Loops

Q. what is loop?

Loop means if we want to perform some task repeat again and again called as loop.
If we want to work with loop we have two types of loop.

1. Entry Control Loop: entry control loop means first check the condition and after that decide loop will be execute or not called as entry control loop.

There are two types of loop in entry control

1. While loop
2. for loop

2. Exit Control Loop: Exit control loop means first execute loop and after that check the condition called as exit control loop.

1. do while loop

If we want to work with any loop we have three major important points.

1. Initialization: initialization means decide from loop will be start called as initialization.

2. Condition: condition means to decide how many times loop will be executed or decide number of iteration

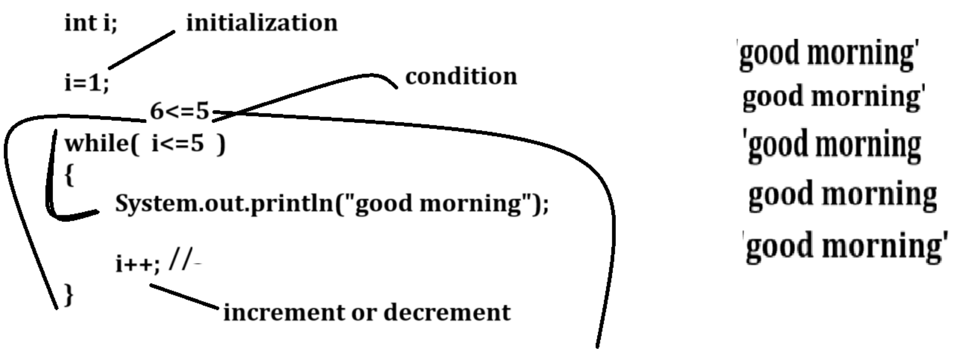
3. Increment or decrement: steps to increase the value and decrease the value or gap between every step called as increment or decrement

Now we want to discuss about while loop

Syntax:

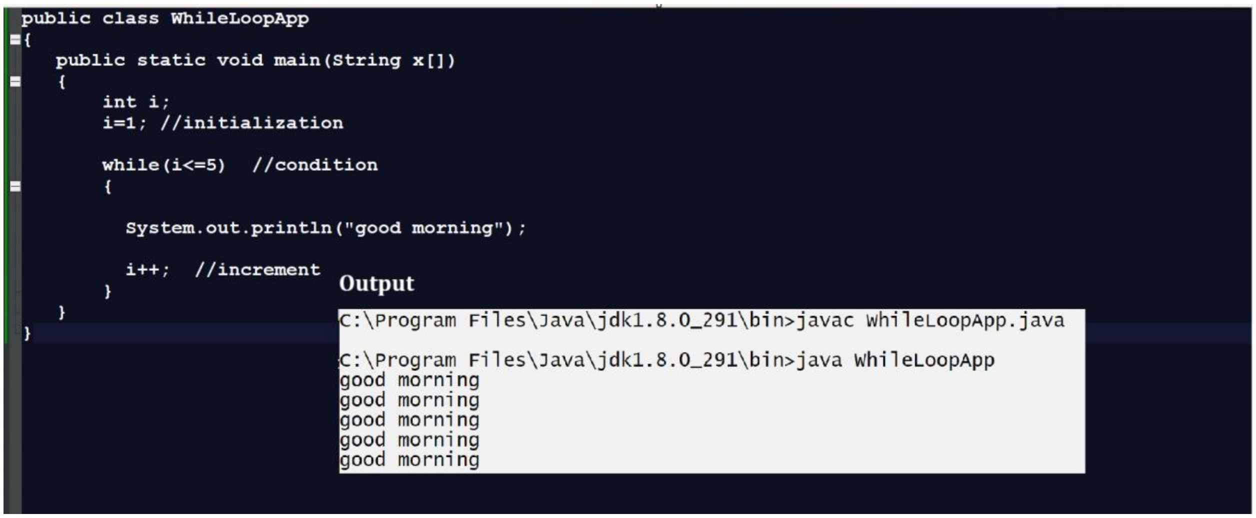
```
Initialization;
while (condition )
{      Write here logics
        Increment or decrement;
}
```

Example: WAP to print good morning five times



```
int i;
i=1;
while( i<=5 )
{
    System.out.println("good morning");
    i++; //-
}
```

```
'good morning'
good morning'
'good morning
good morning
'good morning'
```

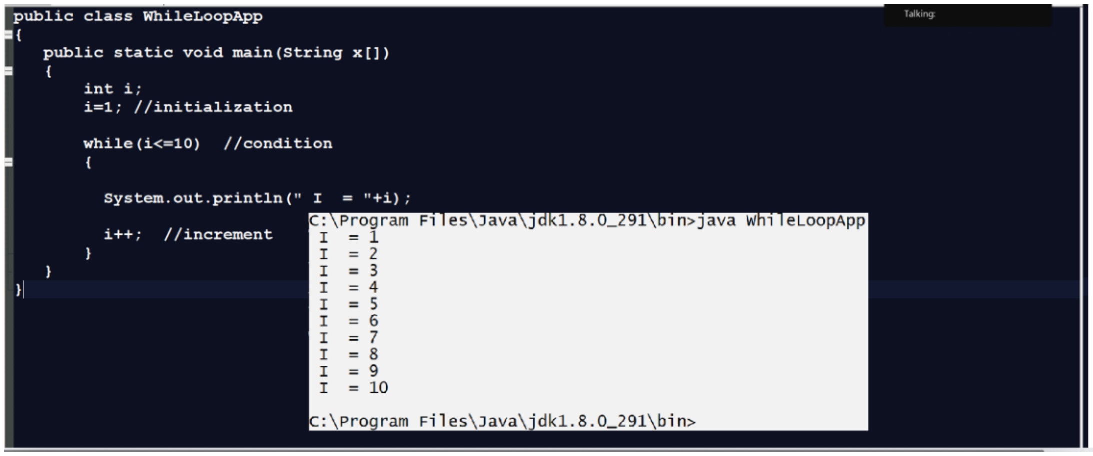


```
public class WhileLoopApp
{
   public static void main(String x[])
   {
      int i;
      i=1; //initialization

      while(i<=5)  //condition
      {

        System.out.println("good morning");

        i++;  //increment
      }
   }
}
```

Output

```
C:\Program Files\Java\jdk1.8.0_291\bin>javac WhileLoopApp.java

C:\Program Files\Java\jdk1.8.0_291\bin>java WhileLoopApp
good morning
good morning
good morning
good morning
good morning
```

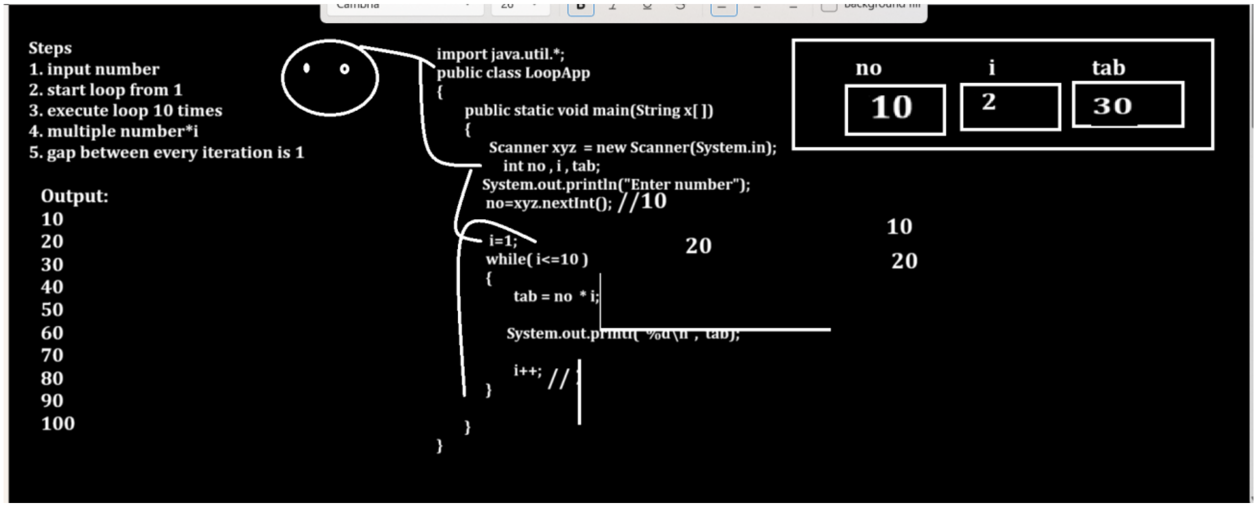
Example: WAP to print the 1 to 10 values while loop?



```
public class WhileLoopApp
{
   public static void main(String x[])
   {
      int i;
      i=1; //initialization

      while(i<=10)  //condition
      {

        System.out.println(" I  = "+i);

        i++;  //increment
      }
   }
}
```

```
C:\Program Files\Java\jdk1.8.0_291\bin>java WhileLoopApp
 I  = 1
 I  = 2
 I  = 3
 I  = 4
 I  = 5
 I  = 6
 I  = 7
 I  = 8
 I  = 9
 I  = 10

C:\Program Files\Java\jdk1.8.0_291\bin>
```

Example: WAP to input number and print its table.

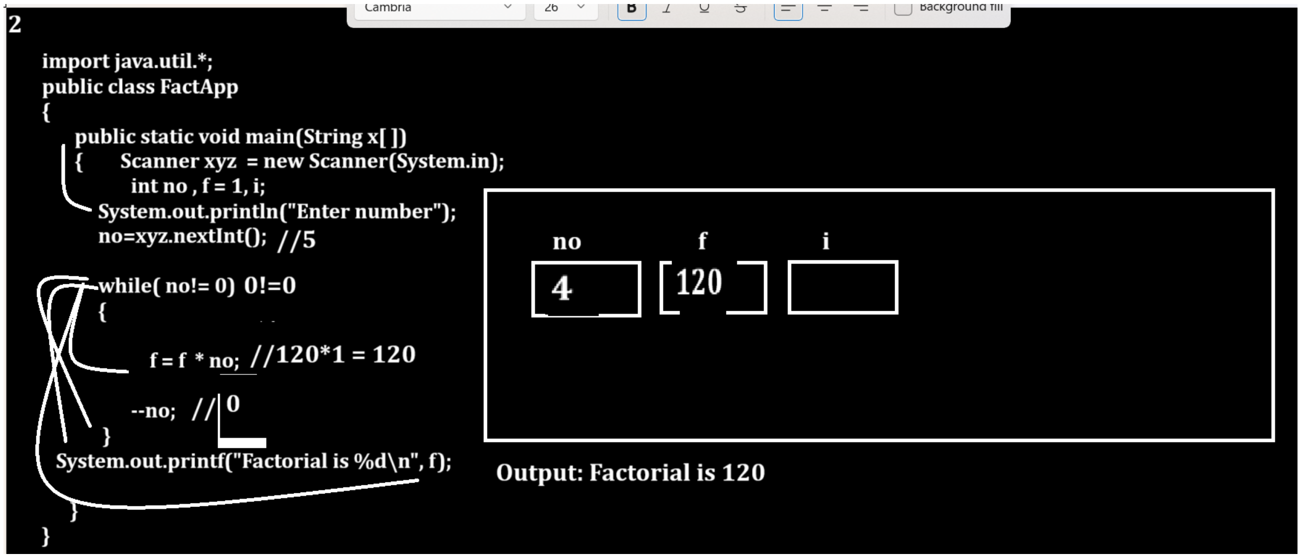


Steps
1. input number
2. start loop from 1
3. execute loop 10 times
4. multiple number*i
5. gap between every iteration is 1

Output:
10
20
30
40
50
60
70
80
90
100

```
import java.util.*;
public class LoopApp
{
    public static void main(String x[ ])
    {
      Scanner xyz  = new Scanner(System.in);
        int no , i , tab;
     System.out.println("Enter number");
     no=xyz.nextInt(); //10

     i=1;
     while( i<=10 )
     {
         tab = no  * i;

        System.out.printf("%d\n", tab);

         i++; //
     }

    }
}
```

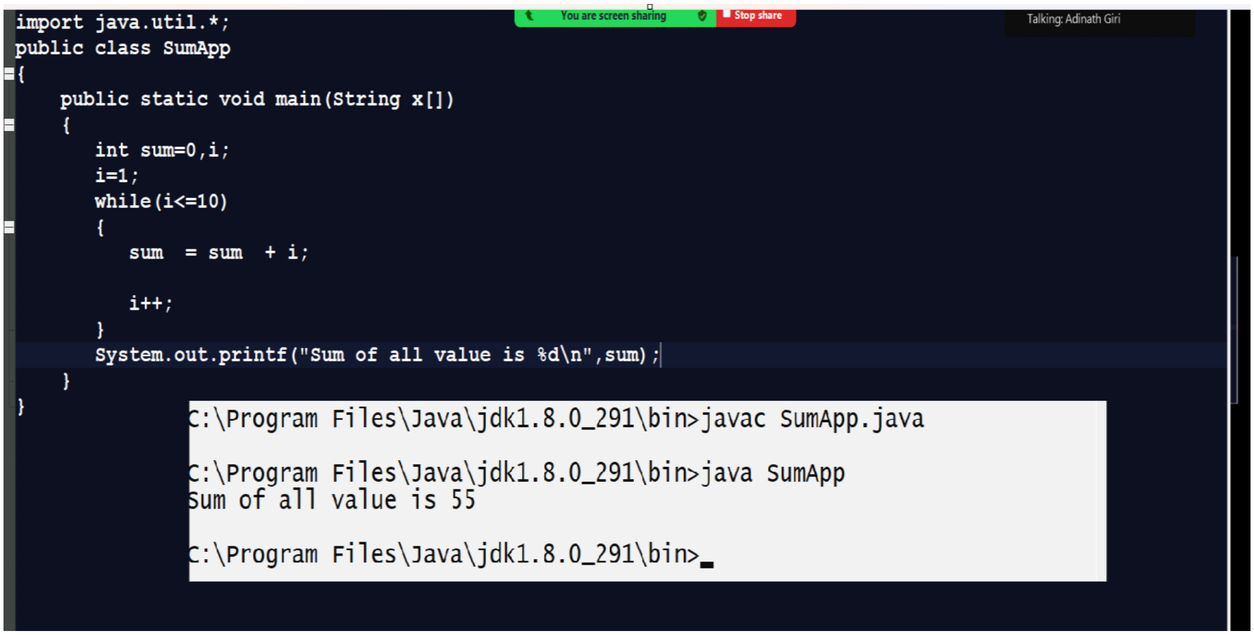
Example: WAP to input number and calculate its factorial?



```
import java.util.*;
public class FactApp
{
    public static void main(String x[ ])
    {     Scanner xyz  = new Scanner(System.in);
            int no , f = 1, i;
        System.out.println("Enter number");
        no=xyz.nextInt(); //5

        while( no!= 0) 0!=0
        {

            f = f * no; //120*1 = 120

           --no; //0
        }
    System.out.printf("Factorial is %d\n", f);
     }
}
```

Output: Factorial is 120

Example: WAP to calculate sum of all natural number between 1 to 10?



```
import java.util.*;
public class SumApp
{
    public static void main(String x[])
    {
       int sum=0,i;
       i=1;
       while(i<=10)
       {
          sum  = sum  + i;

          i++;
       }
       System.out.printf("Sum of all value is %d\n",sum);
    }
}
```

```
C:\Program Files\Java\jdk1.8.0_291\bin>javac SumApp.java

C:\Program Files\Java\jdk1.8.0_291\bin>java SumApp
Sum of all value is 55

C:\Program Files\Java\jdk1.8.0_291\bin>
```

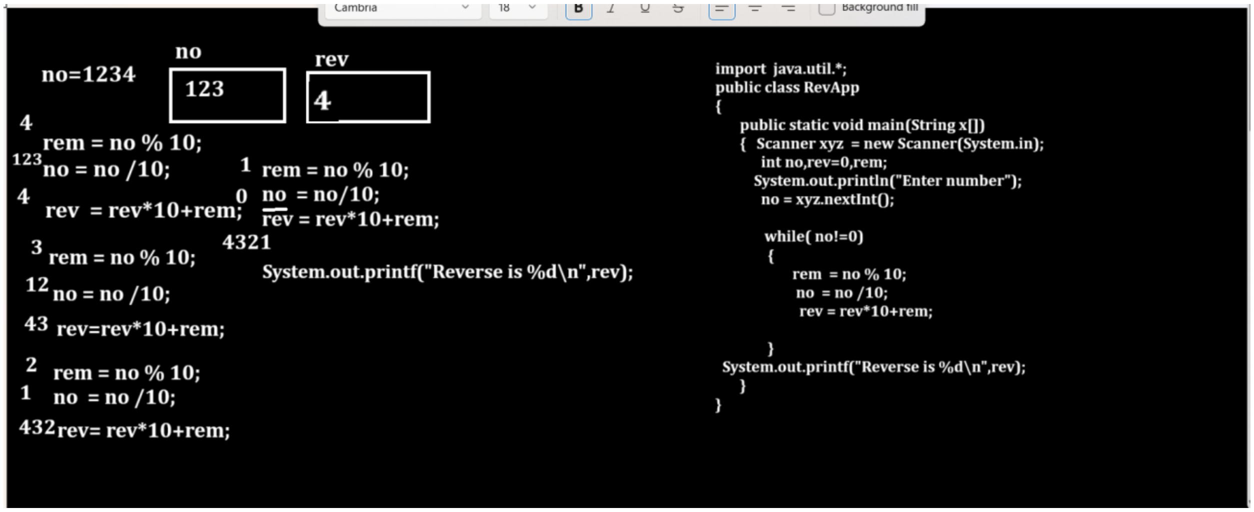
Example: WAP to input number and reverse it?

Input: 1234

Output: 4321

Input: 12345

Output: 54321



```
no=1234        no              rev
               123             4
4
rem = no % 10;
123 no = no /10;            1  rem = no % 10;
4                           0  no  = no/10;
 rev  = rev*10+rem;            rev = rev*10+rem;
                         4321
3 rem = no % 10;               System.out.printf("Reverse is %d\n",rev);
12 no = no /10;
43 rev=rev*10+rem;

2  rem = no % 10;
1  no  = no /10;
432rev= rev*10+rem;
```

```
import  java.util.*;
public class RevApp
{
     public static void main(String x[])
     {  Scanner xyz  = new Scanner(System.in);
         int no,rev=0,rem;
        System.out.println("Enter number");
         no = xyz.nextInt();

         while( no!=0)
         {
              rem  = no % 10;
              no  = no /10;
               rev = rev*10+rem;

         }
  System.out.printf("Reverse is %d\n",rev);
     }
}
```

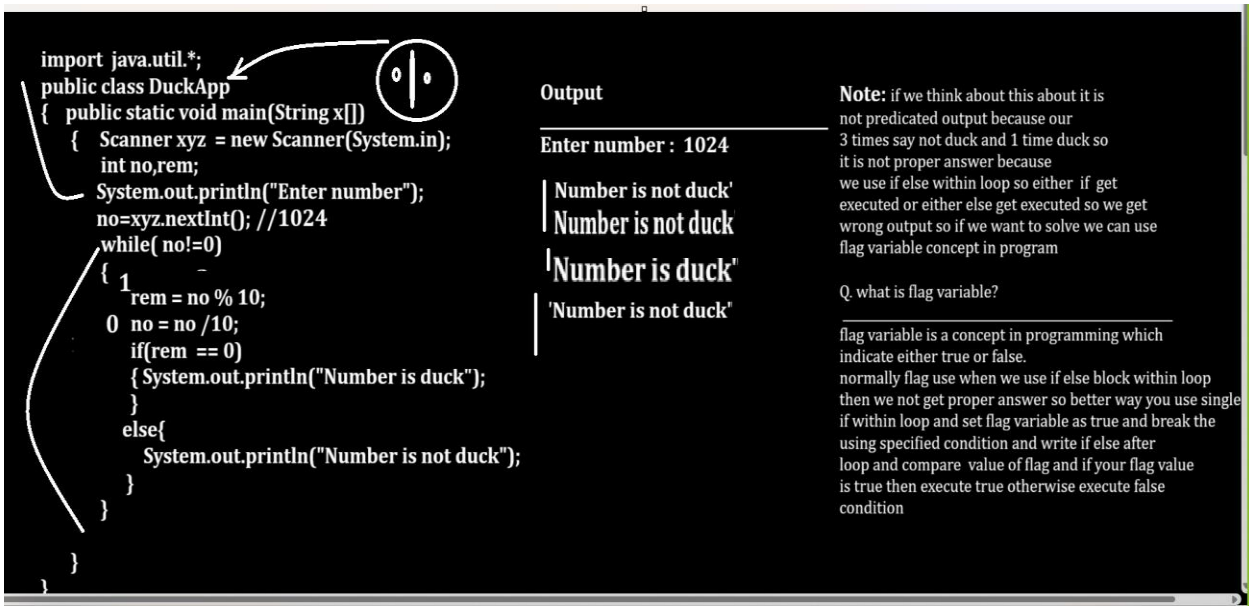
Example: WAP to input number and check number is duck or not

Duck number means number contain 0 called as duck

Input: 1024 - it is duck number

Input: 1234 – it is not duck number.

Code without using flag variable concept



```
import  java.util.*;
public class DuckApp
{   public static void main(String x[])
    {   Scanner xyz  = new Scanner(System.in);
        int no,rem;
       System.out.println("Enter number");
       no=xyz.nextInt(); //1024
        while( no!=0)
        { 1
          rem = no % 10;
        0  no = no /10;
           if(rem  == 0)
           { System.out.println("Number is duck");
           }
          else{
             System.out.println("Number is not duck");
           }
        }

    }
}
```

Output

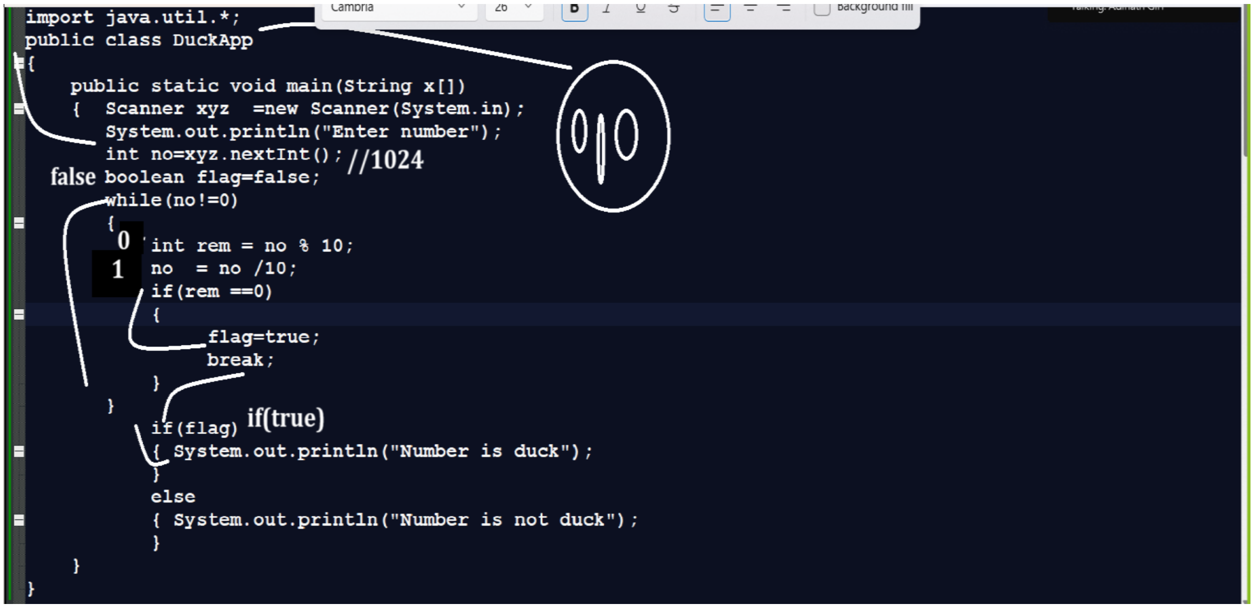
```
Enter number :  1024

Number is not duck'
Number is not duck
Number is duck'
'Number is not duck'
```

Note: if we think about this about it is not predicated output because our 3 times say not duck and 1 time duck so it is not proper answer because we use if else within loop so either if get executed or either else get executed so we get wrong output so if we want to solve we can use flag variable concept in program

Q. what is flag variable?

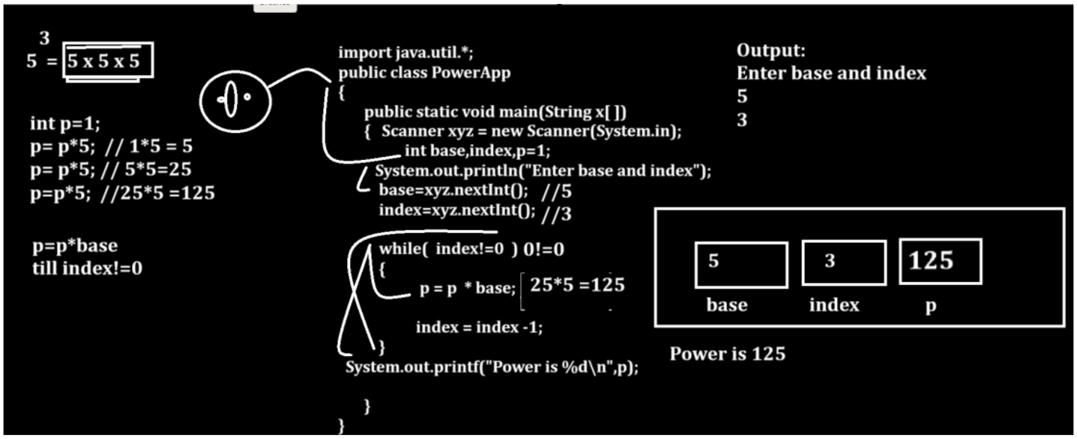
flag variable is a concept in programming which indicate either true or false.
normally flag use when we use if else block within loop then we not get proper answer so better way you use single if within loop and set flag variable as true and break the using specified condition and write if else after loop and compare value of flag and if your flag value is true then execute true otherwise execute false condition

Q. What will be output of given code?



```
import java.util.*;
public class DuckApp
{
    public static void main(String x[])
    {   Scanner xyz  =new Scanner(System.in);
        System.out.println("Enter number");
        int no=xyz.nextInt(); //1024
false   boolean flag=false;
        while(no!=0)
        {
      0     int rem = no % 10;
      1     no  = no /10;
            if(rem ==0)
            {
                flag=true;
                break;
            }
        }
            if(flag) if(true)
            { System.out.println("Number is duck");
            }
            else
            { System.out.println("Number is not duck");
            }
    }
}
```

Example: WAP to input the two values consider first value as base and second value as index and calculate power of number?



5^3 = 5 x 5 x 5

```
int p=1;
p= p*5;  // 1*5 = 5
p= p*5; // 5*5=25
p=p*5;  //25*5 =125

p=p*base
till index!=0
```

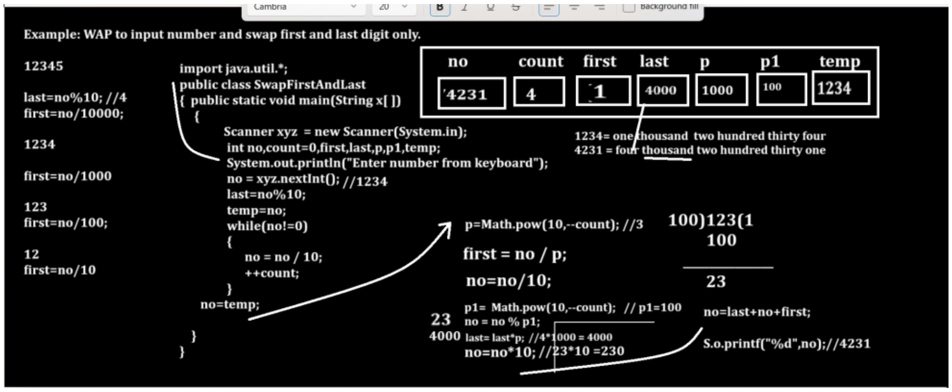
```
import java.util.*;
public class PowerApp
{
   public static void main(String x[ ])
   {  Scanner xyz = new Scanner(System.in);
         int base,index,p=1;
      System.out.println("Enter base and index");
      base=xyz.nextInt();  //5
      index=xyz.nextInt(); //3

      while( index!=0 ) 0!=0
      {
           p = p  * base;  25*5 =125

          index = index -1;
      }
    System.out.printf("Power is %d\n",p);

   }
}
```

```
Output:
Enter base and index
5
3
```

base	index	p
5	3	125

```
Power is 125
```

Example: WAP to input number and swap its first and last digit?



Example: WAP to input number and swap first and last digit only.

```
12345

last=no%10; //4
first=no/10000;

1234

first=no/1000

123
first=no/100;

12
first=no/10
```

```
import java.util.*;
public class SwapFirstAndLast
{  public static void main(String x[ ])
   {
        Scanner xyz  = new Scanner(System.in);
        int no,count=0,first,last,p,p1,temp;
        System.out.println("Enter number from keyboard");
        no = xyz.nextInt(); //1234
        last=no%10;
        temp=no;
        while(no!=0)
        {
            no = no / 10;
            ++count;
        }
     no=temp;

   }
}
```

no	count	first	last	p	p1	temp
'4231	4	1	4000	1000	100	1234

1234= one thousand two hundred thirty four
4231 = four thousand two hundred thirty one

```
      p=Math.pow(10,--count); //3
      first = no / p;
      no=no/10;
      p1=  Math.pow(10,--count);   // p1=100
23    no = no % p1;
4000  last= last*p; //4*1000 = 4000
      no=no*10; //23*10 =230
```

```
100)123(1
    100
    ----
     23
```

```
no=last+no+first;

S.o.printf("%d",no);//4231
```

Example with source code

```
import java.util.*;
public class SwapFirstAndLastDigit
{
  public static void main(String x[])
       {
         Scanner xyz  = new Scanner(System.in);
              int no,temp,count=0,p,p1,first,last;
              System.out.println("Enter number");
              no=xyz.nextInt();
              temp=no;
              System.out.printf("\nBefore swapping first and last digit %d\n",no);
              while(no!=0)
              {  no  = no /10;
                ++count;
              }
              no=temp;
              last=no%10;
              p=((int)Math.pow((double)10,(double)(--count)));
              first=no/p;
              no = no /10;
              p1=((int)Math.pow((double)10,(double)(--count)));
              no=no%p1;
              last=last*p;
              no = no *10;
              no=last+no+first;
              System.out.printf("\nAfter swapping first and last digit %d\n",no);
       }
}
```

Example: WAP to input number from keyboard and input search digit from keyboard and check digit present in number or not

Output:

Input number:

12345

Input Digit for search

3

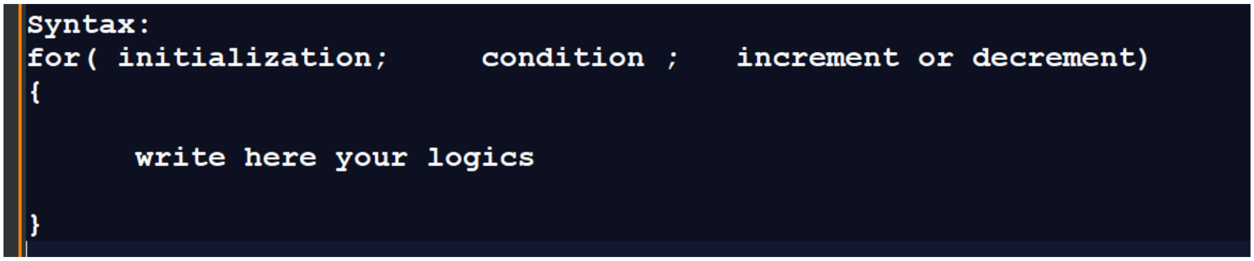
Output: Digit found

for loop and nested for loop or nested loop

---

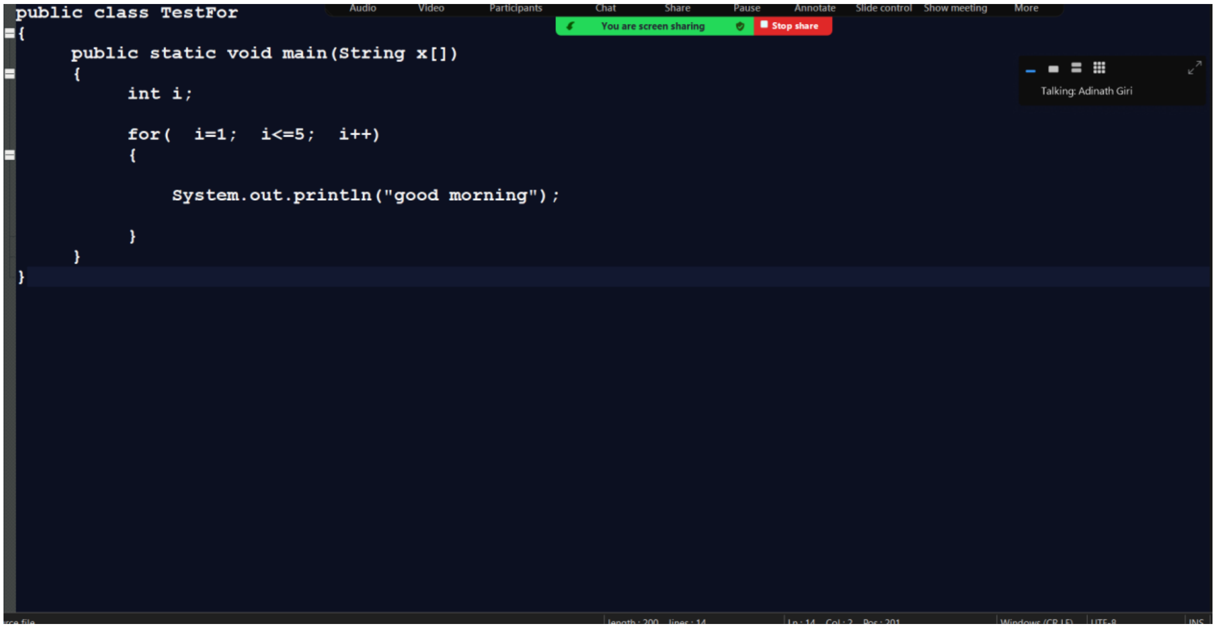
for loop is same like a while loop.

Syntax:



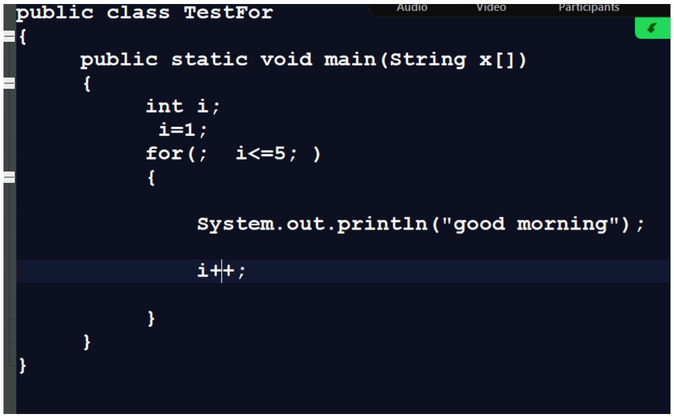
```
Syntax:
for( initialization;     condition ;    increment or decrement)
{

       write here your logics

}
```

Example: print good morning five times using for loop.



```
public class TestFor
{
     public static void main(String x[])
     {
          int i;

          for(  i=1;  i<=5;  i++)
          {

              System.out.println("good morning");

          }
     }
}
```

You can use for loop like as



```
public class TestFor
{
     public static void main(String x[])
     {
          int i;
           i=1;
          for(;  i<=5; )
          {

              System.out.println("good morning");

              i++;

          }
     }
}
```

Example: WAP to input number from keyboard and reverse it using for loop

Input : no=1234
Output: 4321

```
import java.util.*;
public class TestFor
{
    public static void main(String x[])
            {
                Scanner xyz  = new Scanner(System.in);
                        int no,rem,rev=0;

                        System.out.println("Enter number");
                        no = xyz.nextInt();

                        for(;no!=0;)
                        {       rem = no % 10;
                               no  = no /10;
                              rev=rev*10+rem;
                        }
                        System.out.printf("Reverse is %d\n",rev);
            }
}
```

Example: WAP to input the base and index from keyboard and calculate the power of number using for loop.

Example with source code

```
import java.util.*;
public class TestFor
{
    public static void main(String x[])
            {
                Scanner xyz  = new Scanner(System.in);
                        int base,index,p=1;
                        System.out.println("Enter base and index from keyboard");
                        base=xyz.nextInt();
                        index=xyz.nextInt();
                        for(int i=1; i<=index;i++)
                        {
                                     p = p *base;
                        }
                        System.out.printf("Power is %d\n",p);
                }
}
```

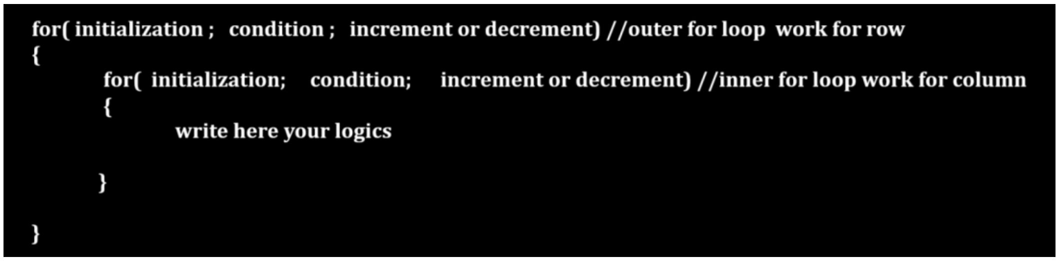
Nested for loop

____

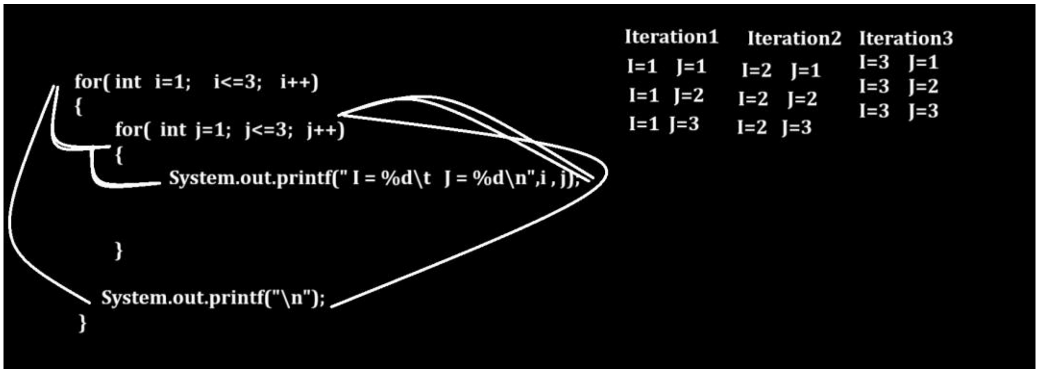
Nested for loop means for within for called as nested for loop so nested for loop specially design for work with matrix and in the case of nested for loop if outer for loop execute once then inner for loop execute all times

Syntax of nested for loop

____



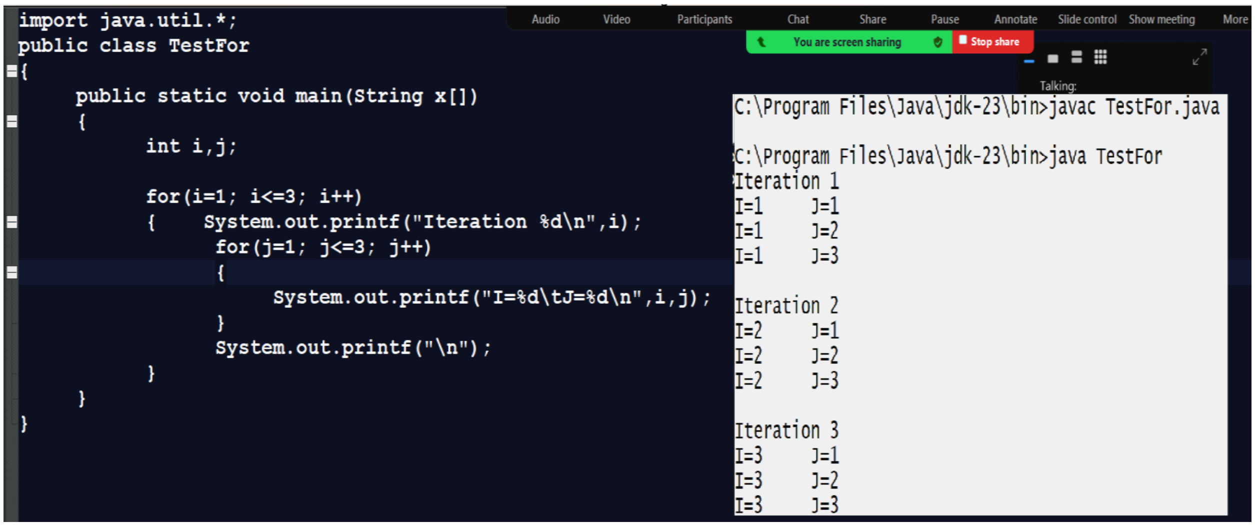
```
for( initialization ;  condition ;  increment or decrement) //outer for loop  work for row
{
        for(  initialization;    condition;    increment or decrement) //inner for loop work for column
        {
                write here your logics

        }

}
```



```
for( int  i=1;   i<=3;   i++)
{
     for(  int  j=1;  j<=3;  j++)
     {
          System.out.printf(" I = %d\t   J = %d\n",i , j);


     }
   System.out.printf("\n");
}
```

```
Iteration1     Iteration2    Iteration3
I=1   J=1      I=2   J=1     I=3   J=1
I=1   J=2      I=2   J=2     I=3   J=2
I=1   J=3      I=2   J=3     I=3   J=3
```

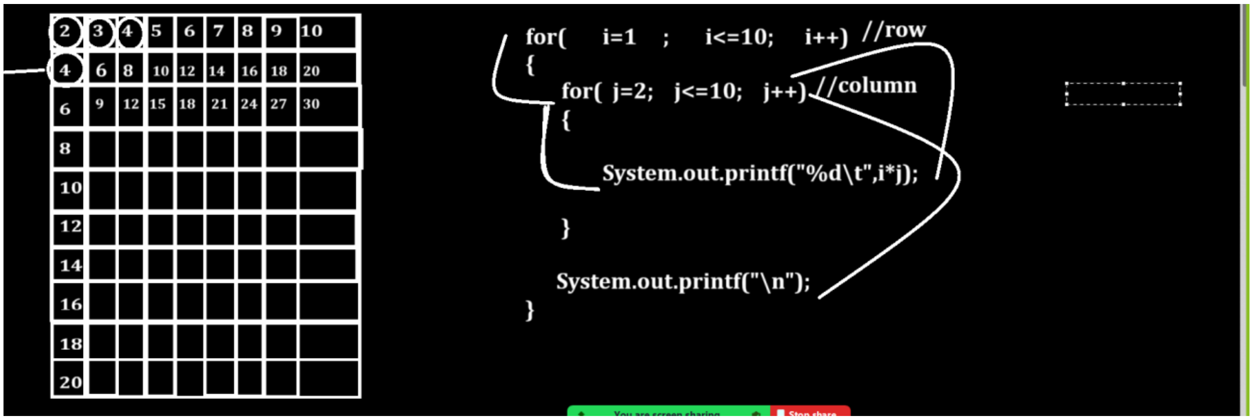
Example with source code



```
import java.util.*;
public class TestFor
{
     public static void main(String x[])
     {
          int i,j;

          for(i=1; i<=3; i++)
          {     System.out.printf("Iteration %d\n",i);
                 for(j=1; j<=3; j++)
                 {
                      System.out.printf("I=%d\tJ=%d\n",i,j);
                 }
                 System.out.printf("\n");
          }
     }
}
```

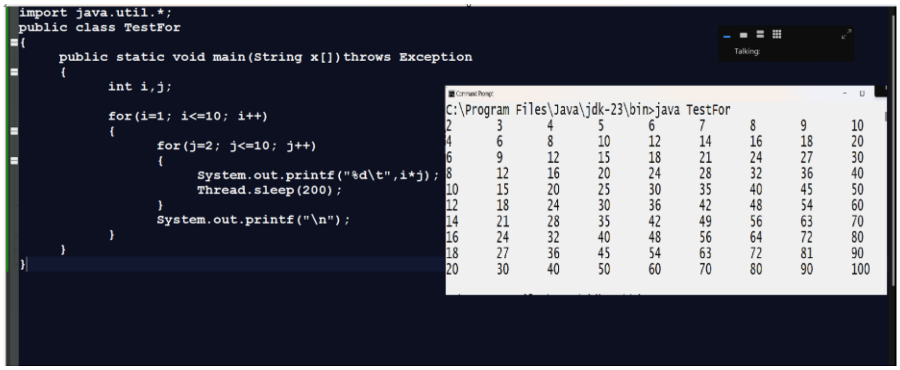
```
C:\Program Files\Java\jdk-23\bin>javac TestFor.java

C:\Program Files\Java\jdk-23\bin>java TestFor
Iteration 1
I=1     J=1
I=1     J=2
I=1     J=3

Iteration 2
I=2     J=1
I=2     J=2
I=2     J=3

Iteration 3
I=3     J=1
I=3     J=2
I=3     J=3
```

Example: WAP to print the all tables between 2 to 10?



2	3	4	5	6	7	8	9	10
4	6	8	10	12	14	16	18	20
6	9	12	15	18	21	24	27	30
8								
10								
12								
14								
16								
18								
20								

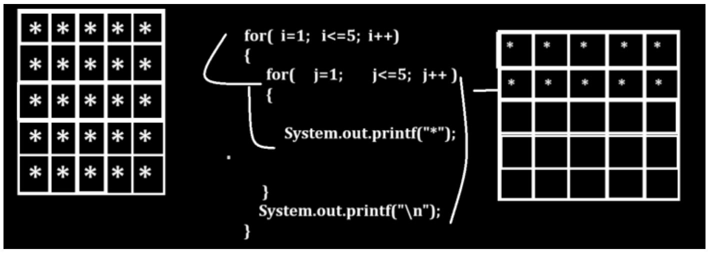
```
for(   i=1   ;   i<=10;   i++) //row
{
   for( j=2;  j<=10;  j++)//column
   {

      System.out.printf("%d\t",i*j);

   }

   System.out.printf("\n");
}
```



```
import java.util.*;
public class TestFor
{
    public static void main(String x[])throws Exception
    {
        int i,j;

        for(i=1; i<=10; i++)
        {
            for(j=2; j<=10; j++)
            {
                System.out.printf("%d\t",i*j);
                Thread.sleep(200);
            }
            System.out.printf("\n");
        }
    }
}
```

```
C:\Program Files\Java\jdk-23\bin>java TestFor
2       3       4       5       6       7       8       9       10
4       6       8       10      12      14      16      18      20
6       9       12      15      18      21      24      27      30
8       12      16      20      24      28      32      36      40
10      15      20      25      30      35      40      45      50
12      18      24      30      36      42      48      54      60
14      21      28      35      42      49      56      63      70
16      24      32      40      48      56      64      72      80
18      27      36      45      54      63      72      81      90
20      30      40      50      60      70      80      90      100
```

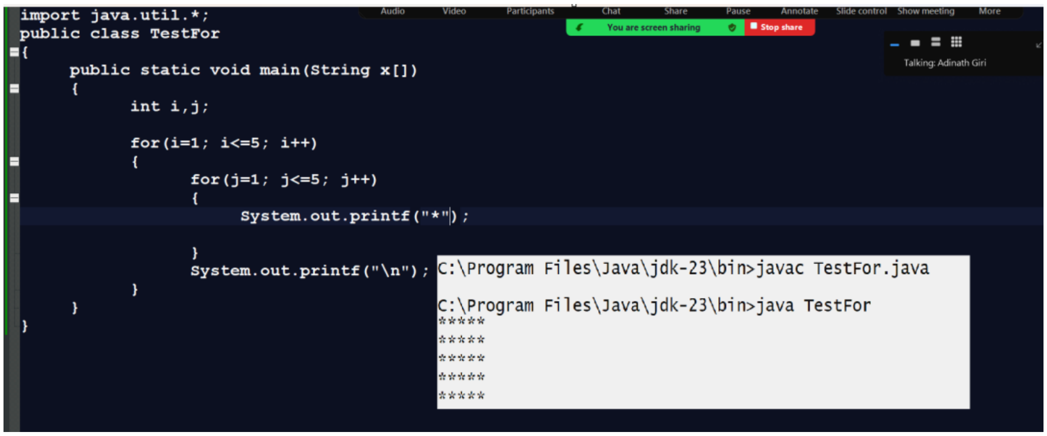
Example: WAP to print the following pattern?



*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*
*	*	*	*	*

```
for( i=1;  i<=5;  i++)
{
   for(   j=1;     j<=5;  j++)
   {

      System.out.printf("*");


   }
   System.out.printf("\n");
}
```

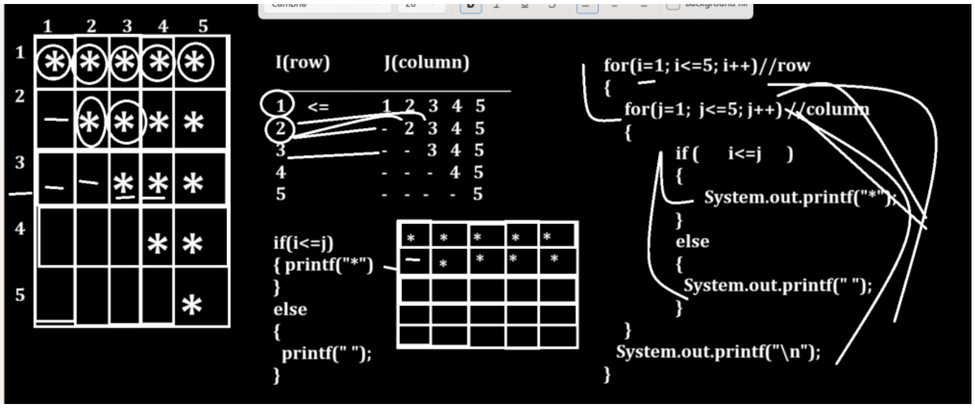
*	*	*	*	*
*	*	*	*	*



```
import java.util.*;
public class TestFor
{
    public static void main(String x[])
    {
        int i,j;

        for(i=1; i<=5; i++)
        {
            for(j=1; j<=5; j++)
            {
                System.out.printf("*");

            }
            System.out.printf("\n");
        }
    }
}
```

```
C:\Program Files\Java\jdk-23\bin>javac TestFor.java

C:\Program Files\Java\jdk-23\bin>java TestFor
*****
*****
*****
*****
*****
```

Example: WAP to print the following patterns



	1	2	3	4	5
1	*	*	*	*	*
2	—	*	*	*	*
3	—	—	*	*	*
4				*	*
5					*

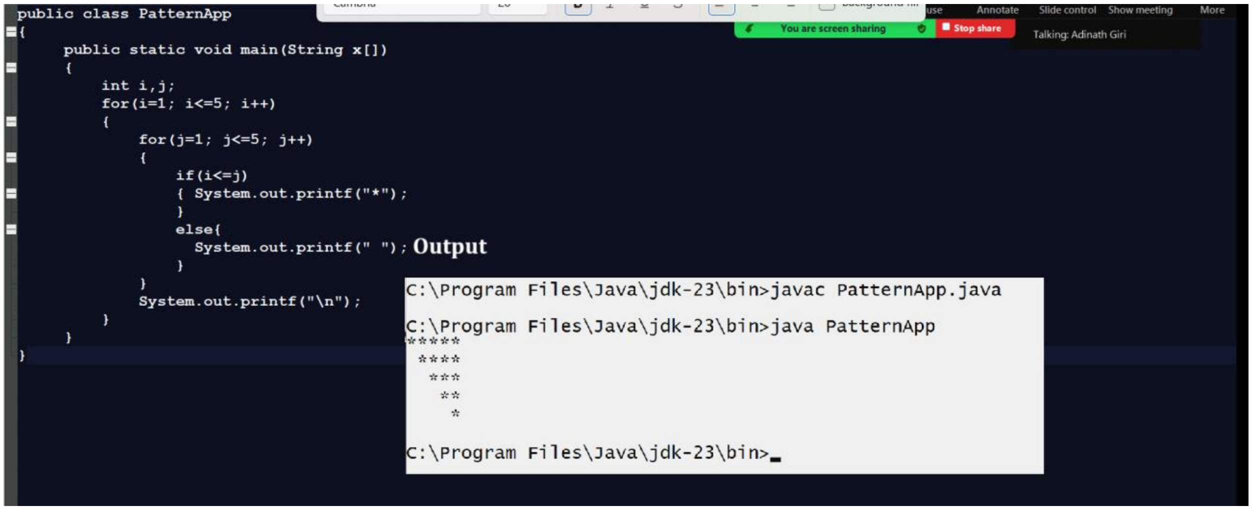
I(row)		J(column)				
1	<=	1	2	3	4	5
2		-	2	3	4	5
3		-	-	3	4	5
4		-	-	-	4	5
5		-	-	-	-	5

```
if(i<=j)
{ printf("*")
}
else
{
 printf(" ");
}
```

*	*	*	*	*
—	*	*	*	*

```
for(i=1; i<=5; i++)//row
{
  for(j=1; j<=5; j++)//column
  {
     if (    i<=j    )
     {
       System.out.printf("*");
     }
     else
     {
      System.out.printf(" ");
     }
  }
 System.out.printf("\n");
}
```

Example with source code

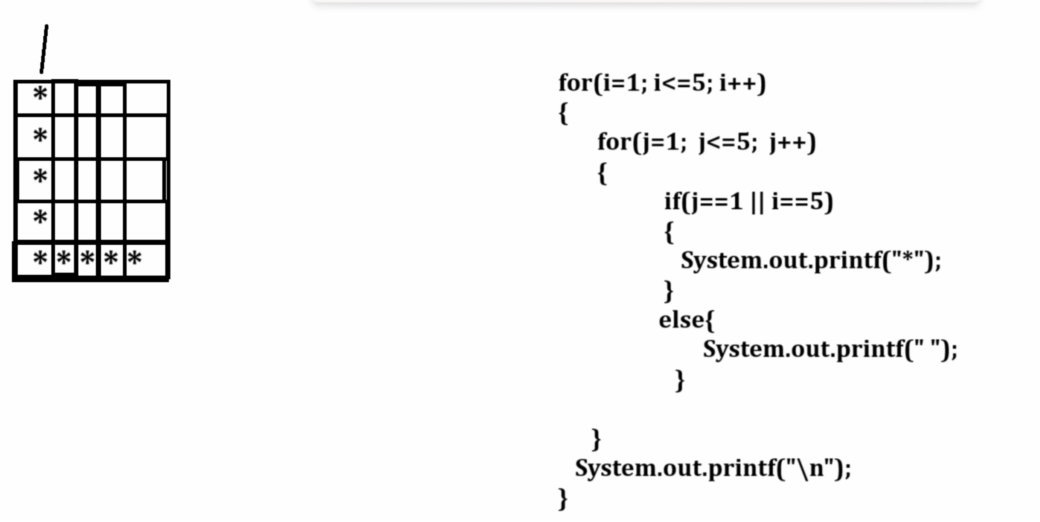


```
public class PatternApp
{
     public static void main(String x[])
     {
          int i,j;
          for(i=1; i<=5; i++)
          {
               for(j=1; j<=5; j++)
               {
                    if(i<=j)
                    { System.out.printf("*");
                    }
                    else{
                      System.out.printf(" ");
                    }
               }
               System.out.printf("\n");
          }
     }
}
```

Output

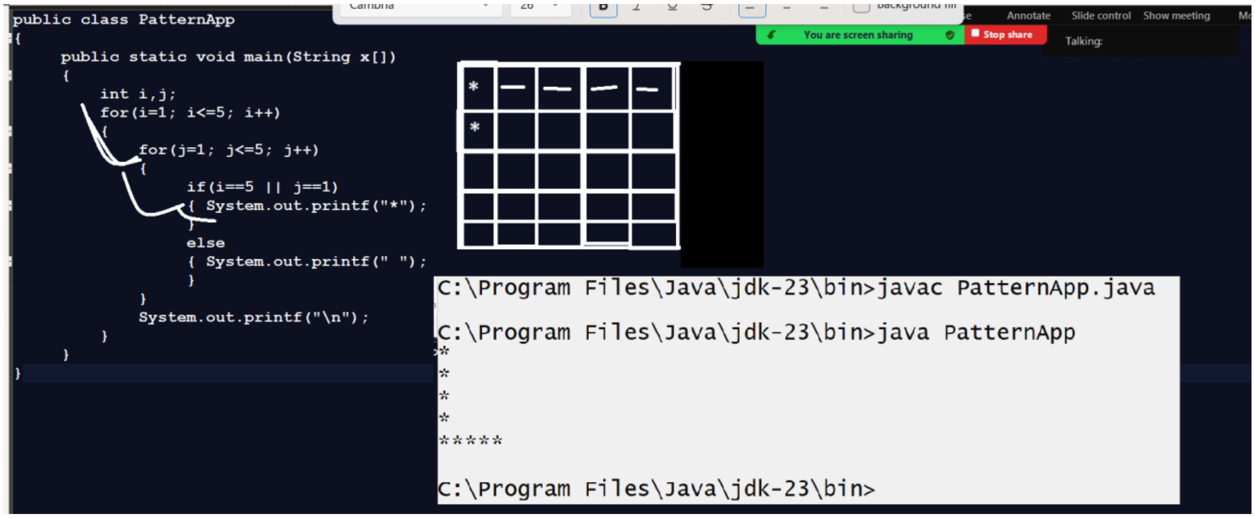
```
C:\Program Files\Java\jdk-23\bin>javac PatternApp.java

C:\Program Files\Java\jdk-23\bin>java PatternApp
*****
 ****
  ***
   **
    *

C:\Program Files\Java\jdk-23\bin>
```

Example: WAP to print the following pattern?



*				
*				
*				
*				
*	*	*	*	*

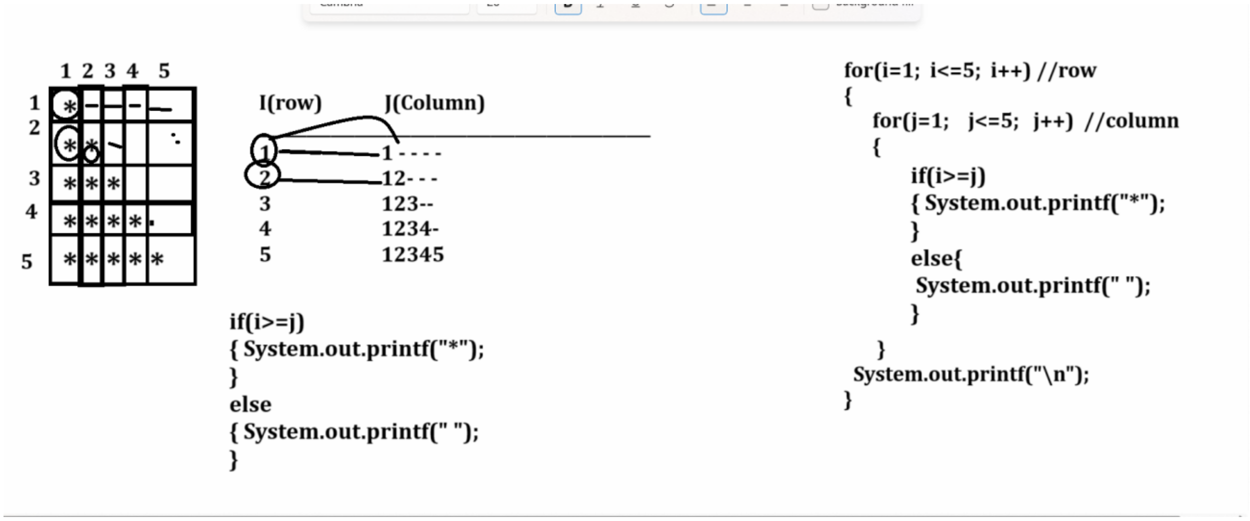
```
for(i=1; i<=5; i++)
{
    for(j=1;  j<=5;  j++)
    {
          if(j==1 || i==5)
          {
            System.out.printf("*");
          }
         else{
              System.out.printf(" ");
           }

     }
  System.out.printf("\n");
}
```



```
public class PatternApp
{
    public static void main(String x[])
    {
        int i,j;
        for(i=1; i<=5; i++)
        {
            for(j=1; j<=5; j++)
            {
                if(i==5 || j==1)
                { System.out.printf("*");
                }
                else
                { System.out.printf(" ");
                }
            }
            System.out.printf("\n");
        }
    }
}
```

```
C:\Program Files\Java\jdk-23\bin>javac PatternApp.java

C:\Program Files\Java\jdk-23\bin>java PatternApp
*
*
*
*
*****

C:\Program Files\Java\jdk-23\bin>
```

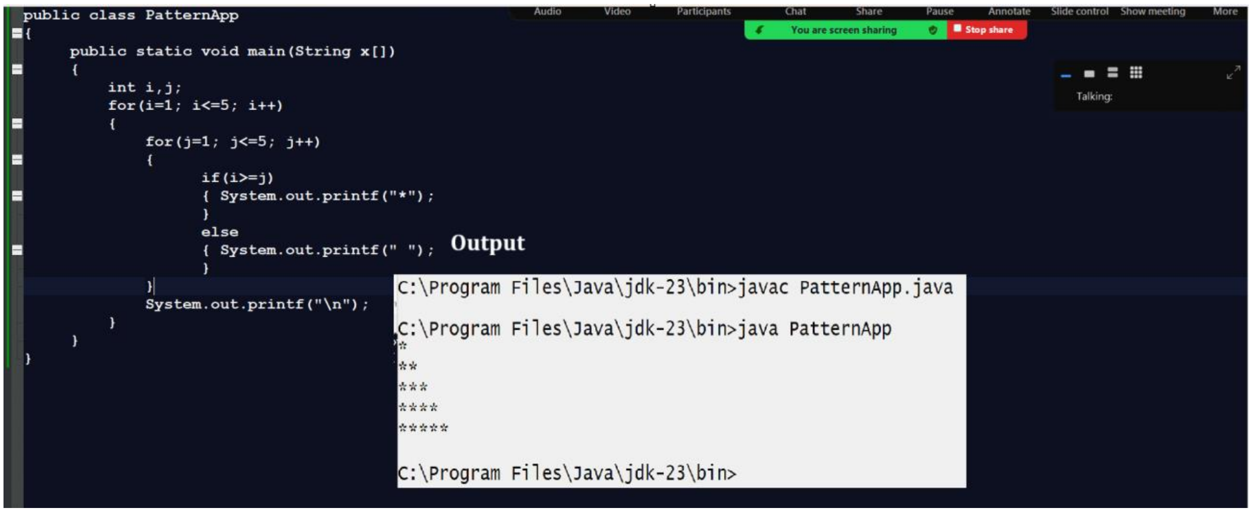
Example: WAP to print the following patterns?



```
I(row)     J(Column)
1          1----
2          12---
3          123--
4          1234-
5          12345
```

```
if(i>=j)
{ System.out.printf("*");
}
else
{ System.out.printf(" ");
}
```

```
for(i=1; i<=5; i++) //row
{
   for(j=1;  j<=5;  j++) //column
   {
       if(i>=j)
       { System.out.printf("*");
       }
       else{
        System.out.printf(" ");
       }
   }
 System.out.printf("\n");
}
```

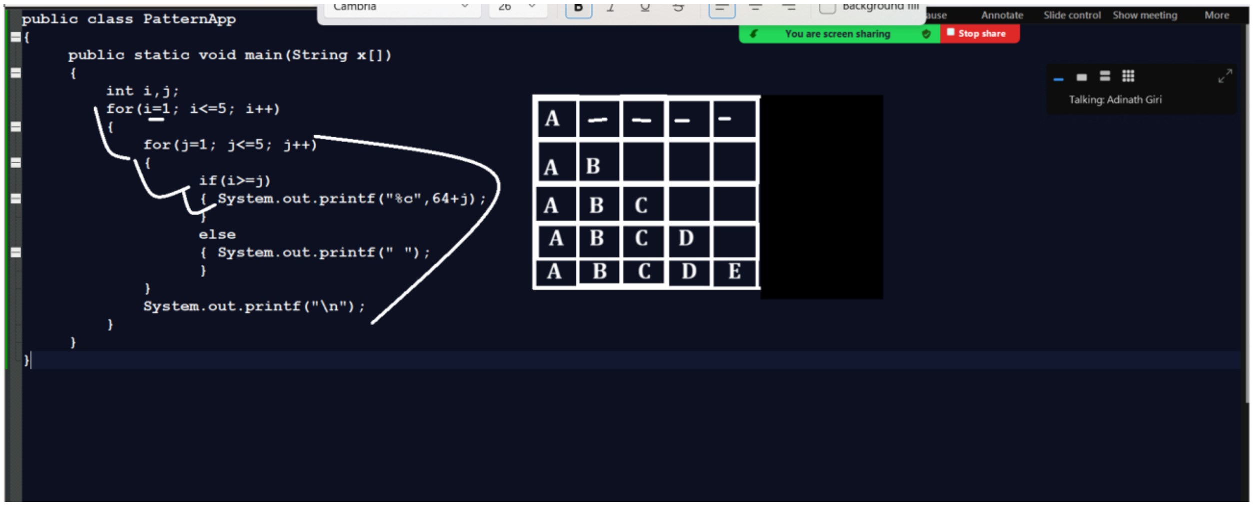


```
public class PatternApp
{
    public static void main(String x[])
    {
        int i,j;
        for(i=1; i<=5; i++)
        {
            for(j=1; j<=5; j++)
            {
                if(i>=j)
                { System.out.printf("*");
                }
                else
                { System.out.printf(" ");
                }
            }
            System.out.printf("\n");
        }
    }
}
```

Output

```
C:\Program Files\Java\jdk-23\bin>javac PatternApp.java

C:\Program Files\Java\jdk-23\bin>java PatternApp
*
**
***
****
*****

C:\Program Files\Java\jdk-23\bin>
```

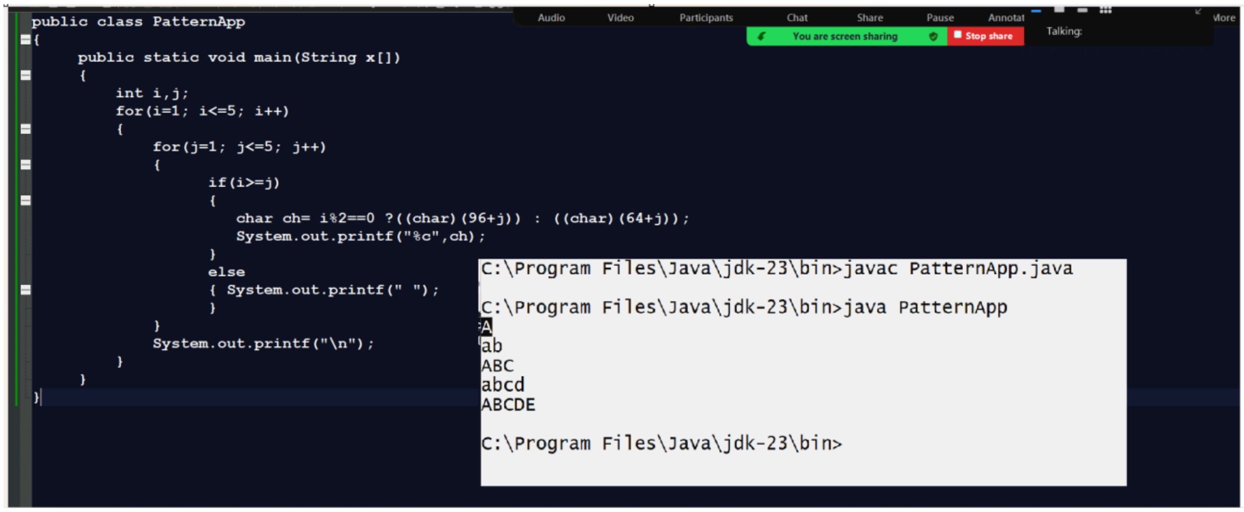
Example: WAP to print the following pattern?



```
public class PatternApp
{
      public static void main(String x[])
      {
          int i,j;
          for(i=1; i<=5; i++)
          {
              for(j=1; j<=5; j++)
              {
                    if(i>=j)
                    { System.out.printf("%c",64+j);
                    }
                    else
                    { System.out.printf(" ");
                    }
              }
              System.out.printf("\n");
          }
      }
}
```

A	–	–	–	–
A	B			
A	B	C		
A	B	C	D	
A	B	C	D	E

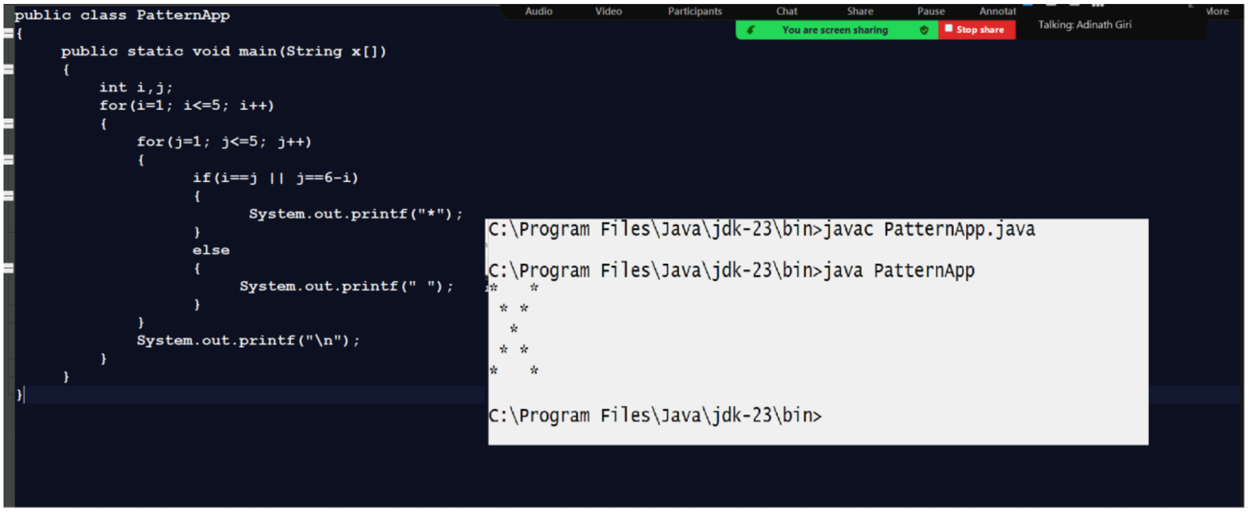
Example: WAP to print the following pattern?



```
public class PatternApp
{
      public static void main(String x[])
      {
          int i,j;
          for(i=1; i<=5; i++)
          {
              for(j=1; j<=5; j++)
              {
                    if(i>=j)
                    {
                       char ch= i%2==0 ?((char)(96+j)) : ((char)(64+j));
                       System.out.printf("%c",ch);
                    }
                    else
                    { System.out.printf(" ");
                    }
              }
              System.out.printf("\n");
          }
      }
}
```

```
C:\Program Files\Java\jdk-23\bin>javac PatternApp.java

C:\Program Files\Java\jdk-23\bin>java PatternApp
A
ab
ABC
abcd
ABCDE

C:\Program Files\Java\jdk-23\bin>
```

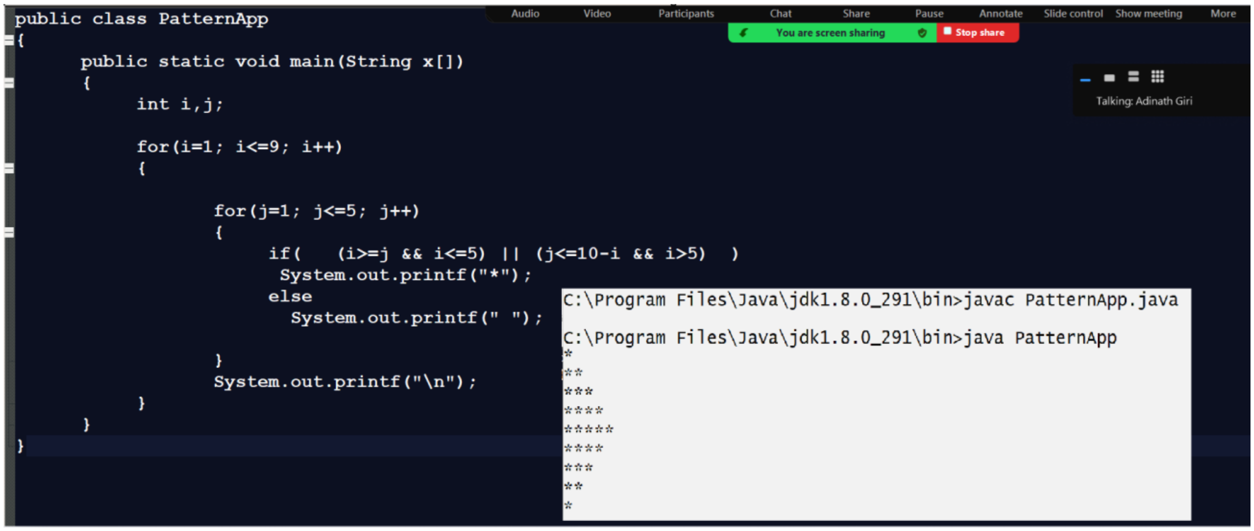
Q. WAP to print the following pattern?



```
public class PatternApp
{
      public static void main(String x[])
      {
          int i,j;
          for(i=1; i<=5; i++)
          {
              for(j=1; j<=5; j++)
              {
                    if(i==j || j==6-i)
                    {
                          System.out.printf("*");
                    }
                    else
                    {
                         System.out.printf(" ");
                    }
              }
              System.out.printf("\n");
          }
      }
}
```

```
C:\Program Files\Java\jdk-23\bin>javac PatternApp.java

C:\Program Files\Java\jdk-23\bin>java PatternApp
*   *
 * *
  *
 * *
*   *

C:\Program Files\Java\jdk-23\bin>
```

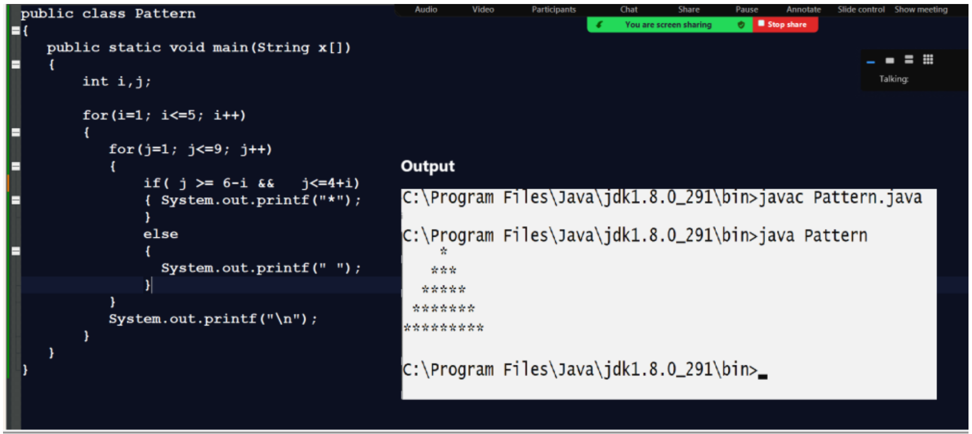
Example: WAP to print the following patterns?



```
public class PatternApp
{
	public static void main(String x[])
	{
		int i,j;

		for(i=1; i<=9; i++)
		{

			for(j=1; j<=5; j++)
			{
			   if(   (i>=j && i<=5) || (j<=10-i && i>5)   )
			    System.out.printf("*");
			   else
			     System.out.printf(" ");

			}
			System.out.printf("\n");
		}
	}
}
```

```
C:\Program Files\Java\jdk1.8.0_291\bin>javac PatternApp.java

C:\Program Files\Java\jdk1.8.0_291\bin>java PatternApp
*
**
***
****
*****
****
***
**
*
```

Example: WAP to print the following pattern?

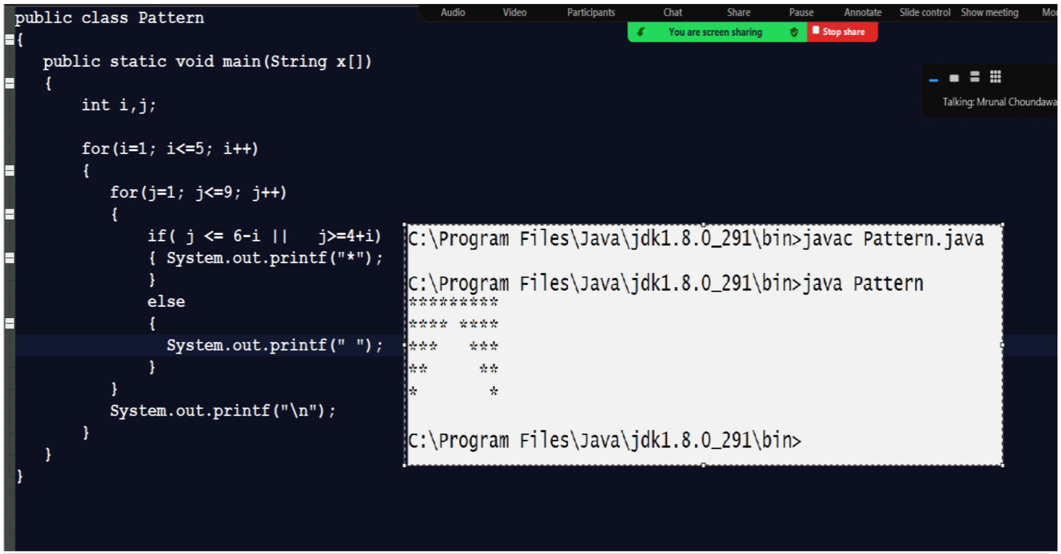


```
public class Pattern
{
   public static void main(String x[])
   {
      int i,j;

      for(i=1; i<=5; i++)
      {
         for(j=1; j<=9; j++)
         {
            if( j >= 6-i &&   j<=4+i)
            { System.out.printf("*");
            }
            else
            {
               System.out.printf(" ");
            }
         }
         System.out.printf("\n");
      }
   }
}
```

Output

```
C:\Program Files\Java\jdk1.8.0_291\bin>javac Pattern.java

C:\Program Files\Java\jdk1.8.0_291\bin>java Pattern
    *
   ***
  *****
 *******
*********

C:\Program Files\Java\jdk1.8.0_291\bin>_
```

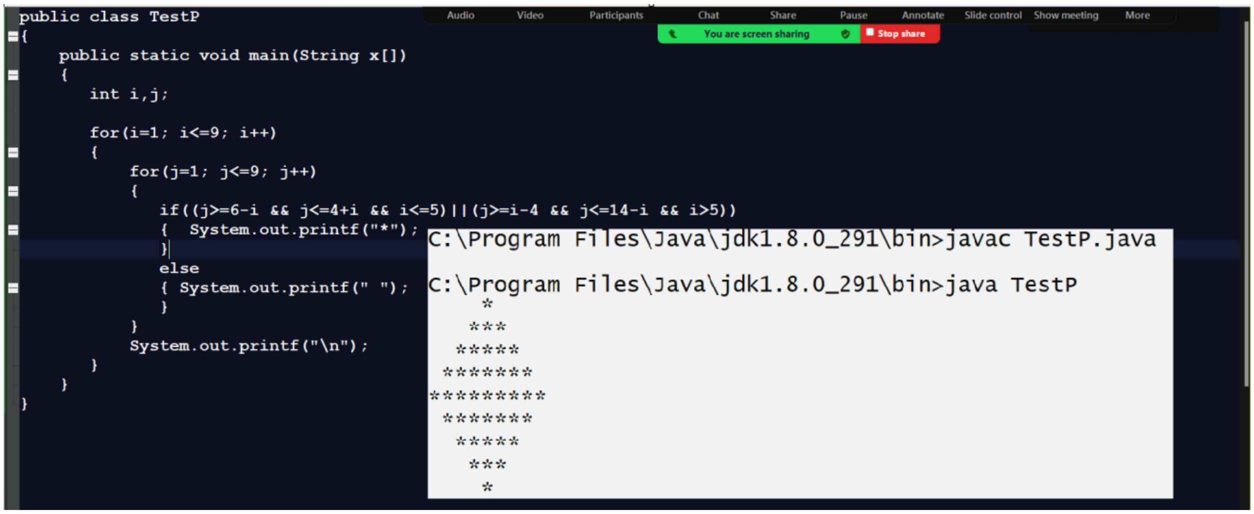
Example: WAP to print te following pattern?



```
public class Pattern
{
   public static void main(String x[])
   {
      int i,j;

      for(i=1; i<=5; i++)
      {
         for(j=1; j<=9; j++)
         {
            if( j <= 6-i ||   j>=4+i)
            { System.out.printf("*");
            }
            else
            {
              System.out.printf(" ");
            }
         }
         System.out.printf("\n");
      }
   }
}
```

```
C:\Program Files\Java\jdk1.8.0_291\bin>javac Pattern.java

C:\Program Files\Java\jdk1.8.0_291\bin>java Pattern
*********
**** ****
***   ***
**     **
*       *

C:\Program Files\Java\jdk1.8.0_291\bin>
```

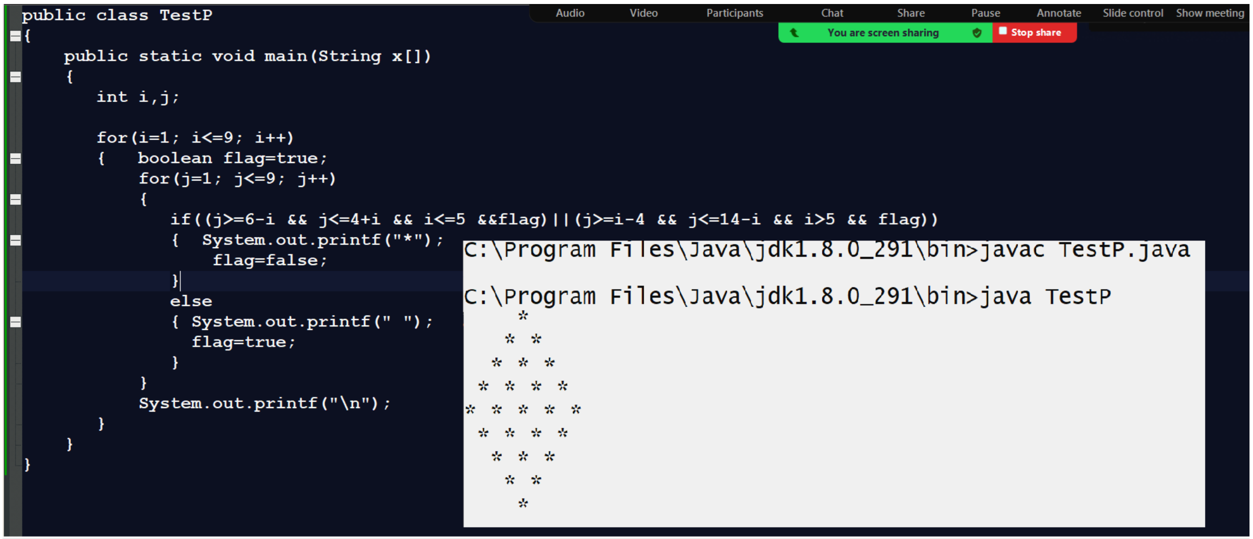
Example: WAP to print the following pattern?



```
public class TestP
{
    public static void main(String x[])
    {
       int i,j;

       for(i=1; i<=9; i++)
       {
           for(j=1; j<=9; j++)
           {
              if((j>=6-i && j<=4+i && i<=5)||(j>=i-4 && j<=14-i && i>5))
              {  System.out.printf("*");
              }
              else
              { System.out.printf(" ");
              }
           }
           System.out.printf("\n");
       }
    }
}
```

```
C:\Program Files\Java\jdk1.8.0_291\bin>javac TestP.java

C:\Program Files\Java\jdk1.8.0_291\bin>java TestP
    *
   ***
  *****
 *******
*********
 *******
  *****
   ***
    *
```

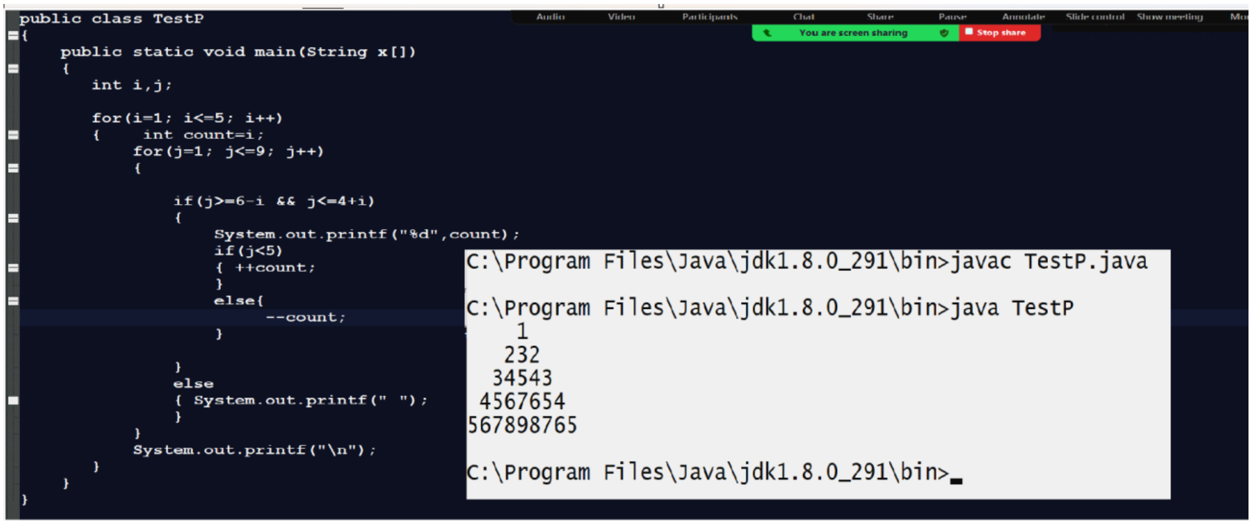
Example:WAP to print the following pattern?



```
public class TestP
{
    public static void main(String x[])
    {
       int i,j;

       for(i=1; i<=9; i++)
       {   boolean flag=true;
           for(j=1; j<=9; j++)
           {
              if((j>=6-i && j<=4+i && i<=5 &&flag)||(j>=i-4 && j<=14-i && i>5 && flag))
              {  System.out.printf("*");
                 flag=false;
              }
              else
              { System.out.printf(" ");
                flag=true;
              }
           }
           System.out.printf("\n");
       }
    }
}
```

```
C:\Program Files\Java\jdk1.8.0_291\bin>javac TestP.java

C:\Program Files\Java\jdk1.8.0_291\bin>java TestP
    *
   * *
  * * *
 * * * *
* * * * *
 * * * *
  * * *
   * *
    *
```

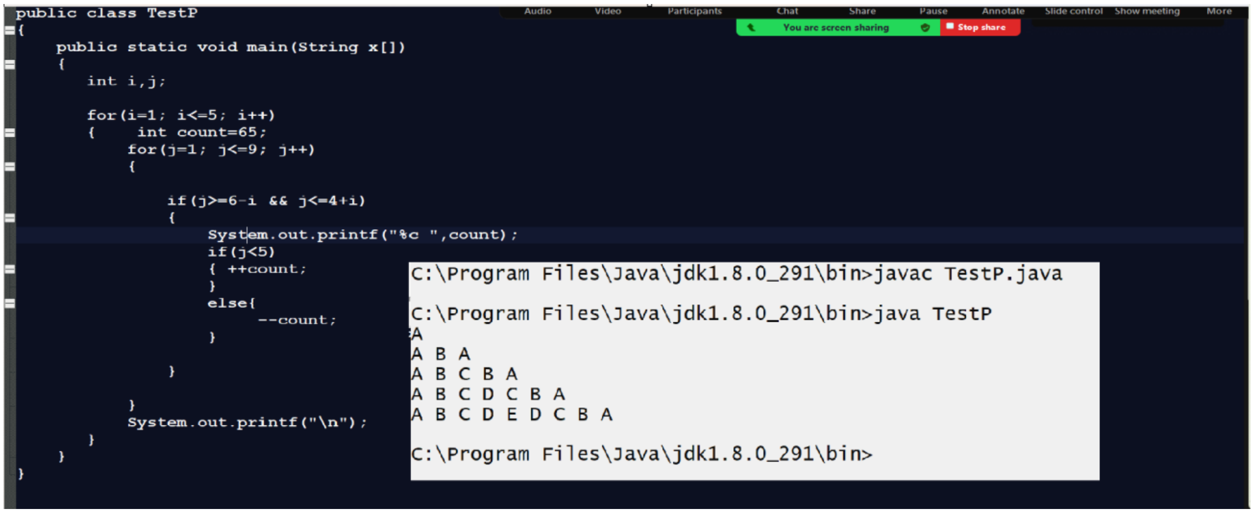
Example with source code



```
public class TestP
{
    public static void main(String x[])
    {
       int i,j;

       for(i=1; i<=5; i++)
       {    int count=i;
           for(j=1; j<=9; j++)
           {

              if(j>=6-i && j<=4+i)
              {
                 System.out.printf("%d",count);
                 if(j<5)
                 { ++count;
                 }
                 else{
                        --count;
                 }

              }
              else
              { System.out.printf(" ");
              }
           }
           System.out.printf("\n");
       }
    }
}
```

```
C:\Program Files\Java\jdk1.8.0_291\bin>javac TestP.java

C:\Program Files\Java\jdk1.8.0_291\bin>java TestP
    1
   232
  34543
 4567654
567898765

C:\Program Files\Java\jdk1.8.0_291\bin>_
```

Example: WAP to print the following pattern?



```
public class TestP
{
    public static void main(String x[])
    {
       int i,j;

       for(i=1; i<=5; i++)
       {    int count=65;
           for(j=1; j<=9; j++)
           {

               if(j>=6-i && j<=4+i)
               {
                   System.out.printf("%c ",count);
                   if(j<5)
                   { ++count;
                   }
                   else{
                        --count;
                   }

               }

           }
           System.out.printf("\n");
       }
    }
}
```

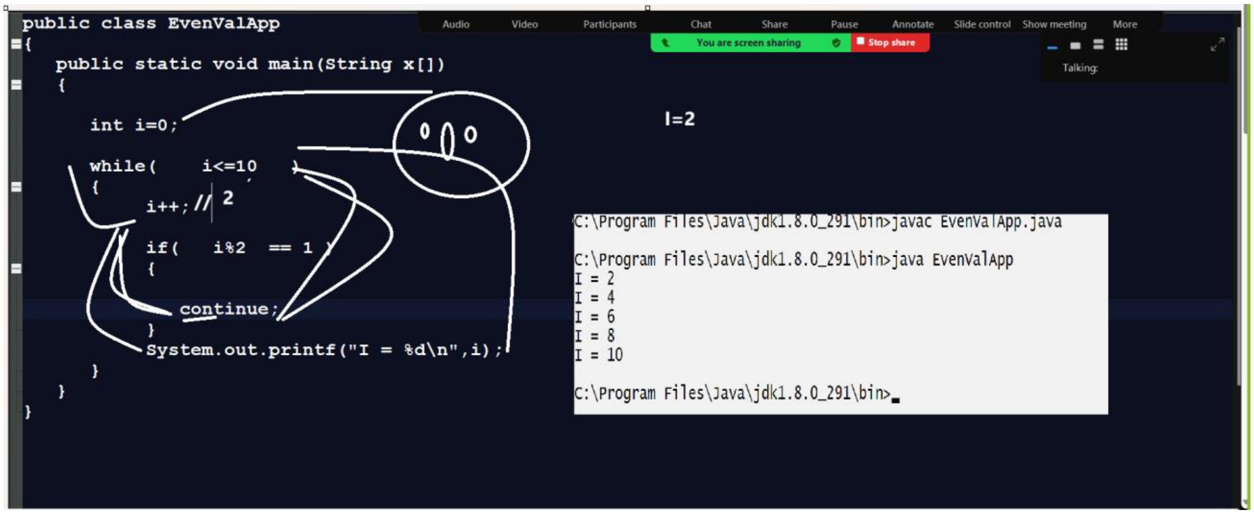
```
C:\Program Files\Java\jdk1.8.0_291\bin>javac TestP.java

C:\Program Files\Java\jdk1.8.0_291\bin>java TestP
A
A B A
A B C B A
A B C D C B A
A B C D E D C B A

C:\Program Files\Java\jdk1.8.0_291\bin>
```

continue keyword

____

Continue keyword is used for skip the some execution steps of loop called as continue keyword. When we found continue keyword in loop then after continue keyword loop not executed directly jump Condition in loop.

Example: we want to print the even values between 1 to 10.



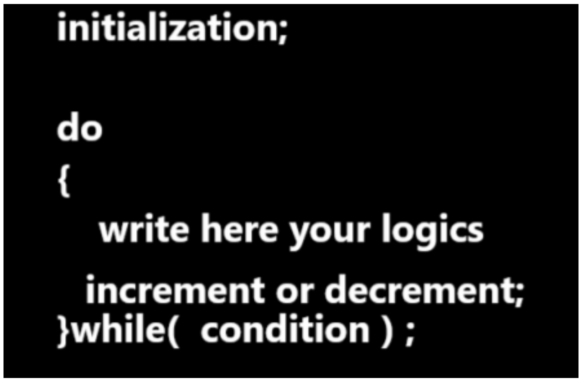
```
public class EvenValApp
{
    public static void main(String x[])
    {

       int i=0;

       while(    i<=10    )
       {
           i++; // 2

           if(   i%2   == 1 )
           {

               continue;
           }
           System.out.printf("I = %d\n",i);
       }
    }
}
```

```
C:\Program Files\Java\jdk1.8.0_291\bin>javac EvenValApp.java

C:\Program Files\Java\jdk1.8.0_291\bin>java EvenValApp
I = 2
I = 4
I = 6
I = 8
I = 10

C:\Program Files\Java\jdk1.8.0_291\bin>
```

Do while loop

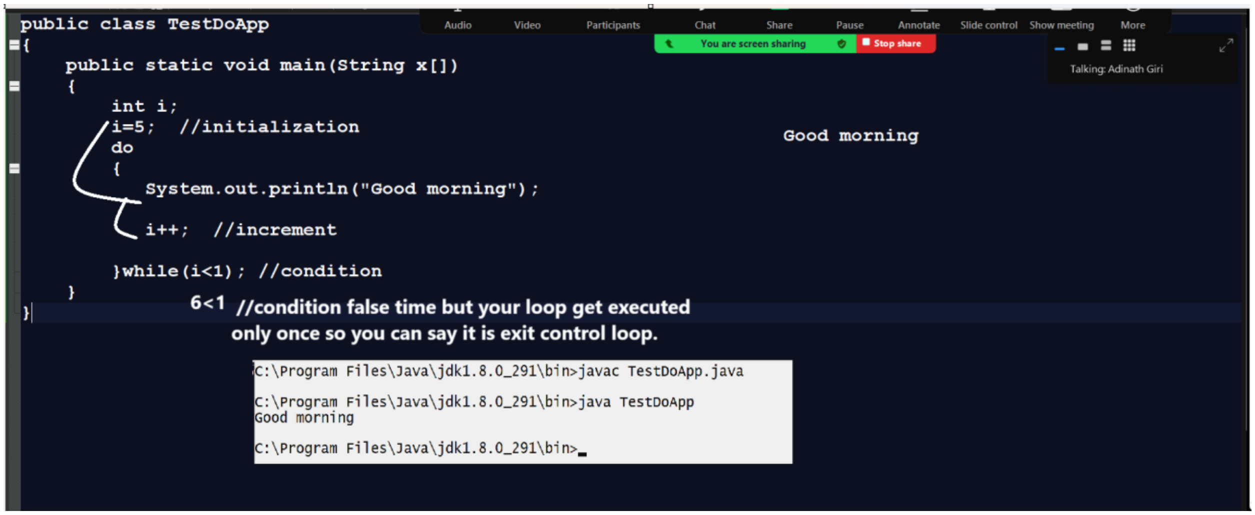
do while loop is exit control loop means if your condition false first time your loop will be execute minimum single time and if your condition is true then your loop execute like as normal loop.

Syntax:



```
initialization;

do
{
    write here your logics

   increment or decrement;
}while(  condition ) ;
```

Example:



```
public class TestDoApp
{
    public static void main(String x[])
    {
        int i;
        i=5;  //initialization
        do
        {
            System.out.println("Good morning");

            i++;  //increment

        }while(i<1); //condition
    }
}
```

Good morning

6<1 //condition false time but your loop get executed only once so you can say it is exit control loop.

```
C:\Program Files\Java\jdk1.8.0_291\bin>javac TestDoApp.java

C:\Program Files\Java\jdk1.8.0_291\bin>java TestDoApp
Good morning

C:\Program Files\Java\jdk1.8.0_291\bin>_
```

Why use do while loop?

Do while loop recommend to work with ODD Loop

Q. What is Odd loop?

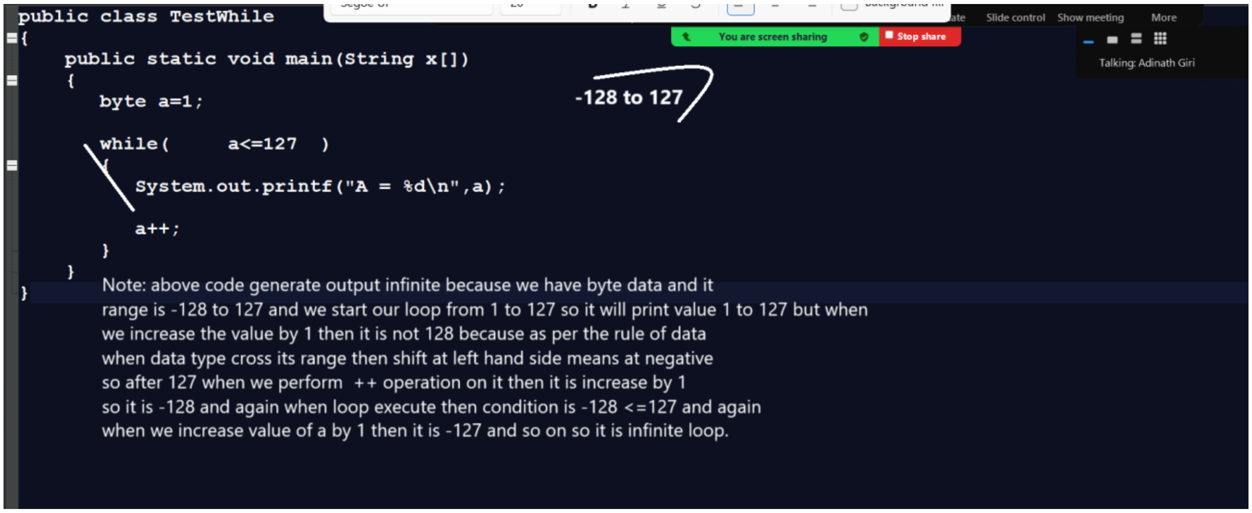
ODD Loop means loop termination is dependent on user choice not on particular condition called as Odd loop.

Example: WAP to input the qty and rate keyword from keyboard and calculate its total bill and system should ask to use do you want to continue and if we press Y or y then loop should execute if we press any other letter then loop should not execute.

Example with source code

```
import java.util.*;
public class ODDLoop
{
   public static void main(String x[])
          {
            int qty,rate,total;
            char ch;
            do{
             Scanner xyz  = new Scanner(System.in);
                    System.out.println("Enter quantity and rate from keyboard");
                    qty=xyz.nextInt();
                    rate=xyz.nextInt();
                    total=qty*rate;
                    System.out.println("total bill is  "+total);
                    xyz.nextLine();
                    System.out.println("Do you want to continue");
                    ch=xyz.nextLine().charAt(0);

            }while(ch=='y' || ch=='Y');
          }
}
```

Q. What will be output of given code?

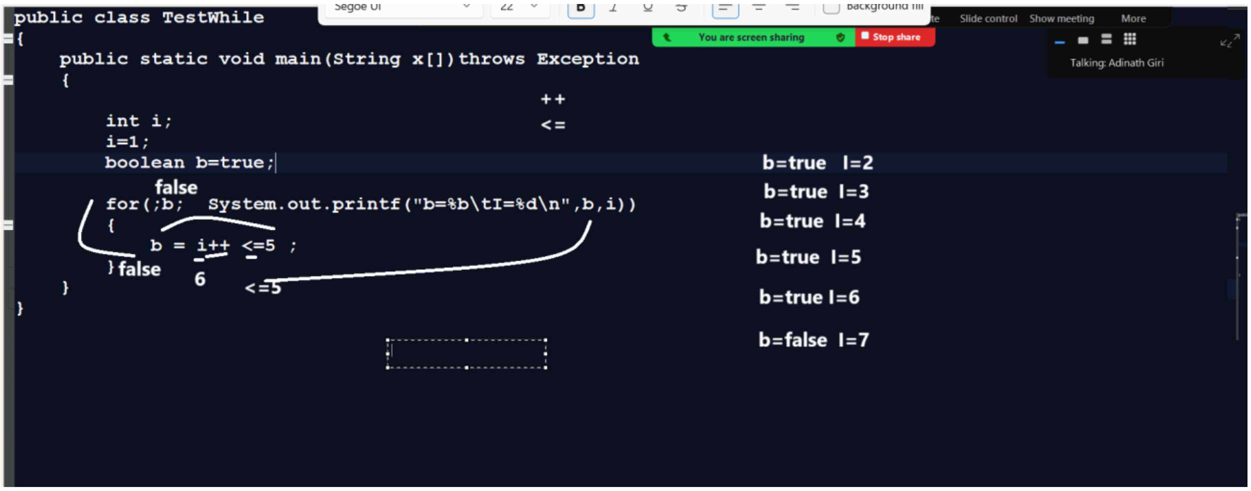


```
public class TestWhile
{
    public static void main(String x[])
    {
       byte a=1;

       while(      a<=127   )
       {
          System.out.printf("A = %d\n",a);

          a++;
       }
    }
}
```

-128 to 127

Note: above code generate output infinite because we have byte data and it range is -128 to 127 and we start our loop from 1 to 127 so it will print value 1 to 127 but when we increase the value by 1 then it is not 128 because as per the rule of data when data type cross its range then shift at left hand side means at negative so after 127 when we perform ++ operation on it then it is increase by 1 so it is -128 and again when loop execute then condition is -128 <=127 and again when we increase value of a by 1 then it is -127 and so on so it is infinite loop.

Q. what will be output of given code?



```
public class TestWhile
{
    public static void main(String x[])throws Exception
    {

        int i;
        i=1;
        boolean b=true;

        for(;b;  System.out.printf("b=%b\tI=%d\n",b,i))
        {
            b = i++ <=5 ;
        }
    }
}
```

++

<=

false

}false

6

<=5

```
b=true  I=2
b=true  I=3
b=true  I=4
b=true  I=5
b=true I=6
b=false  I=7
```