

<CELL padding=" " >

The cell padding in HTML is an attribute for the table element (used for td and th elements). It defines the space between the cell edges/walls and its content all padding is an inline attribute used under the table tag that can override the CSS style <table cellpadding , "20">

```
<html>
  <body>
    <table border="20" cellpadding="4">
      <tr> <th> NAME </th> <th> </th> <th>
      <tr> <td> aaa </td> <td> 111 </td> </tr>
      <tr> <td> bbb </td> <td> 222 </td> </tr>
    </table>
  </body>
</html>
```

<colspan "NO of row/column >

The colspan attribute in HTML specifies the number of columns a cell should span it allows the single table cell of to span the width of more than one cell or column it provides the same functionality as merge cell in a spreadsheet program like excel.

```
<html>
  <body>
    <table border="1">
      <tr>
        <th colspan="2" > INFO </th>
      </tr>
```

```
<tr>
<td> aaa </td>
<td> 111 </td>
</tr>
<tr>
<td> bbb </td>
<td> 222 </td> </tr>
</table>
</body>
</html>
```

`<rowspan="No.of Row/col">`

the rowspan attribute in HTML specifies the number of row a cell should span. that is if a row spans two row it mean it will take up the space of two rows in that table. it allow table cell to span the height of more than one cell or row it provides the same functionality as "merge cell" in the spread sheet program like Excel.

Usage :- It can be used with `<td>` and `<th>` element in HTML Table.

```
+td :- <td rowspan="value"> table content ... </td>
+th :- <th rowspan="value"> table content ... </th>

<html>
<body>
<table border="1" cellpadding="40">
<tr>
<th rowspan="3">INFO </th> </tr>
</tr><tr> <td> aaa </td>
<td> 111 </td> </tr>
<tr> <td> bbb </td> <td> 222 </td> </tr>
</table>
</body> </html>
```

ham particular row <td>, <tr>, <th> ke
corresponding bgcolor tag ka use kar sakte hai

```
<html>
<body>
<table border="1" bgcolor="red">
<tr bgcolor="blue"> <td>
<td> 111 </td> </tr>
<tr bgcolor="cyan"> <td>
<td> 111 </td> </tr>
</table>
</body>
</html>
```

* अदि टिप्पी के अंदर हम categories बनाती हैं तो :-

```
<html>
<body>
<ol>
<li> php <li>
<li> python <li>
<li> java
<ul>
<li> J2SE <li>
<li> J2EE <li>
<li> J2ME <li>
</ul>
<li> css <li>
<li> js <li>
</ol>
</html>
</body>
</ul>
```

dl :- definition list
dd :- definition discussion
dt :- definition table

```
<html>
<body>
<dl>
  <dt> java </dt>
  <dd> 100 line definition </dd>
  <dt> python </dt>
  <dd> 100 line definition </dd>
</dl>
</body>
</html>
```

<td> by default left side se print karta hai

Δ <th> center karne ke liye <th> ka use karte hai

```
<html>
<body bgcolor="cyan">
  <center>
    <form action = "lg4">
      <table border="1" cellpadding="12" cellspacing="0" style="width: 100%; border-collapse: collapse; margin-left: auto; margin-right: auto;">
        <caption> LoginPage </caption>
        <tr>
          <td> Enter Name </td>
          <td> <input type = "text" placeholder = "Enter Name .." name = "u1" > </td>
        </tr>
        <tr>
          <td> Enter Pass </td>
```

```
<td> <input type="password" placeholder="Enter pass">  
      name="v2"> </td>  
</tr>  
<tr>  
<th colspan="2"><input type="submit" value="Login" />  
</th>  
</tr>  
</table>  
</form>  
</center>  
</body>  
</html>
```

① class demo

```
public static void main (String ar[])
{
    String s1=new String ("Ram");
    StringBuffer sb=new StringBuffer ("ram");
    System.out.println ("s1==sb");
}
```

② class demo

```
public static void main (String args[])
{
    boolean x=true;
    int y=20;
    System.out.println(x==y);
}
```

③ class demo

{
public static void main (String args)

{
int x=10;

OP- error: incom-

String x1 = new String ("ram"); comparable type:
System.out.println (x==y); int and string

3 3

④ class demo

{
public static void main (String args)

{
boolean x=true; OP-error: incomparable type
String y="ram"; int and string

System.out.println (x==y);

3 3

⑤ class Demo

{
public static void main (String args)

{
Object o=new String ("ram");

Object sb=new StringBuffer ("ram");

System.out.println (o==sb); OP- false

3 3

⑥ class demo

{
public static void main (String args)

{
Object s1=new String ("ram");

Object s2=new String ("ram");

Object sb=null;

System.out.println(s1==sb); O/P- false

3

Junior Java - 10

CSS (cascading style sheet)

- CSS stands for cascading style sheet
- CSS describes how HTML element are to be displayed on screen, paper or in another media
- CSS saves a lot of work - It can control the layout of multiple web page all at once.
- External StyleSheet are stored in css files.

Type of CSS:- There are three type of CSS are:

- Inline CSS
- Internal or Embedded CSS
- External CSS

i) Inline CSS - {<p>—</p>} मात्र ऐसे single, single
<p>—</p> आरे particular tag में
CSS का use करता है तो inline CSS का use करता पड़ेगा।
(single line ke purpose se inline CSS ka use करते हैं)

" Inline CSS contains the CSS property in the body section attached with element is known as inline CSS. the kind of style is specified within an HTML tag using the style attribute."

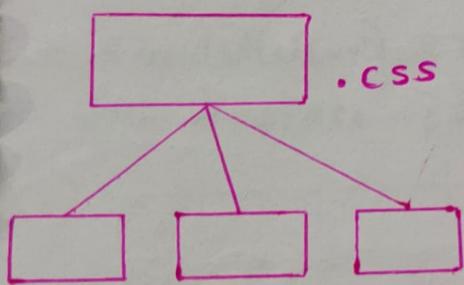
ii) Internal CSS -:: ek page me kaiseisme 10 line me change karna hai to internal CSS ka use karege। (single page ke purpose se internal CSS ka use

कहते हैं, हमारे page में कितने भी paragraph हैं।
उन सब में CSS proper work करना बहुत ही

"this can be used when a single HTML document must be styled uniquely. the CSS rule set should be within the HTML file in the head section i.e. the CSS is embedded within the HTML file."

iii) External CSS :- पर्याप्त हो में more than one page में CSS apply करना है तो हमें external CSS का

use करना पड़ेगा।



इसे एक page बनाना पड़ेगा, CSS नाम से उसकी link वह page से मरनी पड़ेगी फिर इसमें change होंगे तो सब में change होगा।

"External CSS contains separate CSS file which contains only style property with which contains only style property with (for e.g. class, id, heading etc).css"

the help of tag attribute (for e.g. class, id, heading etc) .css property written in a separate file with .css extension and should be linked to the HTML document using link tag. this means that for each element, style can be set only and that will be applied across web pages."

21/08/23

Monday -

inline CSS Syntax:

- i) `<p style = "p: v"> </p>`
- ii) `<p style = "p1: v1; p2: v2; p3: v3;"> </p>` more
than one property के लिए semicolon (;) से separate
karna padega.
- iii). `<p style = "p: v1 v2 v3"> </p>`
अगर एक ही property को space से ज्यादा बार use करना है
तो space से separate karna padega!

NOTE :- Text align property values :-

- i). left side से proper data aayega.
- ii). right : right side से data proper aaye ga.
- iii). center : data center से show hoga.
- iv). justify : books में जैसे रहता है left और right dono
side से data proper show hoga.

①

```
<html>
```

```
  <body>
```

```
    <p align = "center"> ram </p>
```

```
    <p style = "text-align: center"> ram </p>
```

```
    <p style = "text-align: left"> ram </p>
```

```
    <p style = "text-align: right"> ram </p>
```

```
    <p style = "text-align: justify"> ram </p>
```

```
  </body>
```

```
</html>
```

css

Property

* एक letter से दूसरे letter का word, से दूसरे word के बीच में space के purpose से properties होते हैं।

② <html>

<body>

<p>my name is ram</p>

<p style="line-height: 50px"> my name is ram </p>

<p style="line-height: 5px"> my name is ram </p>

<p style="word-spacing: 50px"> my name is ram </p>

<p style="word-spacing: 58px"> my name is ram </p>

<p style="letter-spacing: 20px"> my name is ram </p>

</body>

</html>

* letter or word ke bich me space ke purpose से :-

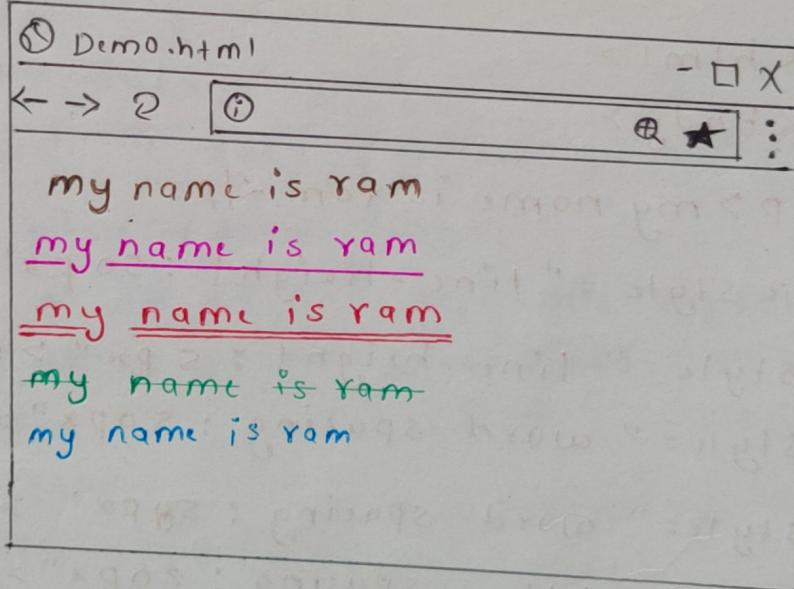
- * letter spacing - letter ke bich me space karta hai
- * word spacing - word ke bich me space karta hai
- * line spacing - Dono line ke bich me space karta hai

③

```

<html>
<body>
<p>my name is ram</p>
<p style="text-decoration: underline">my name is ram</p>
<p style="text-decoration: overline">my name is ram</p>
<p style="text-decoration: line-through">my name is ram</p>
<p style="text-decoration: none">my name is ram</p>
</body>
</html>

```



* इस normal paragraph में लिखेंगे तब भी normal message आएगा और <p style="text-decoration: none"> से CSS ki help से none pass करेंगे तब भी तो difference मिलता है?

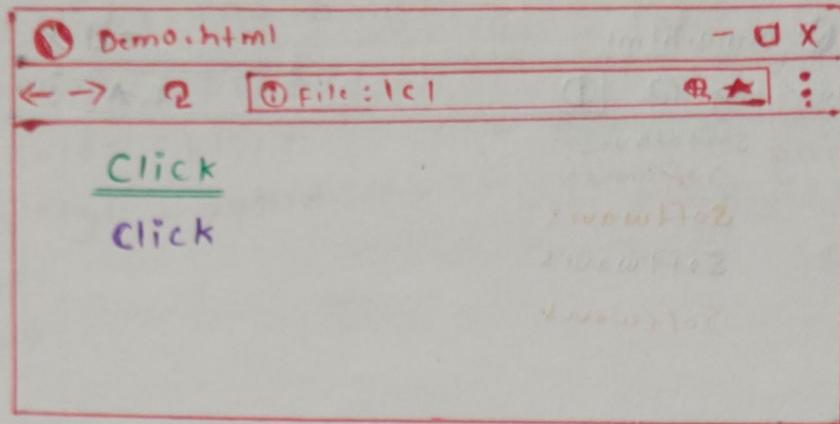
④

```

<html>
<body>
<p><a href="Demo.html">click</a></p>
<p><a href="Demo.html" style="text-decoration: none">click</a>
</body>
</html>

```

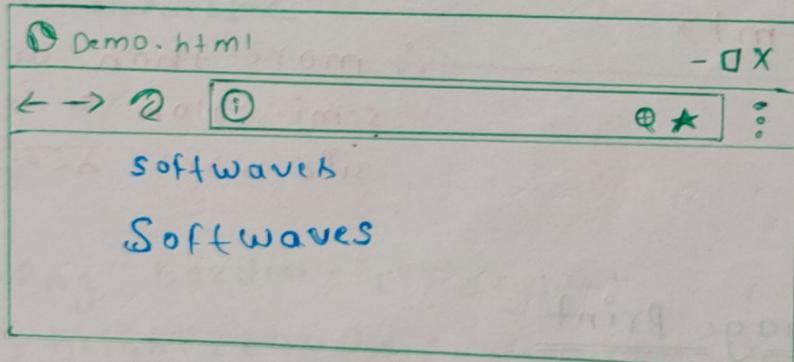
* line hatane
ki purpose se
name tag ko
use karte hai



.. anchor tag
ko by default
underline hatt
hai so set
hatane se
purpose se
text-decoration
none ka use hoga
hai.

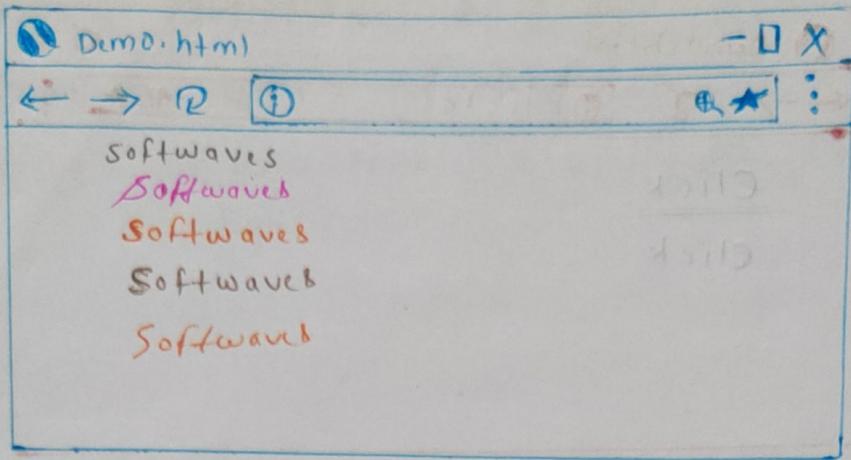
⑤

```
<html>  
<body>  
<p> softwaves </p>  
<p style="font-size: 30px"> softwaves </p>  
<p style="font-size: 200px"> softwaves </p>  
</body>  
</html>
```



⑥

```
<html>  
<body>  
<p> softwaves </p>  
<p style="font-style: italic"> softwaves </p>  
<p style="font-weight: bold"> softwaves </p>  
<p style="font-weight: goo"> softwaves </p>  
<p style="font-weight: Brush Script MT"> softwaves </p>  
</body>  
</html>
```



⑦ `<html>`
`<body>`
`<p>softwaves </p>`
`<p style="font-weight: bold; font-style: italic">`
my name is ram `</p>`
`</body>`
`</html>`

∴ more than one property ko(;) semi-colon se separate karne ke liye work hoga

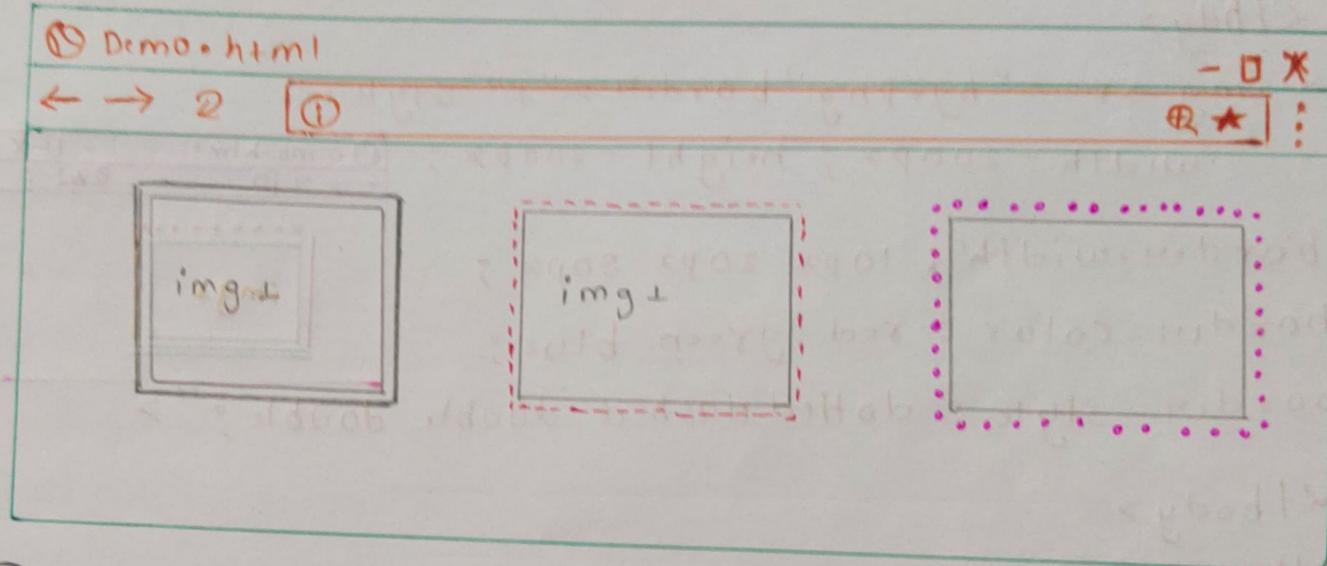
For Image Print

⑧ `<html>`
`<body>`
``
``

```

      ∵ img उसी same folder में
border-style: dotted;"> होना chahiye जिसमें program
save hai
</html>
</body>

```



⑨

```

<html>
<body>

```

```

  
</html>
```

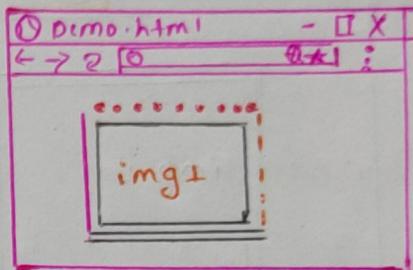
⑩ <html>

```
<body>
```

```


```
</body>
```

```
</html>
```



⑪

```
<html>
```

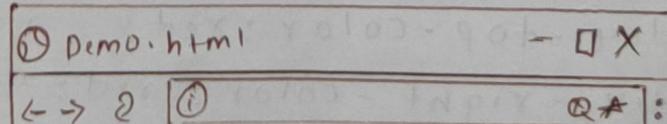
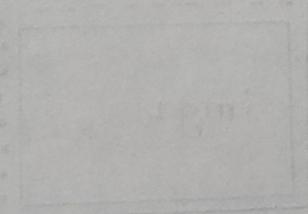
```
<body>
```

```

```

```
</body>
```

```
</html>
```



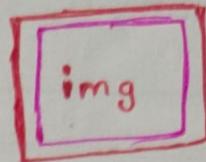
(12) <html>

<body>



</body>

</html>



22/08/23

Tuesday -

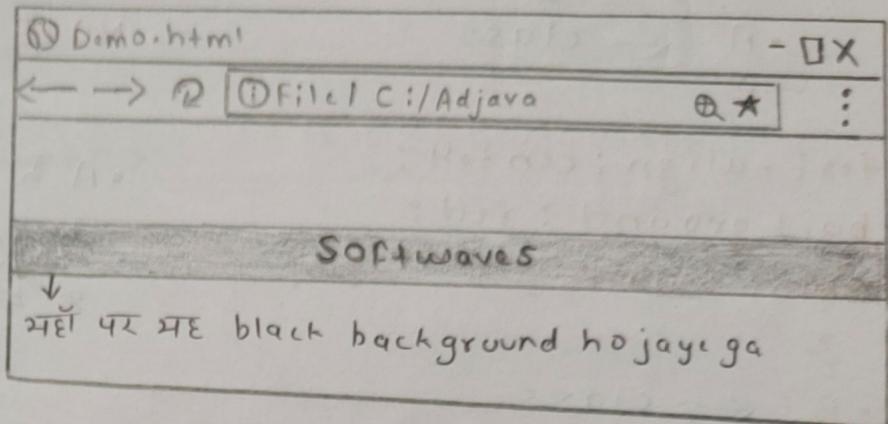
1. <html>

<body>

<p style="text-align: center; color: white"> softwaves </p>

</html>

</body>



### i) INTERNAL CSS Example :-

②

<html>

<head>

<style type="text/css">

p

{

text-align: center;

background: blue;

color: white;

}

</style>

</head>

<body>

<p> softwaves </p>

<p> softwaves </p>

<p> softwaves </p>

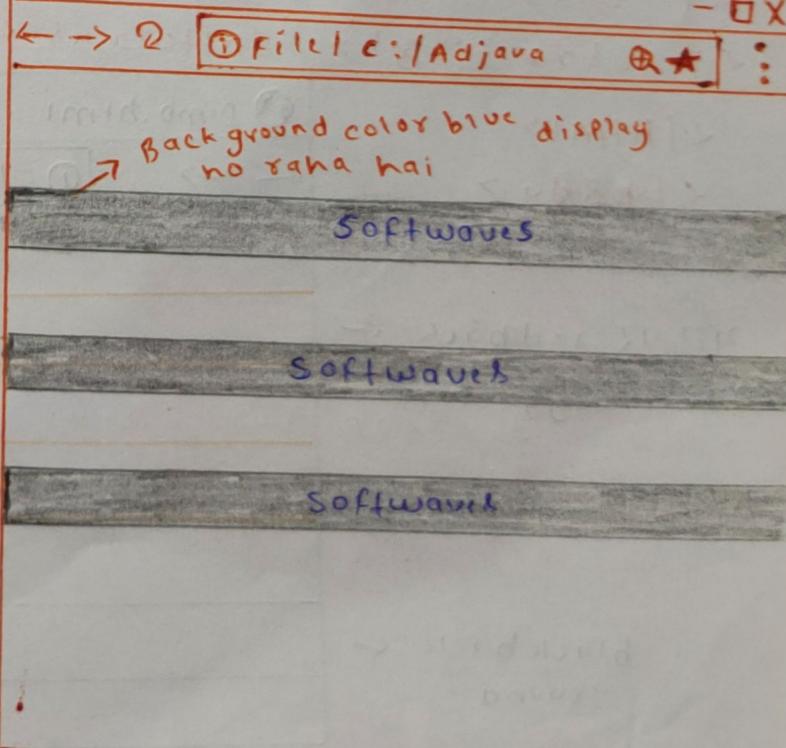
<p> softwaves </p>

</html>

</body>

← अपने पर में को स्ट्रीमिंग नहीं apply  
करते हैं तो सभी textfield पर apply  
नहीं हो जायेगा

④ Demo.html



③ यदि हम कुछ paragraph per different और उस paragraph में different karna hai to class लगायेंगे!

```
<html>
<head>
<style type="text/css">
```

• A ← class

NOTE - class लगाता hai भी (.A)  
से लगेगी

```
text-align: center;
background: red;
color: white;
```

3

• B ← class

```
text-align: center;
background: black;
color: white;
```

3

```
</style>
```

```
</head>
```

```
<body>
```

```
<p class="A"> softwaves </p>
```

```
<p class="A"> softwaves </p>
```

```
<p class="B"> softwaves </p>
```

```
<p class="B"> softwaves </p>
```

```
</html>
```

```
</body>
```

| ① Demo.html    | - X |
|----------------|-----|
| ② F1 C:/Adjara | ★ : |
| softwaves      |     |

अपने पर redback ←  
ground aagya hai

blackback ←  
ground

\* class or id मे kya different hai ?

class(.A) से create hoti hai or id(#A) से create hoti hai

# id :- jquery के concept मे अक्षर से समस्या आएगी

(4) <html>

<head>

<style type="text/css">

# A

{

text-align: center;

background: red;

color: white;

#

B

{

text-align: center;

background: green;

color: white; ← font color change

3

nota hai

</style>

</head>

<body>

<p id="A"> softwaves </p>

<p id="A"> softwaves </p>

<p id="B"> softwaves </p>

<p id="B"> softwaves </p>

</html>

</body>

\* Same as output

A class से  
arraha tha  
bahi

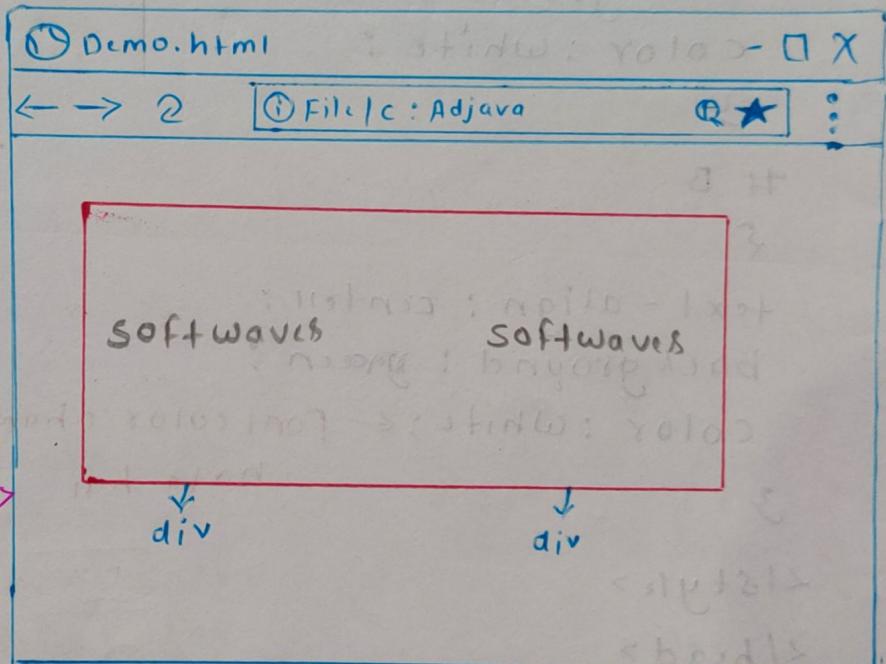
⑤

\* <div></div> (division tag)

```
<html>
<head>
<style type="text/css">
div
{
background: red;
width: 200px;
height: 200px;
color: white;
font-size: 30px;
padding: 20px;
float: left;
```

3

```
</style>
</head>
<body>
<div>
<p> softwaves </p>
</div>
<div>
<p> softwaves </p>
</div>
</body>
```



⑥ <html>

<head>

<style type="text/css">

body

{

background: linear-gradient(red, blue, green, yellow);

}

</style>

</head>

<body>

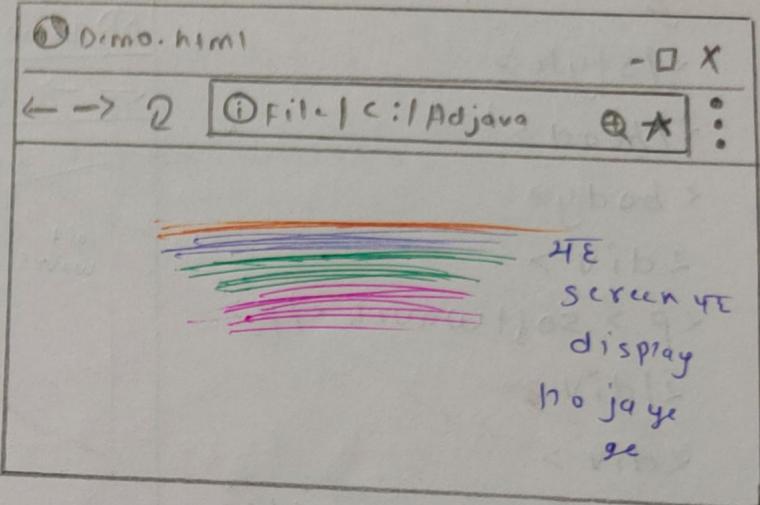
<div>

<p> softwaves </p>

</div>

</body>

</html>



⑦

<html>

<head>

<style type="text/css">

<body>

{

background: black;

}

div

{

\* div:hover = इसके अन्तर  
जो भी लिंक है वो div के  
अपर्याप्त cursor ले जाने पर  
perform hota hai

background: linear-gradient(red, blue, green);

width: 200px;

height: 200px;

color: green;

font-size: 20px;

padding: 20px;

float: left;

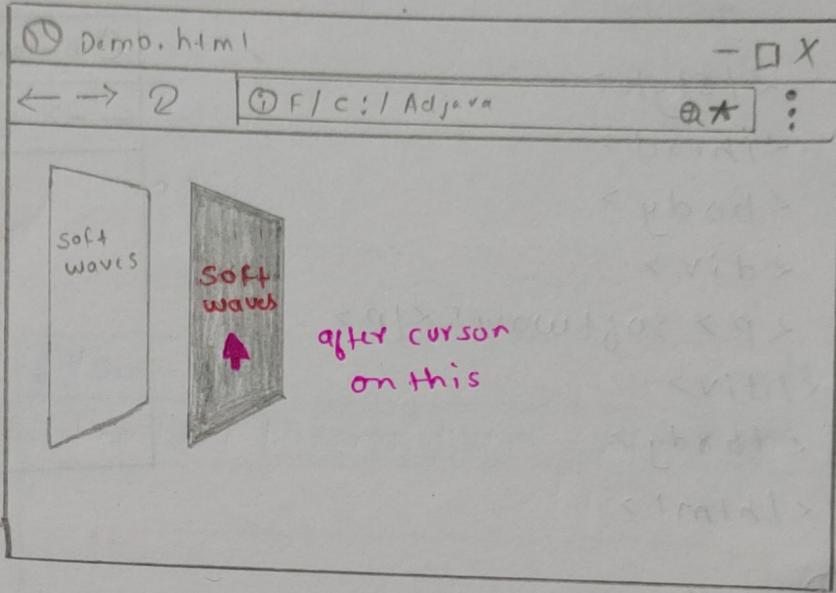
margin-right: 50px;

```

div : hover
{
background : black;
font-size : 10px;
box-shadow : 10px 10px white;
border-radius : 120px 0px 120px 20px;
}

</style>
</head>
<body>
<div>
<p> softwaves </p>
</div>
<div>
<p> softwaves </p>
</div>
</body>

```



⑧

```

<html>
<head>
<style type="text/css">
body
{
background-image : url (ab.jpg);
}

mymenu
{
background : rgba (0,0,0,0.5);
color : white;
text-align : right;

mymenu ul li

```

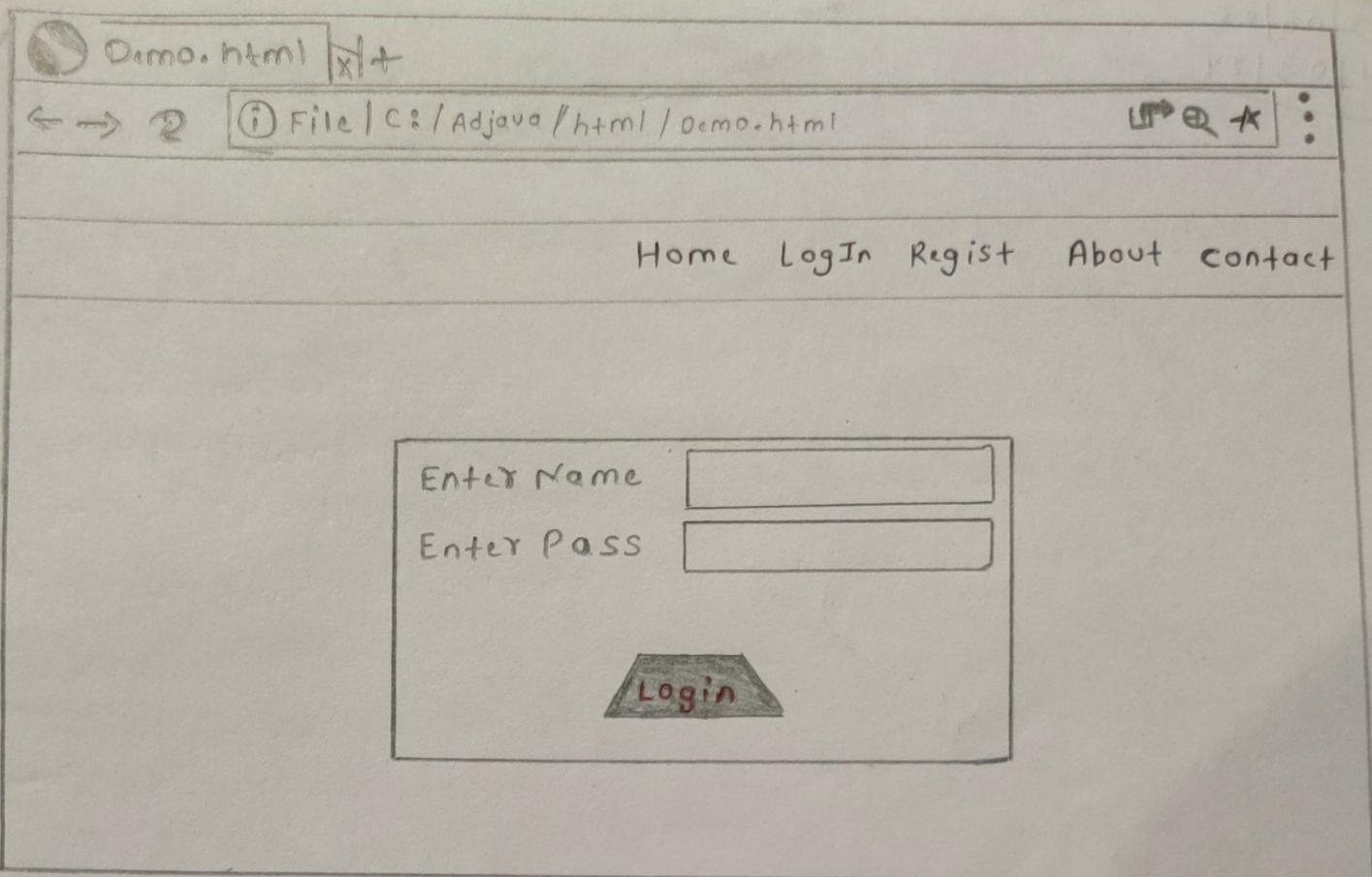
```
5
display: inline-block;
padding: 20px;
3
table
5
background: rgba(0,0,0,0.5);
color: white;
margin-top: 100px;
box-shadow: 10px 10px 10px white;
3
.oB
5
width: 85px;
height: 40px;
background: linear-gradient(red, black);
color: white;
border-radius: 40px 0px 40px 0px;
3
.oB : hover
5
width: 85px;
height: 40px;
background: linear-gradient(to right, black, red);
color: white;
border-radius: 0px 40px 0px 40px;
3
</style>
</head>
<body>
<div id="my menu">

 Home
```

```
 LogIn
 Registration
 About
 Contact

</div>

<div id="mydata">
<center>
<table cellpadding="12">
<tr>
<td> Enter Name </td>
<td> <input type="text" placeholder="Enter Name">
</td>
</tr>
<tr>
<td> Enter Pass </td>
<td> <input type="password" placeholder="Enter pass word">
</td>
</tr>
<tr>
<th colspan="2"><input type="submit" class="B" value="Log In"></th>
</tr>
</table>
</center>
</div>
</body>
</html>
```

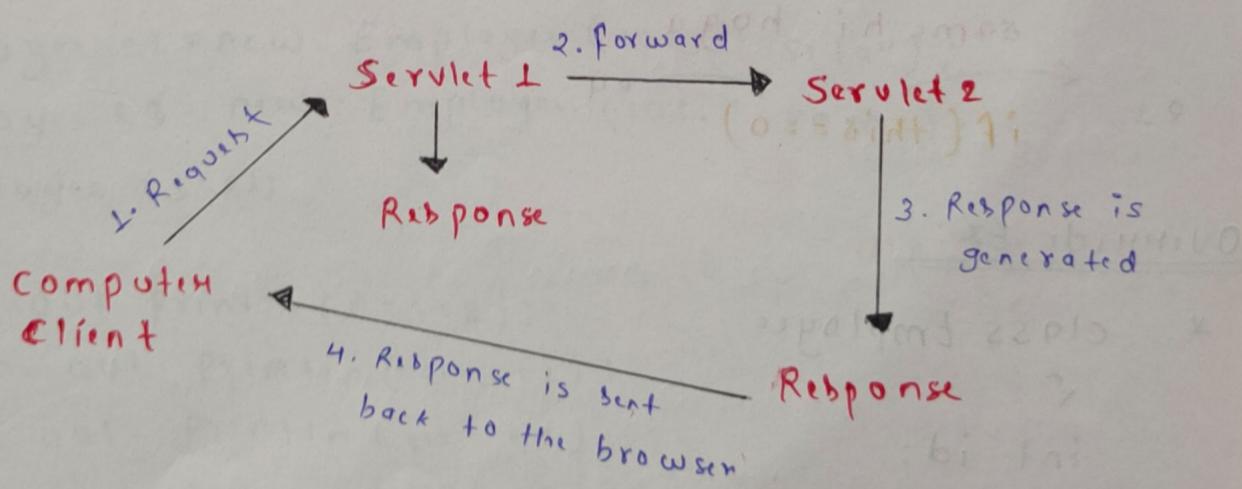


## \* Request Dispatcher in Servlet \*

The RequestDispatcher interface provide the facility of dispatching the request to another resource it may be html, servlet or jsp this interface can also be used to include the content of another resource also it is one of the way of servlet collaboration. there are two method defined in the RequestDispatcher interface they are :-

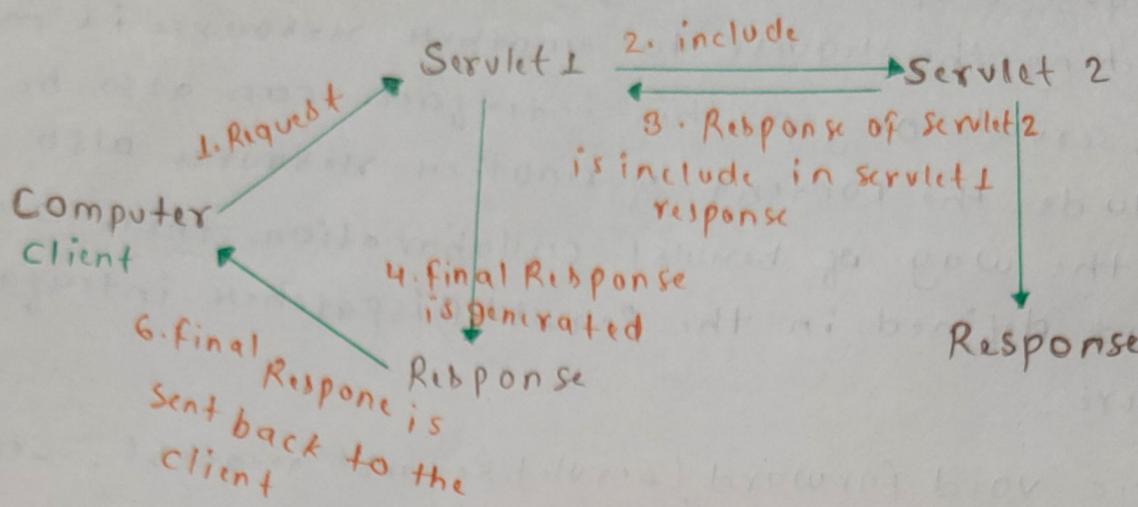
1. public void forward (ServletRequest request, ServletResponse response) throws ServletException, java.io.IOException : Forward a request from a Servlet to another resource ( servlet/JSP file or HTML file ) on the server.
2. public void include (ServletRequest request, ServletResponse response) throws ServletException, java.io.IOException : Include another content of a resource ( servlet, JSP page, or HTML file ) in the resource.

### forward() method:



As you see in the above figure, response of second servlet is sent to the client response of the first servlet is not display to the user.

## include() Method :-



\* As you can see in the above Figure, response of second servlet is included in the response of the first servlet that is being sent to the client

\* == method ko override kyu karke hai  
== method ko override content comparison ke purpose se kya jata hai

\* e1 == e4 ka difference same hai to content bhi same hi hogा

e1 ← if(this==0) → e4

## Override EX:

\* class Employee

{

    int id;

    String name;

Employee (int id, string name)

{

    this.id=id;

    this.name=name;

```
public boolean equals(Object o)
{
 if (this==o)
 return true;
 if (o instanceof Employee)
 {
 Employee e=(Employee)o;
 if (id==e.id & name.equals(e.name))
 return true;
 return false;
 }
 return false;
}
```

```
class Demo
{
 public static void main(String args[])
 {
 Employee e1=new Employee(101,"ram");
 Employee e2=new Employee(102,"sita");
 Employee e3=new Employee(101,"ram");
 Employee e4=e1;

 System.out.println(e1==e2);
 System.out.println(e1==e2);
 //System.out.println(e1==e3);
 System.out.println(e1==e3);

 System.out.println(e1.equals(e2));
 System.out.println(e1.equals(e3));
 System.out.println(e1.equals(e4));
 }
}
```

OIP- F

F

+

F

+

F

F

F

```
System.out.println(e1.equals(null));
System.out.println(e1.equals(101));
System.out.println(e1.equals("Yam"));
```

3

3

② class Employee

{

int id;

String name;

Employee(int id, String name)

{

this.id=id;

this.name=name;

3

public boolean equals(Object o)

{

return this==o;

3

3

class Demo

{

public static void main(String args)

{

Employee e1=new Employee(101,"Yam");

Employee e2=new Employee(102,"Sita");

Employee e3=new Employee(101,"Yam");

Employee e4=e2;

System.out.println(e1==e2);

System.out.println(e1==e3);

|                                       |            |
|---------------------------------------|------------|
| System.out.println(e1==e4);           | 01P. False |
| System.out.println(e1.equals(e2));    | False      |
| System.out.println(e1.equals(e3));    | true       |
| System.out.println(e1.equals(e4));    | false      |
| System.out.println(e1.equals("ram")); | false      |
| System.out.println(e1.equals("101")); | false      |
| System.out.println(e1.equals(null));  | false      |

3  
3

1. What is garbage object?

In java garbage mean 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.

\* unreferenced or unreachable object ko ham garbage object ~~बना~~ कै

# what is Garbage Collection?

Garbage means unreferenced objects.

Garbage collection is a process of reclaiming the runtime unused memory automatically. in other words, its a way to destroy the unused memory.

★ Advantage of Garbage Collection

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

Q. Way to make an object eligible for G.C in java?

there are five way to make an object eligible for GC in java :-

- i.) Null assign the Reference Variable.
- ii) Reassign the Reference variable.
- iii) By creating object inside a method (Local Object)
- iv) Anonymous Object.
- v) Island of isolation.

### i) Nullifying the Reference Variable :- Program :-

assign null to the Reference variable of all those Object which are no longer required. This makes the useless object automatically eligible for the purpose of garbage collection.

eg:- class A

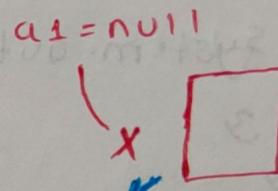
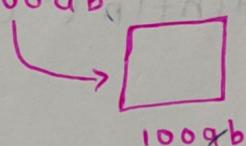
{

void show()

{

System.out.println("class A");

}



इसमें null pass किया हाई तो  
point nahi karega अतः  
Runtime Exception aayegi

O/p - class A

3

Exception :- NULL pointer

class DEMO

Exception

{

public static void main(String args[])

{

A a=new A();

a.show();

do or

a=null;

a.show();

3

A a=new A();

ये object hai

इसकी memory kuch  
Reference rahaega to a  
me jaka store ho  
jaye ga.

② Reassign the reference variable :- जब फिरी class में दो reference variable पड़ते हों different memory को point कर रहे हैं। But वहाँ से same को point करने लगे like  $a_1 = a_2$  : इस case में  $a_2$  भी  $a_1$  की ही memory को point करेगा or  $a_2$  की memory free हो जायेगी और उस memory को point करने के liye koi nahi hai isliye उसे GC ने पास पढ़वा देगा।

class A

{

void show()

{

System.out.println("class A");

obj-class A

class A

class A

class A

3  
3

class Demo

{

public static void main (String args)

{

A a1=new A();

A a2=new A();

a1.show();

a2.show();

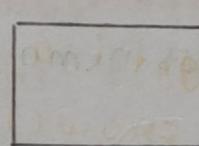
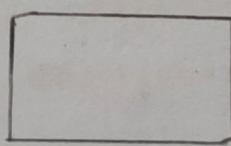
a2=a1;

a1.show();

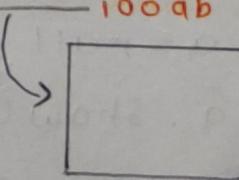
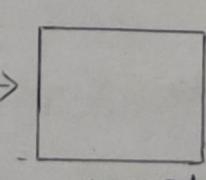
a2.show();

$a_1 = 100\ ab$

$a_2 = 200\ cd$



$a_2 = a_1$



इस memory को point होते

हाला कोई तरी है इसलिए इसे GC

के पास पढ़वा दिया time आते पर नहीं

इसे GC free कर देगा

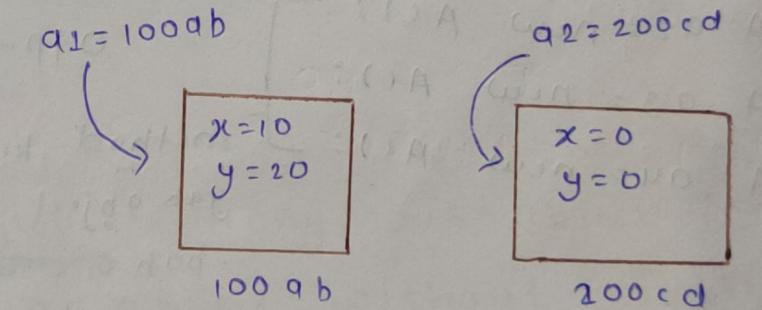
ii) class A  
 {  
 int x, y;  
 void get (int a, int b)  
 {  
 x = a;  
 y = b;  
 }  
 }

3  
 void show ()  
 {  
 System.out.println ("x=" + x);  
 System.out.println ("y=" + y);  
 }  
 }

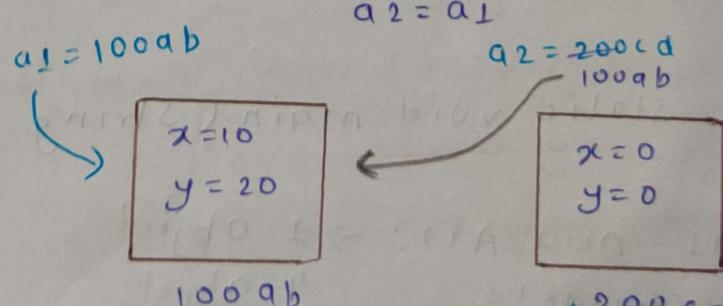
class Demo  
{

public static void main (String args)  
{

A a1 = new A();      a1 = 100ab  
 A a2 = new A();  
 a1.get (10, 20);  
 a1.show ();  
 a2.show ();



a2 = a1;  
 a1.show ();  
 a2.show ();



3

memory free  
 ho gai hai

\* समा एक ही memory को more than one reference variable point kar sakte hai ya nahi  
→ YES

\* समा एक reference variable more than one object ko point kar sakte hai ya nahi  
→ NO

### (iii) By Creating Object inside a method (Local Object) :-

Local Object का मतलब इसी method की body के अंदर बना hua object हो जो method के बहार दूरी से GC के पास पहुँच जाता है!

class A

{

void show()

{

System.out.println("class A")

A a2 = new A();

A a3 = new A();

A a4 = new A();

}

}

class Demo

{

public static void main(String args)

{

A a1 = new A(); → 1 object

a1.show(); → 4 object

System.out.println("Yam");

}

}

Yam  
method ki execute hone ke bad  
yah object destroy ho jayega or  
bah a.c ke पास काले jaye ge.

Q1P- class A  
sam

```
class A{
 void show()
 {
 System.out.println("class A");
 }
}
A a2=new A();
A a3=new A();
A a4=new A();
return a4;
}
```

3  
class Demo  
{  
 public static void main(String args[])  
 {

A a1=new A(); 2 एक object hai

A a5=a1.show(); 2 एक object hai 2 destroy ho jayega

System.out.println("sam"); 2 वर्द्धमान

3  
3

iv) Anonymous Object :- Anonymous Object in java

means creating an object without any reference variable. generally when creating object in java you need to assign a name to the object. But the anonymous object in Java allow you to create an object without any name assigned to that object.

O/P- class A constructor

```
class A
{
 A()
}
```

System.out.println("class A constructor");

3

3

class Demo

2

public static void main(String args)

2

new A(); → इस line के execute होते ही अपे object  
अपे के पास eligible ho jaye ga इसलिए

System.out.println("Yam"); इसे हम आगे access nahi

kar sakte hai

3

\* class A

2

A()

2

System.out.println("class A constructor");

3

void show()

2

System.out.println("class A show method");

3

class Demo

2

public static void main(String args)

2

new A().show(); → ऐसा हिन्दूल कर सकते हैं show  
method class A में ही नहीं

System.out.println("Yam"); उसे access karuge लो हो

jayega.

3

\* बिना नाम बाले object को हि  
anonymous बोलते हैं

```
class A
{
 int x,y;
 A()
 {
 System.out.println("class A constructor");
 }
 void get(int a, int b)
 {
 x=a;
 y=b;
 }
 void show()
 {
 System.out.println("x=" + x);
 System.out.println("y=" + y);
 }
}
class demo
{
 public static void main(String args)
 {
 new A().get(10, 20);
 new A().show();
 }
}
```

Output - class A constructor  
class A constructor  
 $x=0$   
 $y=0$

26/08/23

Saturday -

① class A  
{  
    int x, y;  
    A()  
    {  
        System.out.println("class A constructor");  
    }  
    void get(int a, int b)  
    {  
        x = a;                                                    OP-ERROR: void cannot be  
        y = b;                                                    dereferenced new A().get  
    }  
    void show()  
    {  
        System.out.println("x=" + x);  
        System.out.println("y=" + y);  
    }  
}

class Demo  
{  
    public static void main(String args)  
    {  
        new A().get(10, 20).show();  
    }  
}

② class A  
{  
    int x, y;  
    A()  
    {  
        System.out.println("class A constructor");  
    }  
    void get(int a, int b)  
    {  
    }

```
x=a;
y=b;
}
void show()
{
 System.out.println("x="+x);
 System.out.println("y="+y);
}
}
```

O/p - incompatible types: void  
cannot be converted to A

```
class Demo
```

```
{
public static void main(String args)
{
 A a=new A().get(10,20);
 a.show();
}}
```

③ class A

```
{
 int x,y;
 A()
{
```

```
 System.out.println("class A constructor");
}
```

```
int get(int a, int b)
```

```
{
 x=a;
 y=b;
 return 100;
```

O/p - class A constructor

100

```
void show()
{
 System.out.println("x="+x);
}
```

System.out.println("y=" + y);

3

class Demo

{

public static void main (String args)

{

int x=new A().get(10,20);

System.out.println(x);

3

3

④

class A

{

int x,y;

A()

{

System.out.println("class A constructor");

3

A get (int a, int b)

{

x=a;

y=b;

return new A();

3

void show()

{

System.out.println("x=" + x);

System.out.println("y=" + y);

3

class Demo

{

public static void main (String args)

{

int x=new A().get(10,20);  
System.out.println(x);  
3  
3  
OIP- error: incompatible types: A cannot  
be converted to int

⑤ class A  
{  
 int x,y;  
 A()  
 {  
 System.out.println("class A constructor");  
 }  
 A get(int a, int b)  
 {  
 x=a;  
 y=b;  
 return new A(); // after this line object created  
 }  
 void show()  
 {  
 System.out.println("x=" + x);  
 System.out.println("y=" + y);  
 }  
}  
class Demo  
{  
 public static void main (String args[])  
 {  
 new A().get(10,20).show();  
 }  
}

OIP- class A constructor  
class A constructor  
x=0  
y=0

⑥ class A  
{  
 int x,y;

A()

{

System.out.println("class A constructor");

}

A get(int a, int b)

{

x=a;

y=b;

return this; मैं इसे return करता हूँ कि 10, 20

output aayega with this दिक्षिणीय वाक्य को  
current object ke reference ko hold

void show()

{

System.out.println("x=" + x);

System.out.println("y=" + y); O/P - class A constructor

}

3

class Demo

{

public static void main(String args[])

{

new A().get(10, 20).show();

3

3

⑦ class A

{

int x, y;

A()

{

System.out.println("class A constructor");

3

int get(int a, int b)

O/P - error: int cannot be

{

x=a;

y=b;

return 100;

defined hence new A().get  
(10, 20).show();

3

```
Void show()
{
 System.out.println("x=" + x);
 System.out.println("y=" + y);
}

class Demo
{
 public static void main(String args[])
 {
 new A().get(10, 20).show();
 }
}

⑧ class A
{
 int x, y;

 A()
 {
 System.out.println("class A const...");

 x = a;
 y = b;

 void show()
 {
 System.out.println("x=" + x);
 System.out.println("y=" + y);
 }

 void show2()
 {
 System.out.println("class A");
 }
 }
}
```

Compiler error: void cannot be  
dereferenced new A().  
get(10, 20).show();  
show2();

```
class Demo
{
 public static void main(String args)
 {
 new A().get(10, 20).show1().show2();
 }
}
```

⑨ class A

```
int x, y;
A get(int a, int b)
{
 x = a;
 y = b;
 return this;
}

void show()
{
 System.out.println("x=" + x);
 System.out.println("y=" + y);
}

void show2()
{
 System.out.println("class A");
}
```

class Demo

```
public static void main(String args)
{
 A a = new A().get(10, 20).show();
 a.show2();
}
```

⑩ class A

{

int x,y;

A get(int a, int b)

{

x=a;

y=b;

return this;

}

void show()

{

System.out.println("x=" + x);

System.out.println("y=" + y);

}

void show2()

{

System.out.println("class A");

}

class Demo

{

public static void main(String args)

{

A a=new A(); a.get(10,20);

a.show();

a.show2();

}

⑪ class A

{

int x,y;

A get(int a, int b)

{

x=a;

y=b;

O/p - x=10  
y=20  
class A

```
return this;
}
A show()
{
System.out.println("x=" + x);
System.out.println("y=" + y);
}
void show2()
{
System.out.println("class A");
}
class Demo
{
public static void main(String args)
{
A a = new A(); a.get(10, 20);
a.show();
a.show2();
}
}
```

(12) class A

```
int x, y;
A get(int a, int b)
{
x = a;
y = b;
return this;
}
A show()
{
System.out.println("x=" + x);
System.out.println("y=" + y);
}
return this;
}
```

O/p- x=10  
y=20  
class A

System.out.println("class A");

3

class Demo

2

public static void main(String args)

2

new A().get(10,20).show().show();

3

3

\* अगर previous example में देखा लिखेगे तो error आयेगा

A a=new A().get(10,20).show().show2();

ज्योंकि show2 का return type void है अर्थात् वह एक  
variable store nahi hoga

~~void show2()~~

2

System.out.println("class A");

return this;

3

में चल जाएगा ज्योंकि

return type A है!

class Demo

2

public static void main(String args)

2

A a=new A().get(10,20).show().show2();

3

Q. java Method Chaining ko support karta hai ya  
nahi

→ YES karta hai

new A().show1().show2();

✓

METHOD CHANNING :- ऐसे एक method se दूसरी method  
ko call karna एवं method channing कहा जाता है।

Q. ऐसी किसी भी class jo method channing पर काम  
karta hai?

String class वह ऐसी class है जो method channing  
पर काम करती है। Eg. s1.concat().length() →  
यह एक method से दूसरी method ko call करता है।

Class Demo

```
public static void main(String args)
{
 String s1 = "ram";
 String s2 = "ji";
 System.out.println(s1.concat(s2).length());
}
```

O/p - 5

ISLAND OF ISOLATION :- Object 1 references Object 2  
and Object 2 references Object 1. Neither Object 1 nor  
Object 2 is referenced by any other. That's an island of  
isolation.

Basically an island of isolation is a group of  
object that reference each other but are not referenced  
by any active object in the application. Strictly speaking  
even a single unreferenced object is an island of isolation  
too.

• Object reference to hogा यह को active reference  
nahi hogा us object की स्थापित की गई को 4TB  
पर्टी बायेगा उसी process को island of isolation  
नहीं है।

① class A

{

```
A i;
3
class Demo {
 public static void main(String args[]) {
 A a1 = new A();
 A a2 = new A();
 A a3 = new A();

 a1.i = a2; // Reference variable a1 points to object a2
 a2.i = a3; // Reference variable a2 points to object a3
 a3.i = a1; // Reference variable a3 points to object a1

 a1 = null; // Reference variable a1 is set to null
 a2 = null; // Reference variable a2 is set to null
 a3 = null; // Reference variable a3 is set to null
```

3 do memory & bridge & reflection to java  
3 On Java, bridge method is used to implement multiple inheritance. It is used to implement multiple inheritance by providing a common interface for different classes. It is also used to implement multiple inheritance by providing a common interface for different classes.

3 multiple inheritance is a feature of Java that allows a class to inherit from multiple parent classes. This is achieved through the use of interfaces, which define a set of methods that a class must implement. Interfaces are used to implement multiple inheritance by providing a common interface for different classes.

class A

{

    A i; // static variable to initialize i

3. illustrates bad idea to initialize

class Demo

{

    public static void main(String args)

        A a1 = new A(); // good idea

        A a2 = new A(); // bad idea

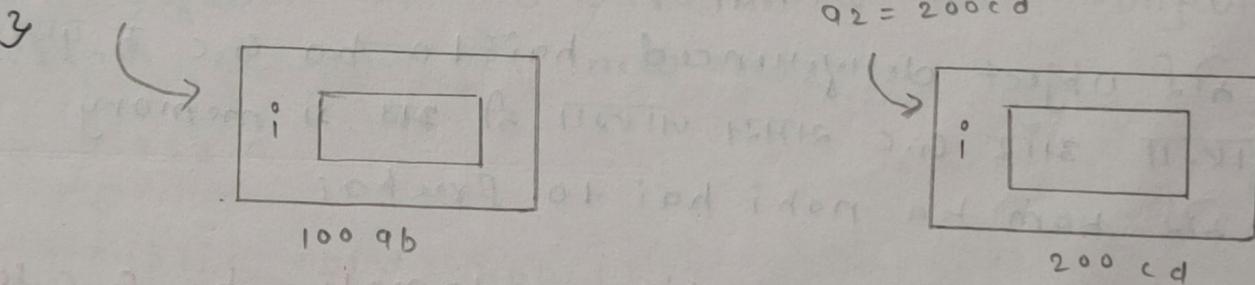
        a1.i = a2.i; // static variable is shared

        a2.i = a1.i; // static variable is shared

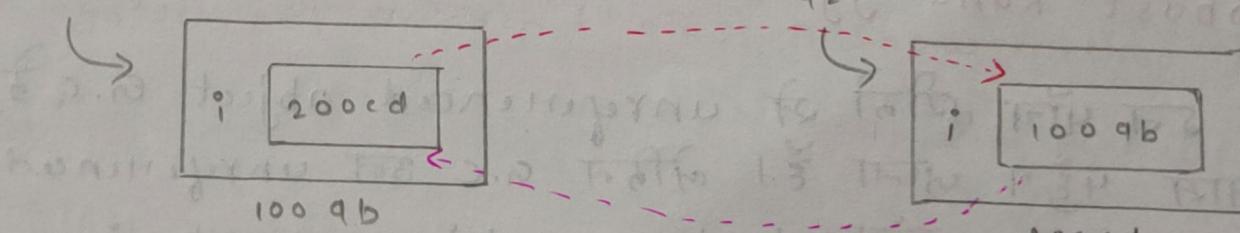
        a1=null;

        a2=null;

3. a1=100ab



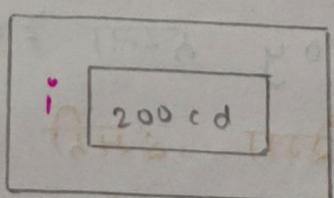
$a_2 = 200cd$



$a_1=null$

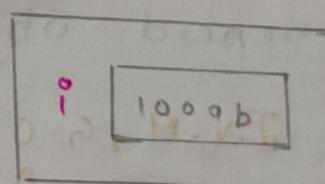
X

bad idea



X

bad idea



28/08/23

Monday

NOTE - Reference से connection cut hone पर inter relation cut nahi hota hai। Internally इन्हें पर्ह मुफ्त होते हैं कार्य करते हैं पर्ह वहाँ से हम access नहीं कर सकते हैं तो G.C के पास चला जाएगा।

Garbage Object :: Unreferenced or unreachable object java को robust (strong, self-dependent) bhi kaha jata hai.

\* C, C++ में memory और allocate गर्भ user को देकरna पड़ता था और उसे free (delete) भी user को देकरna padta tha.

java में memory allocate user करता है but delete (destroy) Java करता है।

जैसे ही कोई object unreferenced है तो bo G.C के पास चला जाएगा और G.C अपने पास ही अब ऐ memory user के कोई काम का nahi है तो free है।

Dynamic memory deallocation automatic by G.C so इसे robust kaha gyahai

NOTE :: इन पाँच तरीकों से unreferenced object G.C के पास पहुँच जाता है। लेकिन G.C उन unreferenced Object को तुरंत destroy nahi karta hai

jab J.V.M, G.C को call karta hai तब G.C उन unreferenced objects को destroy करता है।

लेकिन J.V.M, G.C को नव call करेगा इसकी koi guarantee नहीं है। लेकिन हम JVM को request कर सकते हैं कि आप G.C को call करें।

J.V.M द्वारा request को accept होगा नहीं होगी  
आप नहीं होगी इसकी कोई guarantee नहीं है।

लेटित go-to cases में JVM programmer ने  
request को accept करता है।

\* JVM request karne ke kitne tariche hain?

There are two ways to do it

i) System.gc();

ii) Runtime.getRuntime().gc();

इन दो तरीकों से हम JVM से request कर सकते हैं  
जिसके हम request की तरह सकते हैं. foo.main();

\* JVM के part को ही G.C कहा जाता है।

Q. हम किस type का object create कर सकते हैं

एम default or no parameter वाला object ही  
create कर सकते हैं new A();

class A

{

public void show()

{

System.out.println("class A show method");

}

class Demo

{

public static void main (String args)

{

A a=new A(); ← default object hai यहाँ no para

a.show();

metor है।

} }

By default, default constructor call hota hai instance object ko initialize karne ke liye ye usi chaiti hै जो भी call na hoतो उसे private bana dege.

class A

{

private A()

{ }

public void show()

{

System.out.println("class A");

}

class Demo

{

public static void main(String args)

{

A a=new A();

a.show();

}

}

Constructor ko private bana sakte hai ya nahi  
nahi

\* मर्दि इस चाहते हैं constructor ko private रख के method ke andar ~~जो~~ data ko access करते हैं तो

class A

{

private A()

{ }

```

static A getObject()
{
 return new A();
}

void show()
{
 System.out.println("class A show method");
}

class Demo
{
 public static void main(String args)
 {
 A a1 = A.getObject();
 a1.show();
 }
}

```

NOTE :- Private data class के बाहर access नहीं हो सकता  
मिस्त्र बाहर हो सकता है।

Q. What is Singleton class?

SINGLETON CLASS :- "Singleton is a design pattern that ensures that a class can only have one object. To create a singleton class a class must implement the following properties:-

- Create a private constructor of the class to restrict object creation outside of the class.
- Create a private attribute of the class type that refers to a single object.

• Create a public static method that allows us create and access the object we created. Inside the method we will create a condition that restricts us from creating more than one object.

फिलहाल यह बार object create करें लेकिन output same ही आना चाहिए तो वो singleton class है।

- i) Single या limited object को create करने के purpose से singleton class का use किया जाता है।
- ii) Singleton object को क्यानें के लिए हमें constructor को private ढंग से बनाए करता पड़ता है।
- iii) Singleton class में इस static method होती है जो उसी class के object को return करती है।
- iv) इस static method को हम यहाँ बार call करे same दृष्टि output aayega.
- v). इस static method को hi factory method कहा jata है।

"After the first time, if we try to instantiate the singleton class, the new variable also points the first instance created".

class A

₹

private A()

₹ 3

static A getObject()

₹

return new A();

hashcode different aayega

मुझे dono key lie different object नहीं हैं।

```
void show()
{
 System.out.println("class A show method");
}

class Demo
{
 public static void main(String args)
 {
 A a1=A.getObject();
 A a2=A.getObject();
 a1.show();
 a2.show();

 System.out.println(a1.hashCode());
 System.out.println(a2.hashCode());
 }
}

* class A
{
 A a;
 private A()
 {
 }

 static A getObject()
 {
 if(a==null)
 {
 a=new A();
 }
 return a;
 }

 void show()
 {
 }
}
```

Output - class A show method  
class A show method  
2018699584  
1311053135

```
System.out.println("class A show method");
```

3

```
class Demo
```

{

```
public static void main(String args)
```

{

```
A a1=A.getObject();
```

OIP- error: non-static  
variable a cannot be  
referenced from a static  
context.

```
A a2=A.getObject();
```

```
a1.show();
```

```
a2.show();
```

```
System.out.println(a1.hashCode());
```

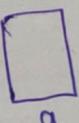
```
System.out.println(a2.hashCode());
```

3

```
class A
```

{

```
static A a;
```



```
private A()
```

{}

```
static A getObject()
```

{

```
if(a==null)
```

{

```
a=new A();
```

}

```
return a;
```

}

```
void show()
```

{

```
System.out.println("class A show method");
```

3

```
class Demo
{
 public static void main (String args)
 {
 A a1 = A. getObject();
 A a2 = A. getObject();
 a1. show();
 a2. show();
 }
}
```

System.out.println(a1.hashCode());  
System.out.println(a2.hashCode());

3

### \* NETBEANS \*\*\*

# यदि हमें netBeans में project बनाना है तो :-

#### ★ (.java file)

Source Package → Default Package → right click →

Servlet → click → class Name → next →

Add (web.xml) → finish → web.xml page और

जाकरी आरे (.java) भी

#### ★ (html page)

Web Page → right click → new → HTML → name  
→ finish

Netbeans ka use क्यों होते हैं?

Netbeans ka use इसलिए होते हैं क्योंकि complexity कम  
हो जाये और user ka time भी बचे

" NetBeans IDE allows our developers to  
build high quality application quickly and easily we

use the IDE to create web application using the java EE platform as well as PHP Javascript and Ajax.

## PROJECT IN NETBEANS

abc.css

```
body
{
background-image: url(ab.jpg);
}

#ram
{
background: rgba(0,0,0,0.2);
color: red;
text-align: right;
}

#ram ul li
{
display: inline-block;
padding: 30px;
color: red;
box-shadow: 5px 5px 5px white;
}

#ram ul li: hover
{
width: 90px;
color: red;
height: 20px;
background: cyan;
border-radius: 20px 20px 20px;
font-family: cooper Black;
}
```

table

{

background:rgba(0,0,0,0.2);

color:white;

font-size:30px;

margin-top:100px;

box-shadow:10px 10px 10px black;

font-family:Rockwell Condensed;

}

.B

{

width:100px;

height:40px;

background:linear-gradient(black,red);

border-radius:0px 30px 0px 30px;

color:white;

}

.B:hover

{

width:85px;

height:40px;

background:linear-gradient(red,black);

border-radius:30px 0px 30px 0px;

color:white;

}

.A:hover

{

width:85px;

height:40px;

background:linear-gradient(red-black);

border-radius:30px 0px 30px 0px;

# Connector को कहो रखना है तो data mysql database में चला जाए

Libraries → Right click → Add → JAR folder  
click → paste connector (mysql).

\* Exception :- java.sql.SQLNonTransientConnectionException (keystore was tampered with or password was incorrect) से exception हवा और क्यों आती है?  
⇒ अगर हमें इसी program में connection के code में java.sql.SQLNonTransientConnectionException आती है तो को useSSL=false लिखने से solve हो jaye!

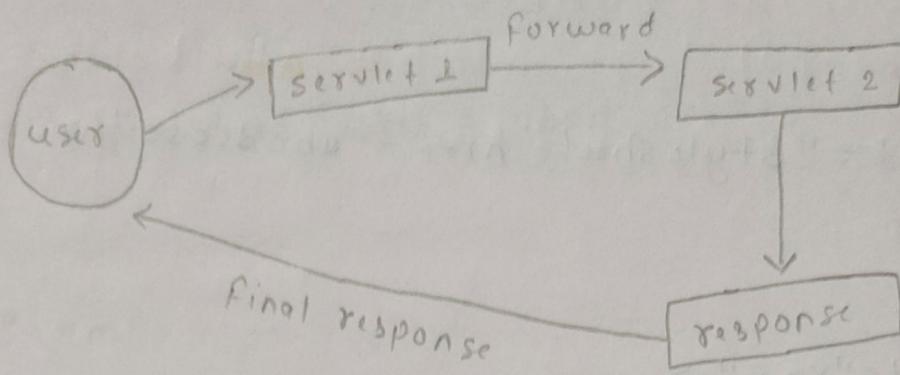
useSSL=false :- for compliance with existing application not using SSL the verifyServer certificate property is set to 'false'. you need either to explicitly disable SSL by setting useSSL=false; or set useSSL=true and provide truststore for server certificate verification.

# Request Dispatcher :- जबकि login page पर गलत username or password enter करते पर error message के साथ page के show होती है लेकिन इस बादते ही उसी page पर show होते ही Request Dispatcher का use करते हैं।

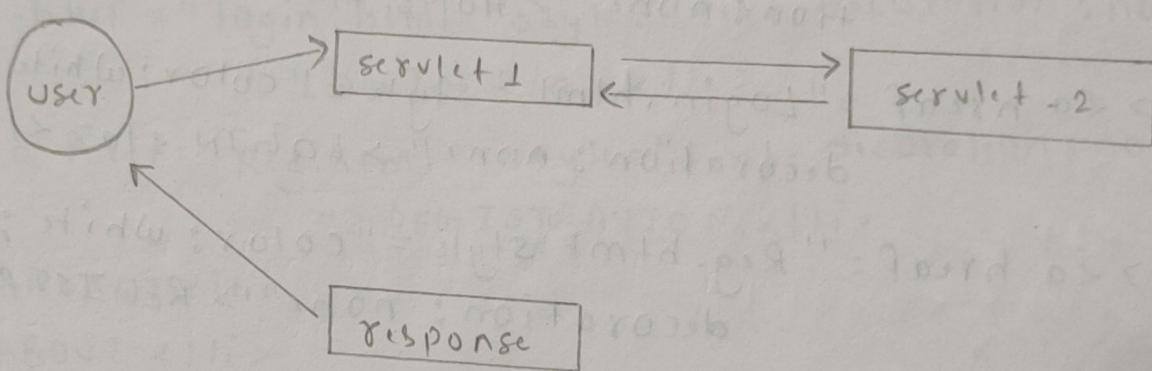
- 1.) The RequestDispatcher interface provides the facility of dispatching the request to another resource it may be html, servlet or jsp.
- 2). there are two methods defined in the RequestDispatcher interface.

forward() and include()

## i) Forward



## ii) include()



i) `public void forward (ServletRequest request, ServletResponse response) throws ServletException, IOException` :- Forwards a request from a servlet to another resource (servlet, JSP file, or HTML file) on the server.

ii) `public void include (ServletRequest, ServletResponse response) throws ServletException, IOException` :- Include the content of a resource (servlet, JSP page, or HTML file)

\* How to get the object of Request Dispatcher

The `getRequestDispatcher ()` method of `ServletRequest` interface return the object of `Request Dispatcher`.

`public RequestDispatcher getRequestDispatcher (String Resource);`

# Home.html

```
<html>
<head>
<link rel="stylesheet" href="abcd.css"/>
</head>

<body>
<div id="ram">

HOME
LOGIN
REGISTRATION
CONTACT
ABOUT

</div>
<div id="ram">
<center>
<table>
<tr>
<th> WELCOME MY HOME Page </th>
</tr>
</table>
</center>
</div>
</body>
</html>
```

## Reg.html

```
<html>
<head>
<link rel="stylesheet" href="abcd.css"/>
</head>
<body>
<div id="ram">

HOME
LOG IN
REGISTRATION
CONTACT
ABOUT

</div>
<center>
<form action="Reg">
<table cellspacing="12">
<tr>
<td>ENTER NAME </td>
<td><input type="text" placeholder="Enter Name" name="un">
</td>
</tr>
<tr>
<td>ENTER PASSWORD </td>
<td><input type="password" placeholder="Enter password" name="up">
</td>
</tr>
```

## Reg.html

```
<html>
<head>
<link rel="stylesheet" href="abcd.css"/>
</head>
<body>
<div id="ram">

HOME
LOGIN
REGISTRATION
CONTACT
ABOUT

</div>
<center>
<form action="Reg">
<table cellspacing="12">
<tr>
<td>ENTER NAME </td>
<td><input type="text" placeholder="Enter Name" name="un">
</td>
</tr>
<tr>
<td>ENTER PASSWORD </td>
<td><input type="password" placeholder="Enter password" name="up">
</td>
</tr>
```

<tr>  
<td> CONTACT NUMBER </td>  
<td> <input type="number" placeholder="Contact Number" name="uc"></td>  
</tr>  
  
<tr>  
<td> EMAIL ID. </td>  
<td> <input type="email" placeholder="Email id" name="uc"></td>  
</tr>  
  
<tr>  
<td> SELECT DOB </td>  
<td> <input type="DATE" ></td>  
</tr>  
  
<tr>  
<td> SELECT GENDER </td>  
<td>  
MALE <input type="radio" name="x">  
FEMALE <input type="radio" name="x">  
</td>  
  
</tr>  
  
<tr>  
<td> SELECT COURSE </td>  
<td>  
<select>  
<option> SOFTWARE DEVELOPER </option>  
<option> WEB DEVELOPER </option>  
<option> ETHICAL HACKING </option>  
</select>  
</td>

```
<tr>
<td>
<td> SELECT LANGUAGE </td>
<td>
<td>
<select>
<option> C/C++ </option>
<option> JAVA </option>
<option> DATA STRUCTURE </option>
<option> PYTHON </option>
<option> HTML </option>
<option> CSS </option>
<option> JS </option>
</select>
</td>
<tr>
<td>
<th colspan="2" <input type="submit" value="REGISTRATION" class="B" ></th>
</td>
</tr>
<td>
<table>
<form>
<center>
<body>
<html>
```

### LOGIN.html

```
<html>
<head>
<link rel='stylesheet' href='abcd.css' />
</head>
<body>
```

```
<div id="Yam">

HOME
LOGIN
REGISTRATION
CONTACT
ABOUT

</div>
```

```
<center>
<form action="Login">
<table cellpadding="15">
<tr>
<td> Enter Name </td>
<td> <input type="text" placeholder="Enter Name" name="u1" > </td>
</tr>
<tr>
<td> Enter Password </td>
<td> <input type="text" placeholder="Enter Name" name="u2" > </td>
</tr>
<tr>
<th colspan="2" > <input type="submit" class="B" value="LOGIN" > </th>
</tr>
</table>
</form>
</center>
</body> </html>
```

## Menu.html

```
<html>
<head>
<link rel="stylesheet" href="abcd.css"/>
</head>
<body>
<div id="ram">

 Menu <a>
 INSERT <a>
 SEARCH <a>
 UPDATE <a>
 showAll <a>
 Delete <a>
 About <a>
 CONTACT <a>
 LOGOUT <a>

</div>
<body>
</html>
```

## Insert.html

```
<html>
<head>
<link rel="stylesheet" href="abcd.css"/>
```

```
</head>

<div id="ram">

 MENU

 INSERT
 SEARCH
 SHOWALL
 Update
 Delete
 About
 Contact
 LOGOUT

</div>

<div id="ram1">
<form action="Insert">
<center>
<table cellpadding="12">

<tr>
<td> Enter Name </td>
<td><input type="text" placeholder="Enter Name" name="un" > </td>
</tr>

<tr>
<td> Enter ROLL NO </td>
<td><input type="text" placeholder="Enter Rollno" name="ur" > </td>
</tr>
```

```
<tr>
<td> Enter Physics </td>
<td> <input type="text" placeholder="Enter Phy" name="up"></td>
</tr>

<tr>
<td> Enter chemistry </td>
<td> <input type="text" placeholder="Enter chem" name="uc"></td>
</tr>

<tr>
<td> Enter Math </td>
<td> <input type="text" placeholder="Enter Math" name="m"></td>
</tr>

<th colspan="2" ><input type="submit" class="B" value="Insert" type="Reset" ></th>
</tr>
</form>
</center>
</table>
</body>
</html>
```

## Search.html

```
<html>
<head>
<link rel='stylesheet' href='abcd.css' />
</head>
<body>
<div id="ram">

 MENU
```

```
 INSERT
 SEARCH
 SHOW ALL
 UPDATE
 DELETE
 ABOUT
 CONTACT
 LOGOUT

</div>

<center>
<form action = "Search">
<table cellpadding = "15">
<tr>
<td> Enter Name </td>
<td><input type = "text" placeholder = "Enter Name" name = "Se" ></td>
</tr>
<tr>
<th colspan = "2" ><input type = "submit" class = "B" value = "Search" > </th>
</tr>
</table>
</form>
</center>
</body>
</html>
```

Delete.html

```
<html>
<head>
```

```

<link rel='stylesheet' href='abcd.css' />
<head>
<center>
<form action="Delete">
<table cellpadding="15">
<tr>
<td>Enter RollNo. </td>
<td><input type="text" placeholder="Enter Rollno." name="De" /> </td>
</tr>
<tr>
<th colspan="2"><input type="submit" class="B" value="Delete" /> </th>
</table>
</form>
</center>
</body>
</html>

```

### Reg.java

```

import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class Reg extends HttpServlet
{
 public void doGet(HttpServletRequest request, HttpServletResponse response) throws IOException, ServletException
 {
 PrintWriter out=response.getWriter();
 String s1=request.getParameter("un");
 String s2=request.getParameter("up");
 try
 {

```

```
Class.forName("com.mysql.cj.jdbc.Driver");
Connection con=DriverManager.getConnection("jdbc:mysql://111.91.17.2:3306/amit?useSSL=false", "root", "root");

Statement st=con.createStatement();
String s1="insert into amit values ('"+s1+"', '"+s2+"')";

st.executeUpdate(s1);
Response.sendRedirect("Login.html");
out.println("<h1> DATA INSERT </h1>");
con.close();
}

catch (Exception e)
{
 out.println(e);
}
out.close();
```

### 3 Login.java

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class Login extends HttpServlet
{
 public void doGet(HttpServletRequest request, HttpServletResponse response) throws IOException, ServletException
 {
 PrintWriter out=response.getWriter();
```

```
String s1 = request.getParameter("U1");
String s2 = request.getParameter("U2");
try {
 Class.forName("com.mysql.cj.jdbc.Driver");
 Connection con = DriverManager.getConnection("jdbc:mysql://192.168.2.100:3306/test", "root", "root");
 Statement st = con.createStatement();
 String s11 = "select * from amit1 where UNAME = '" + s1 +
 "' AND UPASS = '" + s2 + "'";
 ResultSet rs = st.executeQuery(s11);
 if (rs.next()) {
 response.sendRedirect("Menu.html");
 } else {
 out.println("invalid");
 }
 con.close();
} catch (Exception e) {
 out.println(e);
}
out.close();
```

## INSERT.JAVA

```
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.sql.*;

public class Insert extends HttpServlet
{
 public void doGet(HttpServletRequest, HttpServletResponse
 response) throws IOException, ServletException
 {
 PrintWriter out=response.getWriter();
 String s1=request.getParameter("un");
 String s2=request.getParameter("or");
 String s3=request.getParameter("up");
 String s4=request.getParameter("uc");
 String s5=request.getParameter("m");
 try
 {
 Class.forName("com.mysql.cj.jdbc.Driver");
 Connection con=DriverManager.getConnection("jdbc:mysql://
 amit?useSSL=false","root","root");
 Statement st=con.createStatement();
 String s11="insert into data2 values ('"+s1+"','"+s2+"',
 '"+s3+"','"+s4+"','"+s5+"')";
 st.executeUpdate(s11);
 out.println("<h1> DATA INSERT </h1>");
 con.close();
 }
 catch(Exception e)
 {
 out.println(e);
 out.close();
 }
 }
}
```

## ShowAll.java

```
import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class ShowAll extends HttpServlet {
 public void doGet(HttpServletRequest request, HttpServletResponse response) throws IOException {
 PrintWriter out = response.getWriter();
 out.print("<html>");
 out.print("<head>");
 out.print("<link rel='stylesheet' href='abcd.css'>");
 out.println("</head>");
 out.println("<body>");
 out.println("<div id='ram'>");
 out.print("");
 out.print(" MENU ");
 out.print(" INSERT ");
 out.print(" Search ");
 out.print(" showAll ");
 out.print(" Update ");
 out.print(" Delete ");
 out.print(" About ");
 out.print(" Contact ");
 out.print(" Logout ");
 out.print("");
 out.print("</div>");
 out.print("<div id='ram'>");

 try {

```

```
Class.forName("com.mysql.cj.jdbc.Driver");
Connection con= DriverManager.getConnection ("jdbc:mysql://
 // amit?useSSL=False", "root", "Yoot");

Statement st=con.createStatement();
ResultSet rs=st.executeQuery (sql);

out.print("<center>");

out.print("<table border='2' cellpadding='12'>");
out.print("<tr> <td>NAME </td> <td>ROLLNO </td> <td>
 CHEMISTRY </td> <td> PHYSIC </td> <td> MATH </td>
 </tr>");
while(rs.next())
{
 out.print("<tr> <td>" +rs.getString(1)+"</td>");
 out.print("<td>" +rs.getString(2)+"</td>");
 out.print("<td>" +rs.getString(3)+"</td>");
 out.print("<td>" +rs.getString(4)+"</td>");
 out.print("<td>" +rs.getString(5)+"</td></tr>");
}

out.println("</center>");

out.print("</table>");

con.close();

out.print("</body>");

out.print("</html>");

catch (Exception e)
{
 out.println(e);
}

out.close();
```

## Search.java

```
import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;
public class Search extends HttpServlet
{
 public void doGet(HttpServletRequest request, HttpServletResponse response) throws IOException
 {
 PrintWriter out=response.getWriter();
 String s1=request.getParameter("se");
 out.print("<html>");
 out.print("<head>");
 out.print("<link rel='stylesheet' href='abcd.css' />");
 out.print("<div id='yaml'>");
 try
 {
 Class.forName("com.mysql.jdbc.Driver");
 Connection con=DriverManager.getConnection("jdbc:mysql://localhost:3306/amit?useSSL=false", "root", "root");
 Statement st=con.createStatement();
 String s11="select * from data2 where UROLL='"+s1+"'";
 ResultSet rs=st.executeQuery(s11);
 out.print("<center>");
 out.print("<table border='2' cellpadding='12' >");
 if(rs.next())
 {
 out.print("<tr><td>NAME </td><td>" + rs.getString(1) +
 "</td></tr>");
 }
 out.print("<tr><td>ROLLNO </td><td>" + rs.getString(2) + "</td></tr>");
 out.print("<tr><td>PHYSICS </td><td>" + rs.getString(3) + "</td>");
```

```
out.print("<tr><td> CHEM </td> <td>" + rs.getString(4) + "
 </td> </tr>");
out.print("<tr><td> MATH </td> <td>" + rs.getString(5) + "
 </td> </tr>");
out.print("</table>");
}
else
{
 out.print("<th> INVALID ROLL </th>");
}

out.print("</center>");
con.close();
out.print("</body>");
out.print("</html>");
}
catch (Exception e)
{
 out.println(e);
}
out.close();
}
}
```

### Delete.java

```
import java.io.*;
import java.sql.*;
import javax.servlet.*;
import javax.servlet.http.*;

public class Delete extends HttpServlet
{
 public void doGet(HttpServletRequest request, HttpServletResponse response) throws IOException, ServletException
 {
 response.setContentType("text/html");
 PrintWriter out = response.getWriter();
 String id = request.getParameter("id");
 String query = "DELETE FROM STUDENT WHERE ID = " + id;
 Connection con = null;
 Statement st = null;
 try
 {
 Class.forName("com.mysql.jdbc.Driver");
 con = DriverManager.getConnection("jdbc:mysql://localhost:3306/test", "root", "root");
 st = con.createStatement();
 st.executeUpdate(query);
 out.println("Record Deleted");
 }
 catch (Exception e)
 {
 out.println("Error : " + e);
 }
 }
}
```

```
PrintWriter out = response.getWriter();
String s1 = request.getParameter("De");
out.println("<html>");
out.println("<head>");
out.print("<link rel='stylesheet' href='abcd.css' />");
out.println("</head>");
out.println("<body>");

try {
 Class.forName("com.mysql.jdbc.Driver");
 Connection con=DriverManager.getConnection("jdbc:mysql://192.168.1.10:3306/test?useSSL=false","root","");
 Statement st=con.createStatement();
 String s1="delete from data2 where UROLL='"+s1+"'";
 st.executeUpdate(s1);
 response.sendRedirect("showAll");
 con.close();
 out.println("</body>");
 out.print("</html>");

} catch (Exception e) {
 out.println(e);
}

out.close();
```

Tuesday

∴ SINGLETON PROGRAM :-

① class A

{

    Static A a;

    int x, y;

    private A()

{}

    static A getObject()

{

        if(a==null)

{

            a=new A();

}

        return a;

}

    void get(int a, int b)

{

        x=a;

        y=b;

}

    void show()

{

        System.out.println("x="+x);

        System.out.println("y="+y);

}

3

class Demo

{

    public static void main(String args[])

{

        A a1=A.getObject();

        A a2=A.getObject();

        a1.get(10,20);

        a2.show();

29/08/23

Tuesday

## ∴ SINGLETON PROGRAM ∴

① class A

{

    Static A a;

    int x, y;

    private A()

{}

    static A getObject()

{

        if (a == null)

{

            a = new A();

}

        return a;

}

O/P - x=10  
y=20

    void get(int a, int b)

2018 6 9 9 5 5 4

{

        x = a;

        y = b;

2018 6 9 9 5 5 4

}

    void show()

{

        System.out.println("x=" + x);

        System.out.println("y=" + y);

}

3

class Demo

{

    public static void main (String args[])

{

        A a1 = A.getObject();

        A a2 = A.getObject();

        a1.get(10, 20);

        a2.show();

System.out.println(a1.hashCode());  
System.out.println(a2.hashCode());

## ② class A

{

int x,y;  
void get(int a, int b)

{

x=a;

y=b;

}

void show()

{

System.out.println("x=" + x);

System.out.println("y=" + y);

o/p - x=10  
y=20

}

3

class Demo

{

public static void main(String args)

{  
A a1 = new A();  
A a2 = new A();

a1.get(10, 20);

a1.show();

System.out.println(a1.hashCode());

System.out.println(a2.hashCode());

3

3

\*\*\* What is finalization?

Ans:- Finalization :- यहाँ से unreference object के destroy  
होने से पहले उस object के

corresponding cleanup activity के purpose से J.V.M  
finalize method ko call karta hai is process ko

## Finalization कहते हैं।

"Finalization is a feature of the java programming language that allows you to perform postmortem cleanup on Object that the garbage collector has found to be unreachable."

- finalize() method object class के पास hai.
- Object class की finalize() method protected hai
- Object class की finalize() method की body empty है इसलिए अपनी requirement के according finalize method ko call karna chahiye

Syntax: protected void finalize()

मुझे इसके finalize method ko override nahi kiya hai to object class ki finalize method call hogi

isliye jis object ke corresponding cleanup activity perform karna hai उस class में finalize method ko override karna chahiye

③ class A

  {

    int x,y;

    void show()

  {

    System.out.println("x="+x);

    System.out.println("y="+y);

  }

    protected void finalize()

  {

    System.out.println("finalize method call");

|                                 |     |     |
|---------------------------------|-----|-----|
| Main<br>method<br>call<br>(MMC) | FMC | MMC |
|                                 | MMC | FMC |

\*इन तीनों में से कोई भी output आ सकता है!

class Demo

public static void main(String args)

A a1=new A();  
a1.x=10;  
a1.y=20;  
a1=null;

System.gc();

System.out.println("main method call");

3

3

④ class A

5

int x,y;

void show()

5

System.out.println("x="+x);

System.out.println("y="+y);

3

protected void finalize()

5

System.out.println("finalize method call");

3

class Demo

5

public static void main(String args) { method output  
5 } aayega; 3704

A a1=new A();

O/p- main method call

a1.x=10;

a2.y=20;

System.gc();

System.out.println("main method call");

3

3

⑤ class A

5

int x,y;

```

void show()
{
 System.out.println("x=" + x);
 System.out.println("y=" + y);
}

protected void finalize()
{
 System.out.println("finalize method call");
}

class Demo
{
 public static void main(String args)
 {
 A a1 = new A();
 String s1 = new String("ram");
 a1.x = 10;
 a1.y = 20;
 s1 = null;
 System.out.println("main method call");
 System.out.println("main method call");
 }
}

NOTE :- जो unreferenced variable hota hai or जो destroy
honed bala है उसकी finalize() method call hoti hai
उसके previous example में string class की finalize()
method nahi hai to object class की call hogi or
object class की finalize() method empty hai so
उसकी main method hi output aayega.

```

## ⑥ CLASS A

```

int x,y;
void show()

```

```

System.out.println("x=" + x);
System.out.println("y=" + y);
}
protected void finalize()
{
System.out.println("finalize method call");
}
class Demo
{
public static void main(String args)
{
A a1 = new A();
a1.x = 10;
a1.y = 20;
a1 = null;
System.gc();
System.gc();
}

```

OIP-

|     |     |     |
|-----|-----|-----|
| FNC | MNC | MNC |
| MNC | FNC |     |

⑦ class A

```

int x,y;
void show()
{
System.out.println("x=" + x);
System.out.println("y=" + y);
}
protected void finalize()
{
System.out.println("finalize method call");
}
class Demo
{

```

```
public static void main (String args)
```

```
{
```

```
A a1=new A();
A a2=new A();
```

```
a1.x=10;
```

```
a1.y=20;
```

```
a1=null;
```

```
System.gc();
```

```
a2=null();
```

```
System.gc();
```

```
System.out.println("main method call");
```

3  
3

|   |         |         |         |                  |
|---|---------|---------|---------|------------------|
| ⑧ | class A | SHH     | SHH     | -910             |
|   |         | (empty) | (empty) | main method call |

```
int x,y;
```

```
protected void finalize()
```

```
{
```

```
System.out.println("finalize method call");
```

```
System.out.println(1010);
```

3  
3

```
class Demo
```

```
{
```

```
public static void main (String args)
```

```
{
```

```
A a1=new A();
```

```
a1.x=10;
```

```
a1.y=20;
```

```
a1=null;
```

```
System.gc();
```

```
System.out.println("main method call");
```

3

there are size  
possibilities  
of the output

olp- main method

② finalize method call  
finalize method call  
main method

③ finalize method call  
main method call  
finalize method call  
④ main method  
finalize method call  
finalize method call

⑤ main method call  
finalize method call

⑥ finalize method call  
main method

NOTE:- यदि exception finalize() method में है तो exception  
आएगी ही नहीं मोड़ि finalize method ko direct  
J.V.M ने call kiyा hai to वो उस exception ko  
handle कर सकता है।

### ⑨ class A

3

int x,y;

protected void finalize()

3

System.out.println("finalize method call");

3

3 class Demo

3

public static void main(String args)

3

A a1=new A();

a1.x=10;

a1.y=20;

a1=null;

a1.finalize();

System.out.println("main method call");

3

3

Q. Kya finalize method ko manually call kar sakte hai  
ya nahi?

• YES kar sakte hai

NOTE:- जब एस g.c से request होते हैं और J.V.M finalize  
method ko call करता है तो 1 object के corresponding उस  
एवं उसी वटे finalize() method call hogi जिसके बाद object  
destroy हो जाता किश होता hogi

ii जब एस finalize method ko manually call होते हैं तो वो  
normal method की तरह work होती है और उसे व्यक्तिगत  
अप्टी call kar sakte hai

⑩ class A

```
int x,y;
protected void finalize()
```

```
{
System.out.println("Finalize method call");
}
```

class Demo

```
{
public static void main(String args)
{
A a1=new A();
a1.x=10;
a1.y=20;
a1.finalize();
a1.finalize();
}
```

```
System.out.println("main method call");
}
```

★ ————— off-finalize method call  
a1.finalize();      finalize method call  
a1.finalize();      finalize method call  
a1.finalize();      finalize method call

⑫ class A

{

int x,y;

protected void finalize()

{

```
System.out.println("Finalize method call");
```

```
System.out.println(10/0); → exception exception aane ke bad
flow execute nahi hoga
```

}

}

## Class Demo

```
public static void main (String args[])
{
 A a1 = new A();
 a1.x = 10;
 a1.y = 20;
 a1.finalize();
}
```

O/P: finalize method call

Exception : ArithmeticException

by zero

```
System.out.println ("main method call");
}
}
```

Q. Difference between user call finalize() & J.V.M call finalize()

### User call

### J.V.M call

- i) यह नितनी बार चाहे उतनी बार finalize() को call कर सकता है।
- ii) में normal method के जैसे work करती है as it is.
- iii) इसमें exception आएगी और आगे का flow execute nahi hoga
- iv) अद्वारा दोनों normal method की तरह call karna padega finalize();

- i) J.V.M एक object के corresponding बस एक ही बार finalize() को call करता है।
- ii) में unregistered variable को destroy करने को काम करती है।
- iii) इसे J.V.M call करता है। इसलिए exception आएगी ही नहीं J.V.M. handle कर लेगा।
- iv) अद्वारा दोनों G.C से request मरता पड़ता system.gc();

Q. ऐसा ज्ञानी है कि हर बार हम ही G.C को request करें unregistered object के corresponding memory destroy करने के लिए ?

तभी ऐसा विश्वास भी भजायी नहीं है कि हर बार हम ही G.C को जब G.C के पास maximum no. of object तक पहुँचता हो जायेंगे। तो G.C इतना समझदार है कि ऐसे 3 तक delete करता रहेगा। पर यह guarantee नहीं है कि निलंबने object को ही destroy करने वाले तक उपर्युक्त नहीं है।

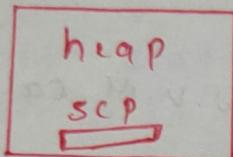
Q. G.C किस type की thread का example है?

\* G.C Daemon thread का example है

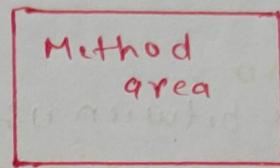
# Daemon thread :- Daemon thread in Java is "a low priority thread that runs in the background to perform tasks such as garbage collection".

Daemon thread in Java is also a service provider thread that provides services to that user thread.

STATIC VARIABLE



INSTANCE VARIABLE



\* G.C किसी heap memory के बारे में object को destroy karta hai

\* SCP (String Constant Pool) के बारे में object को destroy nahi karta hai

Q. What is heap memory?

HEAP MEMORY :- Storage Area of static variable in Java Method area section was used to store static variables of the class. whereas, non-static methods are variables were stored in the heap memory.

After the java 8 version, static variables are stored in the heap memory;

What is SCP (String Constant Pool) memory?

SCP (String Constant Pool) the java String constant pool is an area in heap memory where Java stores literal string values."

the heap is an area of memory used for run-time operations. When a new variable is created and given a value, Java checks to see if that exact value exists in the pool.

31/08/23

thursday

① class A

{

int x,y;

protected void finalize()

{

System.out.println("finalize method call");

}

3

class Demo

{

public static void main(String args)

{

A a1=new A();

a1.x=10;

a1.y=20;

a1=null;

Runtime.getRuntime().gc();

System.out.println("main method call");

3

Singleton class

factory method

★ Runtime r=Runtime.getRuntime();

r.gc();

similar class to  
object to return  
directly!

② class A

{

int x,y;

protected void finalize()

{

System.out.println("finalize method call");

3

class Demo

|     |     |     |
|-----|-----|-----|
| MNC | FMC | MNC |
| FMC | MNC | -   |

```
A a1=new A();
```

```
a1.x=10;
```

```
a1.y=20;
```

```
a1=null;
```

|     |     |     |
|-----|-----|-----|
| NMC | FMC | HMC |
| FMC | NMC |     |

```
Runtime r=Runtime.getRuntime();
```

```
r.gc();
```

```
System.out.println("main method call");
```

3  
3

Example थीमिंग ऐसी नॉव सी class है जो Singleton है:-

\* Runtime class है जो Singleton है!

③ class A

3

```
int x,y;
```

```
protected void finalize()
```

3

```
System.out.println("finalize method call");
```

3  
3

```
class Demo
```

3

```
public static void main(String args)
```

3

```
A a1=new A();
```

```
a1.x=10;
```

```
a1.y=20;
```

```
a1=null;
```

→ मेरे singleton है object  
एकी बात सहते हैं!

```
Runtime r=new Runtime();
```

```
r.gc();
```

```
System.out.println("main method call");
```

\* Runtime, Singleton class है, इसका अपना उपयोग  
constructor private होगा