
{
      return sum;
}
```

3.

Q.

Create a class called ~~encapsulation~~ encapdata.java and Create two private variables to store radius value and pi value.

Inside the main class you have to get the radius value from the users and pass it to ~~encap dat~~ encapdata.java class.

Inside the encapdata.java Class create getter and setter to find the area of the circle and to find the circumference of the circle.

Return the asw answer from the encapdata.java class display the answer.

Code :-

Encapdata class :-

```
package com.mycompany.encap ;

public class Encapdata {

    private double radius , pi = 3.14

    public void setrad (double rad)
    {
        radius = rad ;
    }

    public double getrad ()

    {
        return radius ;
    }

    public double getarea ()
    {

        return pi(a radius x radius ) ;
    }

    public double getcir ()
    {

        return 2 x pi x radius
    }

}
```

main method :-

```
impor java.util.Scanner;

public class encap {

        public static void main(String[] args)
        {
          Scanner objcir=new Scanner(System.in);

          System.out.println("Enter the radius value :");
          double r = objcir.nextDouble();

          Encapdata orad = new Encapdata();

          orad.Setrad(r);

          System.out.println(" Area of the Circle is :" +orad.getarea()

          System.out.println(" Circumference is :" +orad.getcir());
```

Inheritance

- In inheritance we are creating a new class by using an existing class.

- In such situation existing class will become the parent class or based class new class will become child class or derived class

- all the public, protected field and methods which contain inside parent class will available to child class.

- In java extends keyword will use to perform inheritance.

(psvm = shortcut to main method).

Syntax.

```
Class Animal
  {
    // methods and fields
  }

//use extends keyword
// to perfoom inheritance

  Class dog extends Animal
  {

// methods and fields of Animal
// methods and fields of dog

}
```

- you can customize the child class by methods and fields to the child class.

05/10/2022

Java Inheritance is-a relationship

- In inheritance can identify by using the keyword "is-a"
- ~~The keyword~~
- This keyword will use when explanning questions and class in computing.

Ex :-

Orang is a fruit.

- In above Orange will become the child class and fruit is a parent class

Q

Create a java application with two added class call Animal and Cat.

Cat is the derived class (child class) of Animal class (Based Class)

Inside the "Animal class" Create method inside the method print "I am an animal" Inside the "cat class" Create a method and display "I have four legs". Inside the main method Create a Object and display "I am an animal I have four legs"

Code :-

```
class Animal.

package com.mycompany.inheritance

public class Animal {

    public
    protected void Animaldetails()

    {
        System.out.print("Iam an animal");

    }
```

Class Cat :

```
com.mycompany,
package class Cart inheritance

public class Cat { extends Animal {

    protected void catdetails()
     {

       System.out.print("I have four legs");
     }
```

main class :

```
package class com.mycompany.inheritance

public class inheritance {

    public static void main(String[] args)
     {

        Cat objcat = new cat();
        objcat.Animaldetails();
        objcat.Catdetails();

     }

}.
```

Q_2 modify the above Programe by declaring a variable inside animal class and assigning an int value 4 to the variable. get the following print statement by using ~~modifying~~ modified application.

"I am an animal ~~and~~ I hav 4 legs"

code :

Animal class -:

```
package com.mycompany.inheritance

Public Class animal {
  protected int legs = 4 ;
  Protected void Animaldetails ()
  {
      System.out.println(" I am an animal ");
  }
```

Cat class :

```
package com.mycompany.Inheritance

public class cat {
{
      protected

      protected void catdetails( ) {
          System.out.print(" I have " + legs + " legs ");
      }
```

main class -:

```
Package com.mycompany.Inheritance

public class Inheritance {

      public static void main(String[] args) {

      Cat objcat = new cat()
      Objcat.Animaldetails()
      objcat.catdetails()
} }
```

Java Polymorphism.

~~In~~

- In OOP programming you cannot include two or more methods with same method signature inside a single class.
- In reason for that is when methods get call compiler will confuse with the method that you are ~~ty trying to~~ trying to call.

Ex :-

```
Class student
public void Studentmark(int M)
{
        =
}

public void studentmark(int y)
{
        =
}
```

method signature

- In above example bothe given methods are containing the same method signature because of that in OOP programming you are not allow to use it like that

- In Polymophism it describe how a single method can use in different formats.

- In OOP Polymophism can be use in 2 differents ways

  1 :- Methode Overriding.
  2 :- method Overloading.

Java method ~~overlo~~ overriding

- method overriding always ~~w~~ use with inheritance oop - concepts.
- Inside the parent class it's possible to declare the methods and inside the child class some method signature can be declared by overriding parent class.
- Since it use inheritance bothe parent class ~~head~~ methods and childclass methods available to child class.

Ex :

```
Class language {

Public void displayinfo ()
{
   System.out.print("Common language English language");
}
}

Class java extends eng language {
 {
   Public void displayinfo(){
   System.out..print(" java programming languag");
 }
}

main
Class main {

public static void main(String [] args ) {

 // calling language class.
      language L1 = new language();
      L1.displayinfo();

 // calling java class
      Java J1 = new java();
      J1.displayinfo();
   }
}
```

- In above example to call the superclass (parent class) method and object should create using superclass.
- To call the child class method on object should create by using child class name.

Java method overloading.

- In the polymophrism type we can contain n number of methods with same method name by differentiating the parameter list. This concept is known as method overloading.
- In this concepts same methods will perform different operations based on the parameter.

Ex :-

```
void func() {....}
void func(int a) {....}
float func(double a) {....}
float func(int a, float b) {...}.
```

07/10/2020.

Q Create a ~~two added~~ java application with 2 added classes called employee and manager. manage is a employee and inside the employee class create a method to get name and age as user input values. Override the same methods inside the manager class to get employee and manager details.

Code -:

Employee class :-

```
public class employee {
```

→

```
import java.util.Scanner;

public class employee {

    Scanner objemp = new Scanner (System.in)

    public void empdetails()
    {

      System.out.println ("Enter yo Employee name : ");
       int
       String name = Objemp.next();

        System.out.println("Enter ;: -loyee age : ");

        int age = objemp.next();

    }
```

manager class :-

```
import java.util.Scanner ;

public ~~no~~ class manager extends employee {

    Scanner objmng = new Scanner (System.in) ;

    public void empdetails()

    {

        System.out.println ("Enter the manager name : ");

        String name = objemp.next();

        System.out.println("Enter the manager age : ");

        int age = objemp.nextInt();

}
```

main class. -

```
Public class office {

    public static void main(String [] args)
    {
        employee emp = new employee ();
        emp.empdetails ();

        manager mng = new manager();
        mng.empdetails ();
    }

}.
```

java Constructors

- Constructor is a special method type which used in oop program -ming.
- Constructor doesn't have a return type and alway Constructor name should be class name.
- Constructor will execute at the point of object Creationsby using the Class name
- There are 2 types of constructors. in oop programming

    1 : Default Constructor
    2 : Parameter Constructor,

4

Default Constructor.

- There should only one default Constructor in ony given class.
- Defaut Constructors can have an access specifier and usually it's public ..

Parameterize Constructor

- There can be n numbers of parameterize Constructors in a single class by differentiating parameter list.
- Usually parameter constructors defined under public access specifiers.

Ex Default Constractor

```
Class manager
 {
      public manager()
      {
        // Can performance operations
      }
 }
```

parameterize Constractor

```
Class manager {
public manager(int age)
{
    //---
3.

public manager (string name , int age)
{

  //---

3
```

Calling the default Constructor.

- when Creating the class object from another class default Constructor will get call

Ex

```
manager Objmg = new manager();
```

Calling Parameterize Constructor.

- when Creating the Class object parameter should pass accordingly to get call Parameterize Contructor.

Ex manager objmg = new manager(45, 75);.

manager objmg = new manager("Darshana", 73.68);

Q Create a java application which contain class called student. Inside the student class there should be a variable to store student name Inside the default constructor initialise student name as alex.

- add a parameterise Constructor to the same class which ~~pare~~ passes string value to the constructor.
- Display the student name by using both constructors.
- Create the object accordingly and call both constructors to display the name.

Code :

```
public class student {

    public student ()
    {
        String name = "alex";

        System.out.println (name);
    }

    public student (String name)
    {
```

```
        System.out.println(name);
    }

Public main class

public class con {

    public static void main(String[] args)
    {

        Student objs = new Student();

        Student objst = new Student("Darshana");

    }.
```

13/10/2022

Q. Create a java application which contain a class called "Student" with three Constructor as follow

* Default Constructor which take student name as an user Input
* Parameterized constructor which student id (int) inside the Student class variable
* Parameterized Constructor which override Student name

Run the application in the mainclass.

Code :

→

Student class :-

```
import java.util.Scanner;

public class Student {

    ~~publ~~
      Scanner input = new Scanner(System.in);

      public student () {

      System.out.println("Enter student name : ");
      String name= input.next();

      System.out.println("Student name is : " +name);

      }

  ~~public class~~

      public Stud student(int id) {

      System.out.println("Student id number is : " +id);

      }

      public student (String name){

      }
        System.out.println("Student name is" +name);
```

main class :-

```
public class abstract {

        public ~~void~~ static void main(String [] args)
        {
            Stude
              student obst = new student();
              student obst1 = new Student(2751);
              student obst2 = new Student("Darshana");
        }
}
```

Java abstraction.

- In this oop concept we are hiding the implementation details of a method and showing only the functionality of the user.

for an example.

when sending a message only bother about sending and receiving messages rather than worring about the backend process. In abstraction simmilar kind of thing happends

Abstract classes and methods.

- In oop programming abstraction can be acheived by using abstraction Abstract Classes or interfaces. when implementing abstraction a keyword call **abstract** will use which is not access specifier this keyword will users in Classes and methods

Abstract class.

- Abstract class is a restricted class which cannot be access by other classes using class of objects
- Abstract class should have atleast one abstract method.
- an abstract Class can also Contain non abstract methods also
- when accessing abstract class. It should access using inheritance

Abstract method.

- Abstract method does not have implementation. In abstract class abstract methods are only defineing
- when defining abstract method accessability level and return type Can be decieded.

- The body of the abstract method will be written inside Childclass using method overriding concept

Ex -:

```
abstract class Animal {

    public abstract void Animalsound ();

    public void sleep ()
    {
        System.out.println ("z z z ");
    }
```

- Using the above animal class we are not capable of creating a class object since animal class is a abstract class.

Ex -:

```
// abstract class

abstract class Animal {

    public abstract void animalsound ();

    public void sleep()
    {

        System.out.println ("z z z ");
    }
}

// a child class.

class pig extends Animal {

public void ~~Anim~~ animalsound ()
 {
     System.out.println("Mr. pig says wee wee ");

     }
}
```

```
// main class

public class abstraction {

    public static void main (String [] args)

    {
        pig objpig = new pig();

        objpig.sleep();
        objpig.Aanimalsound();
    }
```

Q add an abstract class method to the animal class call animalweight overwrite the method inside subclass display animal weight as 50 kg.

Ex :-

```
Abstract class animal {

abstract
public abstract void animalsound();

public void sleep()
{
   System.out.println("z z z");
}

public abstract void animalweight();

}

Class pig extends animal {

public void animalsound()
{
    System.out.println("The pig say wee wee");
}

public void animalweight() {

   System.out.println("weight is 50 kg");
}
}
```

```
public class abstraction {

    public static void main (String [] args)
    {
        pig objpig = new pig();

        objpig.animalsaupnd();
        objpig.sleep();
        objpig.animalweight();
    }
}.
```

Q Try to create an object by using abstract class name and check the posibilities.

- when creating class objects by using abstract class an application is throwing a syntax error. Because of that Creating object by using Classes are not possible.

Q2 Create an abstract class call employee an add abstract method will call employeeinto manage is a employee and inside monager class override the abstract class method and get manager's name as a input display the manager's name inside the method.

code :-

```
// abstract class

  abstract class employee {

      public abstract void employeeinfo();
  }

// subclass

    ->
```

```
import java.util.Scanner (*System.in);

class manager extends employee {

    Scanner objmng = new Scanner (System.in)
    {

      public void employeeinto ()

      {
          System.out.println("Enter the name : ");
          String name = objmng.next();

          System.out.println("The manage name is : "+name);
      }
}

//main class:

    public class abstraction {

        public static void main (String[] args)

        {
            manager.mng = new manager ();

            mng.employeeinf();

        }
}
```

14/10/2022.

Java Interfaces

- Interface is another way of implementing abstraction in java
- Inside an interface we can declare methods but method doesnot have implementation. which mean an interface is complete abstraction class.

Example :

```
interface Animal {
    public void Animal ( );
    public void run( );
```

above mention methods will automatically becomes abstract method. Since it did not contain any implementation.

- To access the methods inside the interfaces another class should added and that class should implements from the interface by using the keywood "implements" keywood will use.
- The body of abstract methods should provide inside implement Class.

example:

```
//interface.
interface Animal {
    public void animalsound ( ); //interface methode dosn' doesn't haveb
    public void sleep ( );
}

//pig "implements" the animal interface
class pig implements Animal {
    public void animalsound ( ) {
        System.out.println ("The pig say wee wee");
    }
```

```
Public void sleep() {
    System.out.println(" z z z ");
}

Class main {
    public static void main(String [] ags) {
        pig mypig = new pig();
        mypig.animalsound ();
        mypig.sleep ();
    }
```

Q modify the above program by adding abstrace method call "animalshape" and get user inputs the ~~animal~~ color of the animal and display it in the abstract method implementation.

2 Create an abstract method. Call animalfeet () which take an int parameter that is number of feet which animal contain. Get the user input from the user & display the number of feets inside the abotract method.

Q modify the above program by adding an abstract method Call Animalage which take age as a int parameter from there method check whether animal is an adult or child and return 1 or 0 accordinly from the main methode check the whether the return value is 1 and display output as an adult. If the return value is 0 output is child.

Code :-

Interface Animal :-

```
public interface Animal {

    public void animalsound ();
    public void sleep ();
    public void animalshape ();
    public int void animalfeet (); (can.
    public int animalage ();
}
```

```
public interface Animal{

    public void animalsound();
    public void sleep();
    public void animalshape();
    public void animalfeet(int feet);
    public void int animalage(int age);
```

Class pig -:

```
package com.mycomputer.interface;

import java.util.Scanner;

public class pig implements Animal {

    Scanner sc = new Scanner(System.in);

    public void animalsound()
    {

        System.out.println("The pig says woe wee");
    }
```

```
public void sleep()
{
    System.out.println("Zzzz ");
}

public void animalshape(in)
{
    System.out.println("Enter the animal color : ");
    String color = objpig.next();
    System.out.println("The color of the animal is : "+color);
}

public void animalfeet(int feet)
{
    System.out.println("Animal have " +feet+ "feets ");
}

public int animalage(int age)
{
    if(age >= 18)
    {
        return 1;
    }
    else
    {
        return 0;
    }
}
```

main class

->

~~public class interface~~

```
import java.util.Scanner;

public class interface {

    {
      public static void main (String [] args)
      {
          Scanner input = new Scanner(System.in);

          pig objp = new pig();
          objp.animalsound();
          objp.sleep();
          objp.animalshape();

          System.out.println("Enter the foets animal have : ");

          int ft = input.nextInt();
          objp.animalfeet(ft);

          System.out.println("Enter the age of the animal :");

          int a = input.nextInt();

          if(objp.animalaget(a)==1)
            {
               System.out.println("Animal is an adult");
            }

          else
            {
               System.out.println("Animal is a child");
            }
```

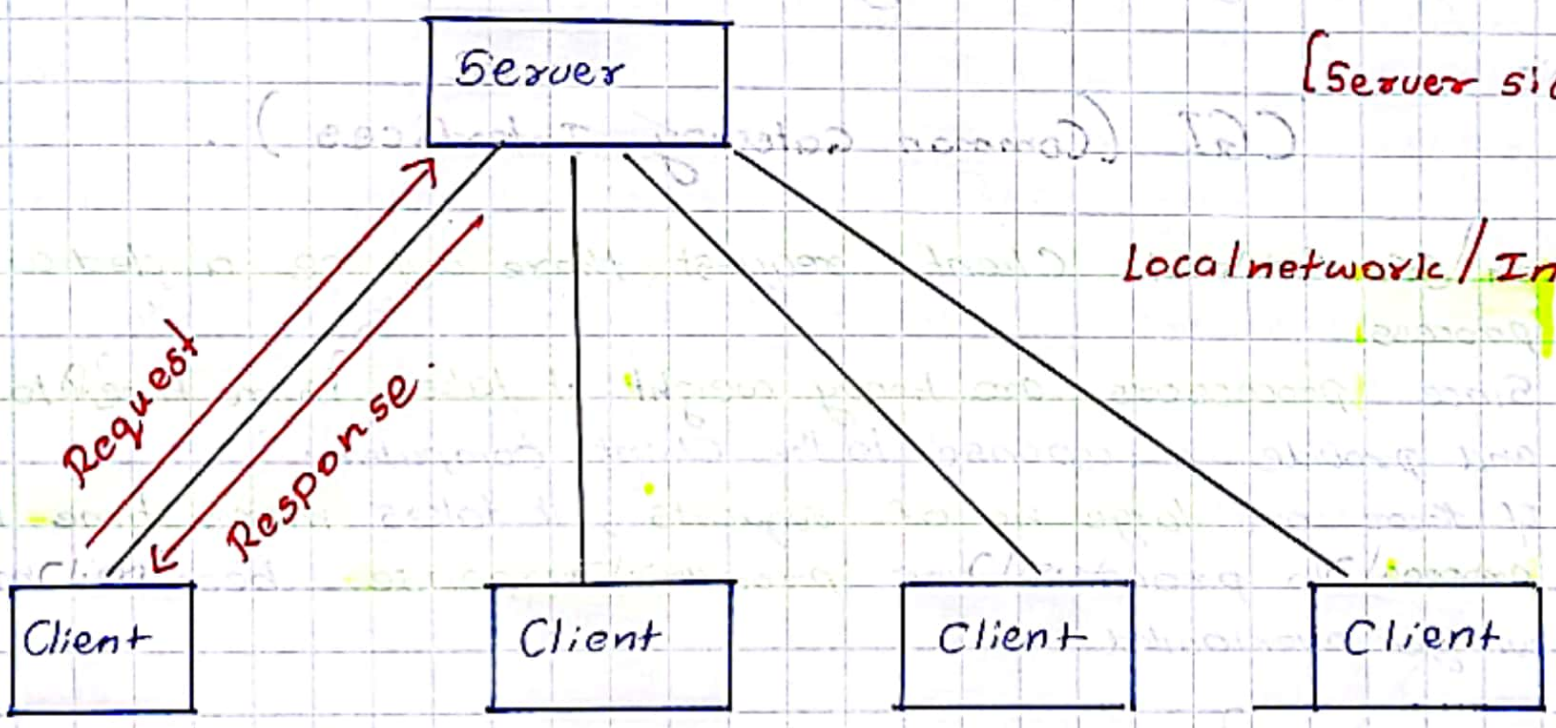
19/10/2022.

Java Servlet

- Java servlet are working based on client server architecture
- which means servlet are reside server side.

Client - Server architecture.

- This is a software architecture which used in application development
- In this architecture there are two main components called Client and Server.
- Server is a dedicated device which contain all application data and Clients are connecting to the server ~~side~~ and making request from from the server.
- In this architecture Clients are making request to the server and server will process them Then respond back to the client Computer over the network (internet).
- In Java servlet it discuss about a technology called Server side programming / Server side Scripting.



Java servlet.

- Servlet is a technology which used to do Serverside programming as a serverside Scripting language to Create a dynamic webpage
- Servlet technology is using Java programming language because of that this is a robust (the capacity of a Computer system to handle the errors during executing and manage the incorrect input of data) and Scalerbility (an application

can deal with more and more users by changing the hardware Configuration rather than the application itself) technology.

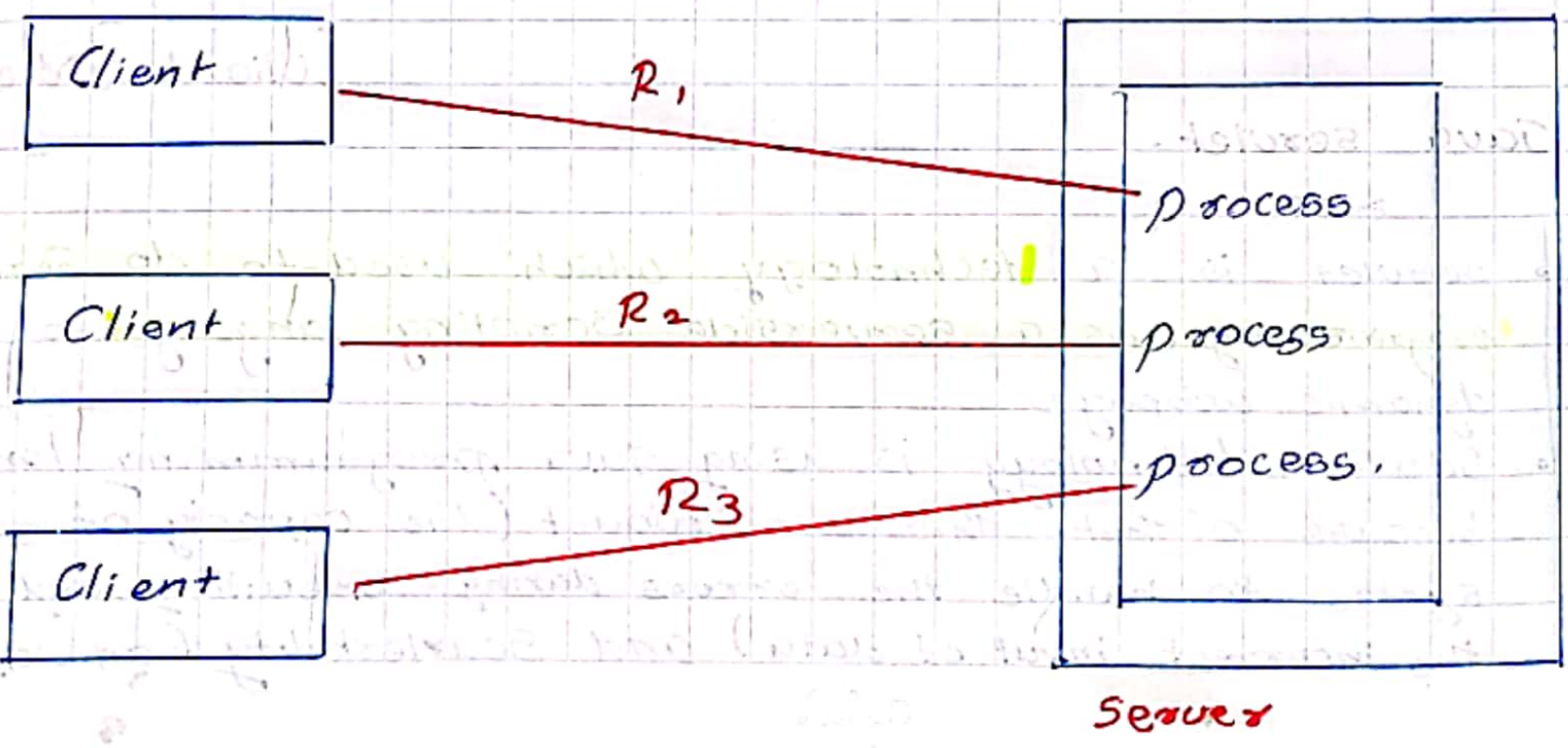
- Before Servlet a technology called Common gate way interface [CGI] is used CGI is a server side scripting language which has identicaly drawbacks

what is a Servlet?

- Servlet are been used to create dynamic webapplications because the content of dynamic pages are different from user to user.
- Servlet will do the configurations based on the user with help of database and will crecite dynamic webpages in Client side.
- The main advantage of Servlet is we cannot configure specific hard coded webpages as specifically user to user Servlet will help to do dynamic configurations for specific user when creating dynamic pages.

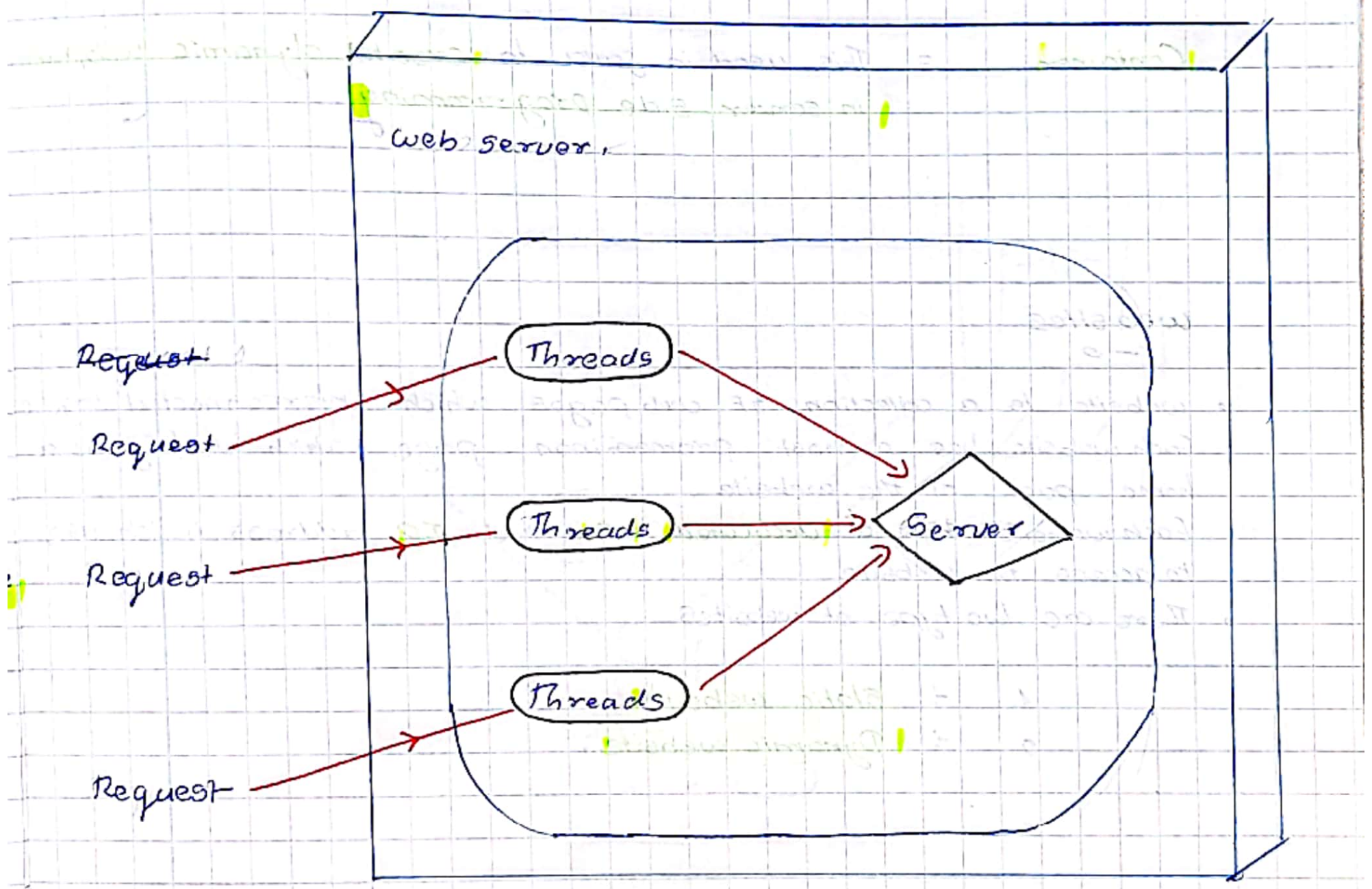
CGI (Common Gateway Interfaces).

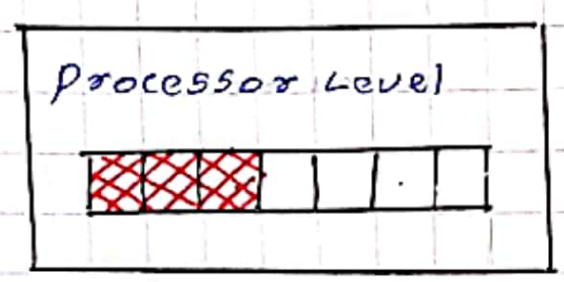
- In CGI for each Client request there will be a dedicated process
- Since processors are heavy weight it takes more time to execute and provide a response to the Client computer.
- If there are large no of requests, it takes more time to process to process and provide response because memory will get overloaded.



Advantages Servlet

- In servlet @ each request will convert as a thread inside the server
- Since threads are lightweight and easy to process threads will execute quicker than processors and provide response back to the client.
- Since threads are lightweight server won't get over loaded even though ~~members~~ numbers of clients get Connected.





web Technology

HTTP = The protocall used to establish Connection between Client and Server.

- In modern day HTTPS protocall also used to for the secured connection between Client and Server.

HTTP Request - This is the request which send by Client to Server by Containing all relevent information.

Container = This used in java to create dynamic webpages in server side programming

web sites

- website is a collection of web pages which interconnected together
- Each website has a most prominion page which identify as a home page of the website.
- Each website has a dedicated URL and IP address which use to access the website
- There are two type of websites.

1 :- Static website
2 :- Dynamic website.

CMD command to find IP address. = tracert

=

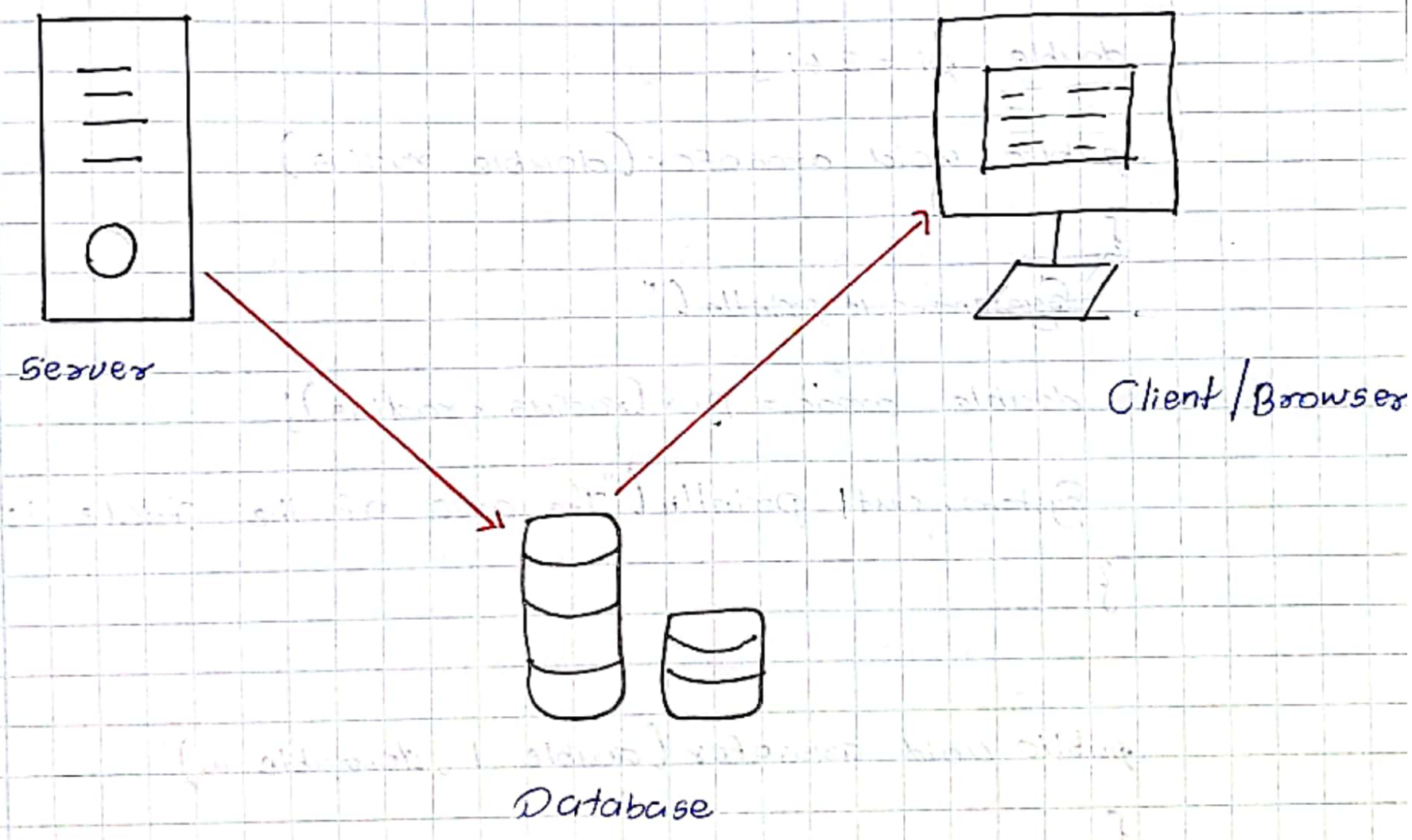
Static websites

- The content of the webpage will not change from user to user when using static webpsites.
- Static pages are directly connected with servers and retriew dat

Dynamic websites

- Information of dynamic website will change from user to user
- This differenciation has been done by using a data base or Content management system [CMS]

Dynamic website



Past Paper Questions

Q. Create a java class to calculate and display the area of shape

a. Create two methods to calculate the area of a rectangle and a circle using methode ~~override~~. Overloading

b. Display values.

Code :

Class Shape

```
public class cShape {

    double pi = 3.41;

    public void areaofcr(double radius)
    {
        System.out.println("
        double areac = pi * (radius * radius);

        System.out.println("The area of the circle is :" + areac);
    }

    public void areaofcr(double l, doouble w)
    {
        double Greaofr

        double arear = l * w;

        System.out.println("Rectangle area :" + arear);

    }
```

main methode :

```
import java.util.Scanner;
public class main {
    ~~static void public void main (String args [])~~
    public static void main (String [] args)
    {
        Scanner input = new Scanner (System.in);
    §
    Shape cr = new Shape();
    System.out.println("Enter the radius of the Circle :");
    double r = input.nextDouble();
    cr.areaofcr(r);
    System.out.println("Enter the length of the rectangle :");
    double length = input.nextDouble();
    System.out.println("Enter the width of the rectangle :");
    double width = input.nextDouble();
    cr.areaofcr(height length, width);
    }.
```

Q Create a class called Animal

a. Implement 2 methods name 'make-sound' and "Move"

b. Create 3 subclasses extending the Animal class called Cat, Fish, and Bird.

C. Assign unique methode which are specific to it's own class.

d. Create Objects from the external

2/11/2022.

HTTP Request.

- In client server application Clients are making request is known as HTTP Request.
- This request is transmitting : over the network and in realworld Scenarios there are multipled requests are recieved by the Server
- Since the request follows HTTP Protocall the behaviour of this request will define by the Protocall

- HTTP request contain multiple informations among those following three ⁰ should be assential.

1. Request line
2. • This Contain the information which requested by the server computer.

2. Source IP address port.
   • Since there are multiple Clients Connected to the server server should identify to which Client it should response
   • Since Client computer also capable of running 2 seperate applications server should know to which application it should response from the port number can be identify.

3. Destination IP address and Port number.

- This will define to which Server the request Shout Should made IP address of the Server will define the Server location of the network.
- In Some situations the Server might Contain number of applications in that situation destination Portnumber will define to which application the request Should made.

HTTP Request Methods

- when Creating form applications we have to include an attribute such as action, method
- Action will define to which Page the submitted data Should transfer
- when transfering those data it useed a rule set under HTTP protocall
- Request method will define how the transaction Should happen.
- In HTTP ~~theore~~ there are multiple request methodes ~~will~~ such as GET, POST, HEAD, TRACE etc.
- Above request methods are ~~hiding~~ having different level of Configurations. Among thos GET and POST request are commonly use.

Get GET or POST Request.

- Get request will display all submited data on the URL which is not a Secured methode Since all Submitted data can monitor through the URL history.
- Post Request methode will not expose the Submitted data since it Cannot be monitered later because of that Post is a secured method.
- In Client Server web development GET methode ~~will~~ also used under different requirements.

JSP Pages

- JSP is a technology to create front end of a web application. JSP is capable of containning HTML, JSP tage.
- JSP page will connect with Servlet which contain the logical program of the web application.
- By using JSP expressions java code can also include inside it.
- Theoritacically JSP page will execute as a client application and Sesulets will execute as the Server application.

Q

Nu	Name	Subjects
01	Thon	SE2
02	Sam	DB
03	Alan	Math.

HTML Code :-

```
<HTML>
<html>
  <head>
    <title> Servlet </title>
  </head>
  <body>
    <table>
```

```
<table border = "1">

<tr>

    <td> Nu </td>
    <td> Name </td>
    <td> Subjects </td>
</tr>

<tr>

    <td> 01 </td>
    <td> Thon </td>
    <td> SE2 </td>

</tr>

<tr>
    <td> 02 </td>
    <td> Sam </td>
    <td> DB </td>

</tr>

<td>
<tr>

    <td> 03 </td>
    <td> Alan </td>
    <td> Math </td>

</tr>

</table>

</body>

</html>
```

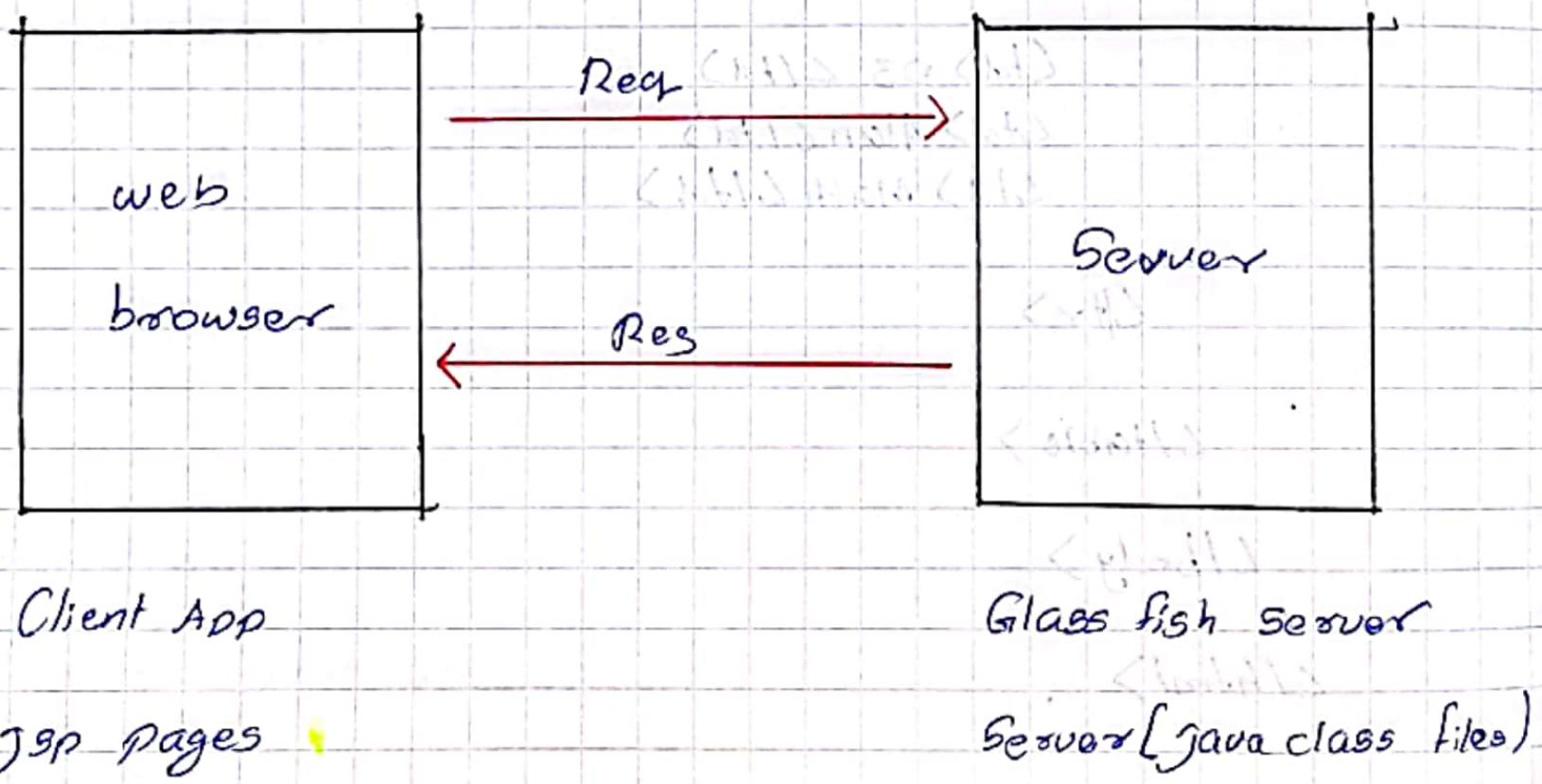
* Virtual Servers

- This will help to create a client server environment inside the IDE.
- Java web application Consist with JSP pages and Servlets
- when execute a java web application JSP Page will loaded into web browser and Servlets are execute in Virtual Server
- In java web there are two main servers known as

~~1. Glass fish~~

2. Glassfish Server
2. Apache tom cat Server

- when Configuring the projects you have to select which server that you use.
- Depending on the select server Servlet will get loaded to the virtual Server and Creates a client server environment.



- In Industrial application development Server will be located in a remotea place and it will transtact data. using a network

- In IDEs Server and the jsp files are located in ~~the~~ Same device and execute inside the local host.

09/11/2022.

Q Create a java web applictio-ation to display your name, batch and degree program in three seperate line.

Code :

```
<html>

 <head>

    <title> java </title>
 </head>

 <body>

    Edi
    <h1> Darshana weerasooriya </h1> <br>
    <h2> 21.1 </h2> <br>
    <h3> Software engineering </h3> <br>

 </body>

</html>
```

Q Create a java with 2 pages and link both pages over a tex in the ~~ba~~ Page body.

code :

```
<html>

 <head>

    <title> java </title>
 </head>
```

```
</body>
    <a href = "index.html"
    <a href = "J1.jsp"> Clicle me </a>
</body>
</html>.
```

Q Create a java web application using given user Controll in the whiteboard.

Sign up.

Name :

Age :

Address :

ID Num :

Submit | Cancel

Code :

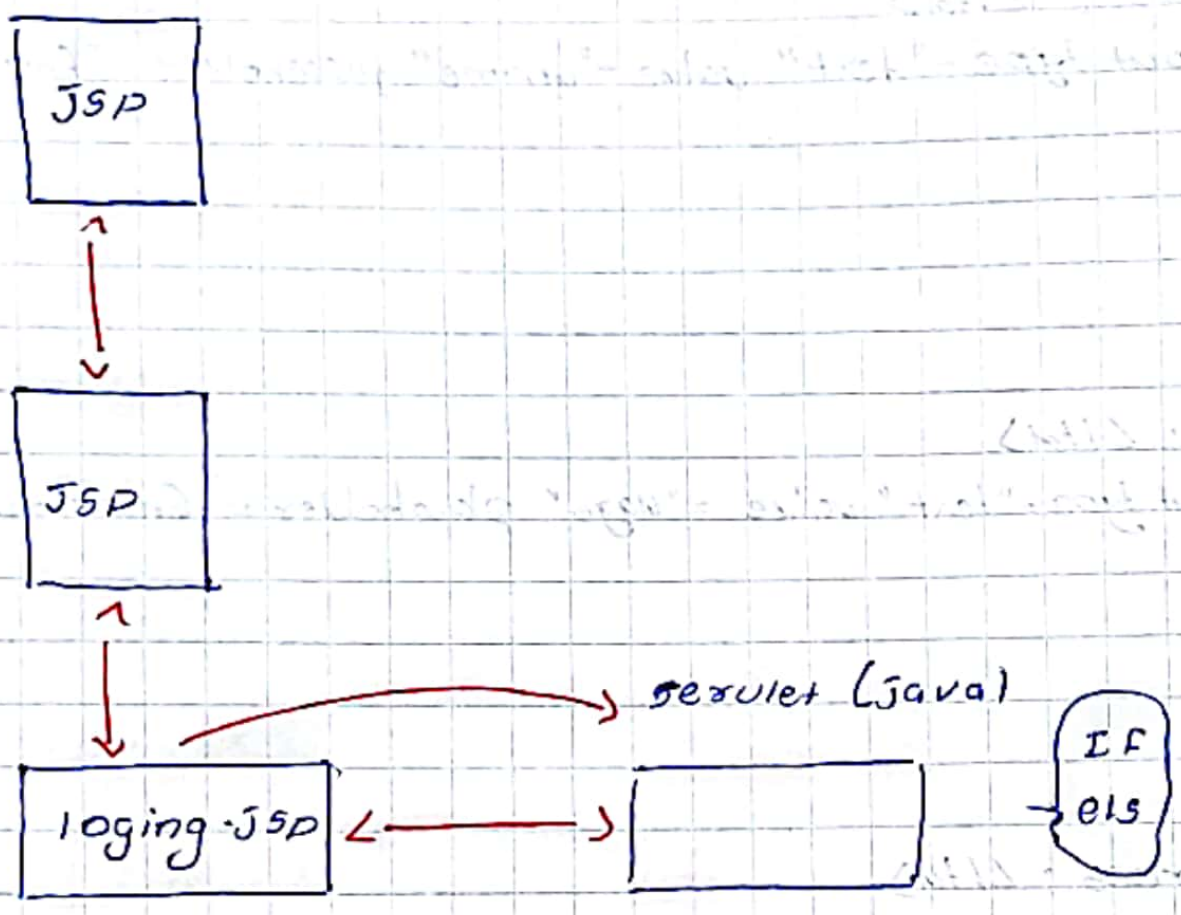
```
<html>
    <head>
        <title> java </title>
    </head>
    <body>
        ->
```

```
<form method="post" action=" ">

<h1> Sign up </h1>

<table>

  <tr>
      <td> Name : </td>
      <td> <input type="text" value="uname" placehol

</tr>

<td> Name : </td>
<td> <input type="text" value="uname" placeholder="Enter the name">

</tr>

<tr>

<td> Age : </td>
<td> <input type="text" value="uage" placeholder="Enter the age">

</tr>

<tr>

<td> Address : </td>
<td> <input type="text" value="uadd" placeholder="Address"> </td>

</tr>

<tr>

<td> Id Number : </td>
<td> <input type="text" value="uid" placeholder="id number"> </td>

</tr>

<tr>

<td> <input type="Submit" value="Submit"> </td>
<td> <input type="reset" value="Cancel"> </td>

</tr>
```

~~Lable~~.

Labels
Textbars
Buttons
} dala handeling.

JSp Servlets



- Servlet is a java class file which can be use to do backend data handeling

- web application might Contain number of jsps frontend files but all of them might not required a backend data processing.

- Create a Servlet inside the web application if only it use back end process

- ~~In which~~

- which mean all front end Pages should not ~~request~~ required a servlet Page

- Since servlet is a java class file it Contain member variables and member methods.

- When Creating a servlet there are 3 inbuilt methods.

1 processRequest ()

when servlet is not handeling data this methode will revoke.

2 doGET ()

This method will invoke once we use GET as the HTTP Request method as the form method. If you use get this will activate.

3 doPOST ()

This methode will activate once we use POST as the method request.

- To display the Contain on the servlet Printwritter object refference should be Create.

```
PrintWritter out = response.getwritter();
```

- In above statement out is an object refference which can use any valid name.

- Using the object refference with the help of Print inbuilt methods Contain Can be display on servlet

```
Out.Print("Hellow I am Servlet");
out.print("This is Config Servlet");
```

- Inside the servlet there should be an appropriate variable to extract frontend from data to backend servlet
- Using the controoller name which define in frontend each texbox value can be extracted.

```
String Name = request.getParameter("username");
```

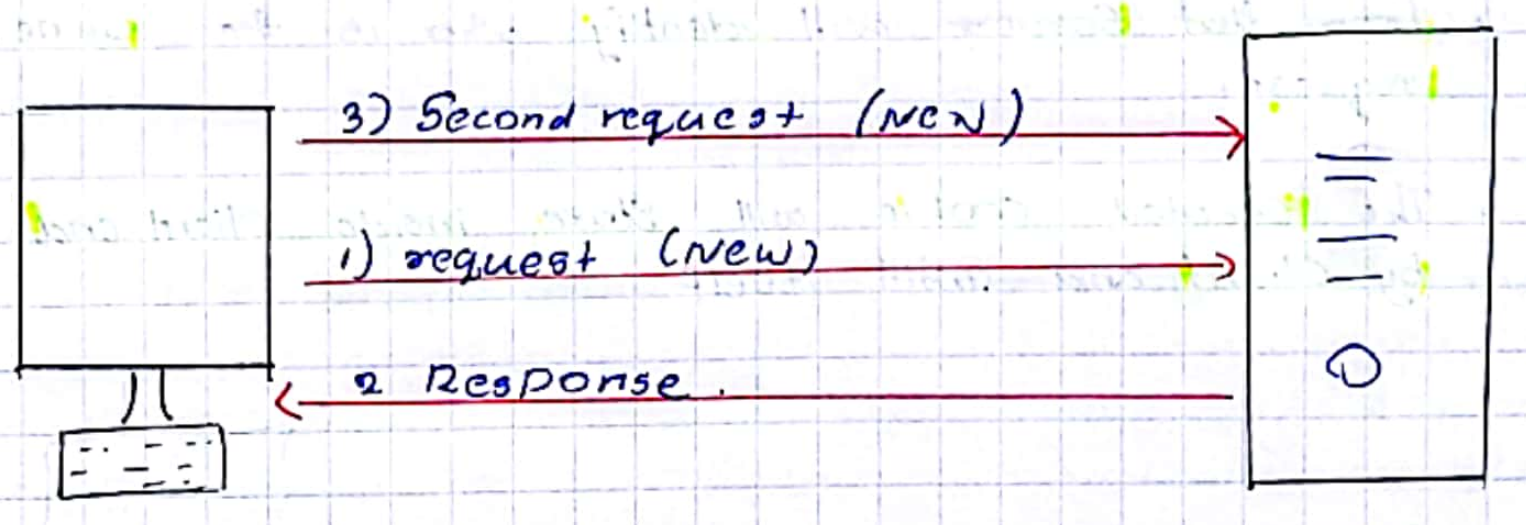
- In above code using request.getParameter inbuilt method will extract data from text box controoller to the variable

23/11/2022.

Session tracking in Servlet.

- In web applications & storing data related to Clients are important to run web applications
- In advantage of remembering this information is to run the webapplication smothly with low response line
- In HTTP protocol all request which send to the server will identify as new request.
- which means Server should know the user detail from request to request. This happens since HTTP is stateless protocol [Stateless :. Not maintainning data of a user]
- Due to that reason using Sessions tracking technologie we maintain user information throughout the web applicatio

Bellow diagram explain how HTTP Protocol transact (communicate) with server



- There are 4 main technologies are beign use for session tracking

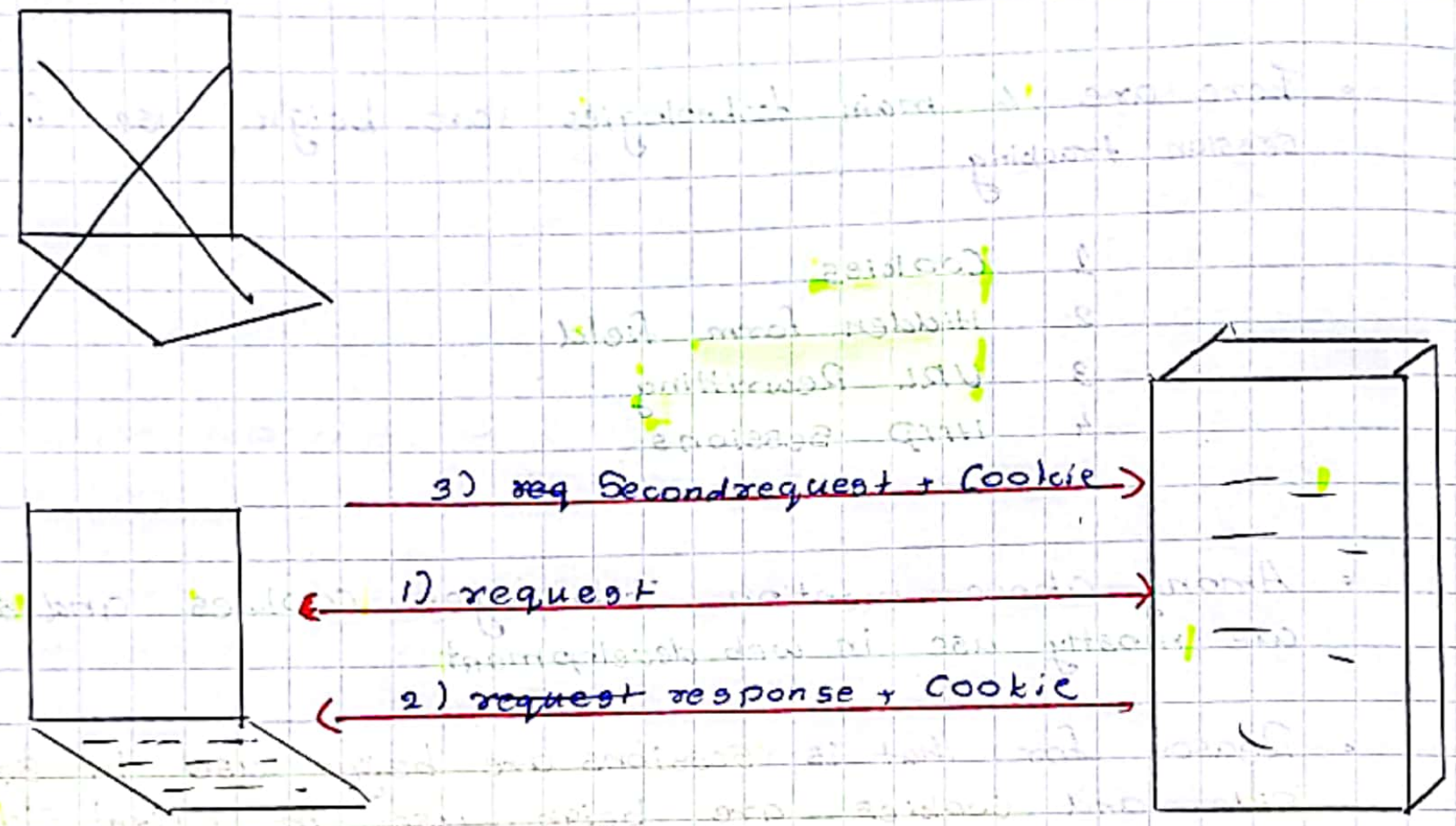
1. Cookies
2. Hidden form field
3. URL Rewritting
4. HTTP Sessions.

- Among above mention Technologies Cookies and sessions are mostly use in web development
- Reason for that is sessions are beign use in Server side and Cookies are beign use in Client Side.
- for verification Purpose both of these technologies are beign use together in web development.

Cookies in Servlet.

- When Client sending the request to the Server will gener generate a set of information which known as a Cookie.
- When Server responding back to the Client the Cookie will append (bind / merge) to the response and send back to the Client.
- When Client requesting again from the Server, Created Cookie will also get attached.

- from that Server will identify who is the owner of this request.
- The Created cookie will Store inside Client end untill destroy by Client ~~end untill~~ itself



- The main advantage of a Cookie it is in the Client end with less memory Consumtion.
- The main disadvantage is if Client disable the Cookie it wont exsist anymore.

Java Database Connections (JDBS)

- Java web application are capable of Connecting with different type of databases such as mysql, oracal oracle etc
- when Connecting web application with the database A relevent Connectors should ~~Add~~ add to the web application accordingly
- These Connectors ~~are~~ or will Create the medium to communicate with database from the application. Note thate Connectors will change based on the database.
- These connectors are existing as .jar files which we should include to the Project library.
- when Configuring the Servlet we should import

  import java.sql.*;

  to get inbuilt Connection classes.
- once we use the imported library multiples classes will available for the Servlet file
- Out of ~~those~~ those classes Connection and Statement Classes are beign use to Create instant and initialize with a null value at the begining.

ex

```
Connection Con = null;
Statement  St  = null;
```

- when connecting 2 different applications (front end and backend) there might be exceptions because of the exception handeling should use in JDBC

- Added jar file should invoke in backend Program using following code

```
Class.forName("com.mysql.jdbc.Driver");
```

- By using DriverManager ~~inbuiltt~~ inbuilt class your application should inform to which database that you are Connecting

- The URL of the database should pass as a parameter along with device mysql username and Password.

```
username = "root"
password = " "
```