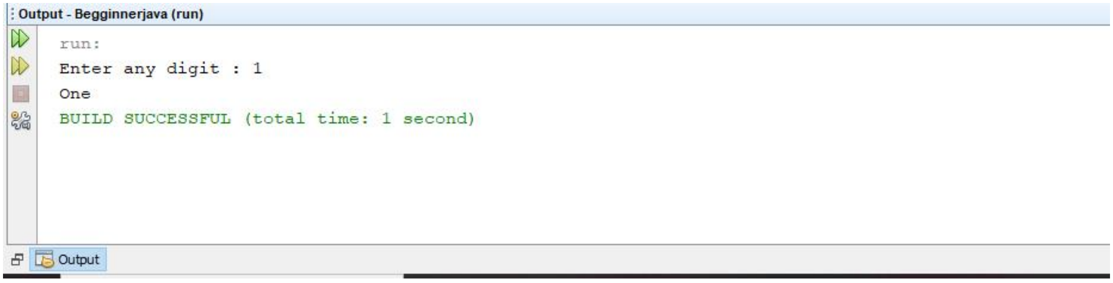
Control Statement Problem Sloving Part 02

1. Write a java to Check the Digit.

Input:

```
package begginnerjava;
import java.util.Scanner;

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

        Scanner input = new Scanner(System.in);

        int digit;
        System.out.print("Enter any digit : ");
        digit = input.nextInt();

        switch(digit){

            case 0:
            System.out.println("Zero");
            break;

            case 1:
            System.out.println("One");
            break;

            case 2:
            System.out.println("Two);
            break;

            case 3:
            System.out.println("Three");
            break;

            case 4:
            System.out.println("Four");
            break;

            case 5:
            System.out.println("Five");
            break;
            default:
                System.out.println("Not a digit");



        }



    }



}
```

Output:



```
run:
Enter any digit : 1
One
BUILD SUCCESSFUL (total time: 1 second)
```

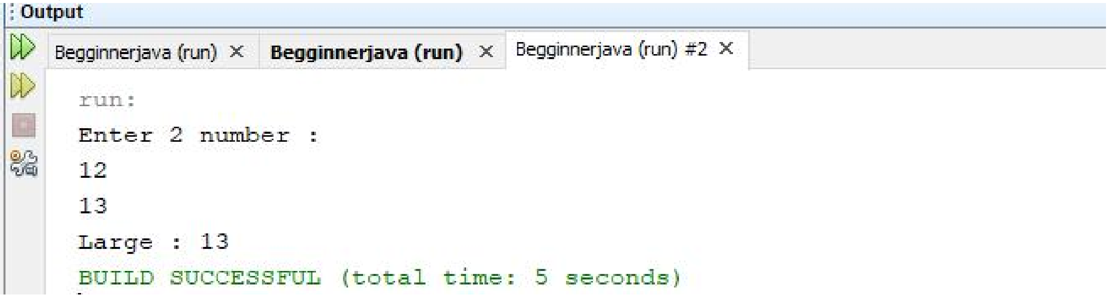
2. Write a Java Programme to find the number large .

Input:

```
package begginnerjava;

import java.util.Scanner;
public class conditionaloperator {

    public static void main(String [] args){

      Scanner input = new Scanner(System.in);
      int num1,num2,large;

     System.out.print("Enter 2 number : ");
```

```
        num1=input.nextInt();
        num2=input.nextInt();

        large = (num1>num2) ? num1 : num2 ;

        System.out.println("Large : "+large);

    }


}
```

Output:



```
run:
Enter 2 number :
12
13
Large : 13
BUILD SUCCESSFUL (total time: 5 seconds)
```

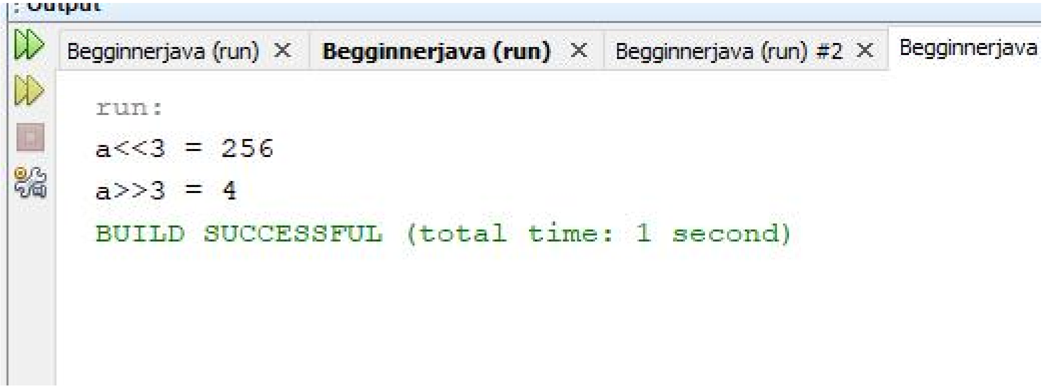
3. Write a java Programme to checch the Bitwise Operator.

Input:

```
package begginnerjava;
import java.util.Scanner;
public class BitwiseOperator {

  public static void main(String [] args){

     int a = 32;
     int b = 23;

     int c;

     c = a<<3;
     System.out.println("a<<3 = " +c);

     c = a>>3;
```

```
        System.out.println("a>>3 = " +c);



    }

}
```

Output:



```
run:
a<<3 = 256
a>>3 = 4
BUILD SUCCESSFUL (total time: 1 second)
```

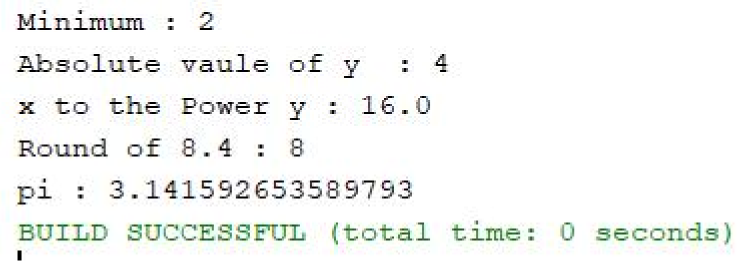
4. Write a Java Programme to find math class operator.

Input:

```
package begginnerjava;

public class math {


    public static void main(String[] args){


     int x = 2;
     int y = 4;


     int max = (Math.max(x,y));
     System.out.println("Maximum : "+max);


         int min = (Math.min(x,y));
         System.out.println("Minimum : "+min);
```

```
        int absolute = Math.abs(y);
        System.out.println("Absolute vaule of y  : "+absolute);


          double power = Math.pow(x,y);
         System.out.println("x to the Power y : "+power);

        int round = Math.round(8.4f);
       System.out.println("Round of 8.4 : "+round);

         double pi = Math.PI;
          System.out.println("pi : "+pi);




    }

}
```

Output:



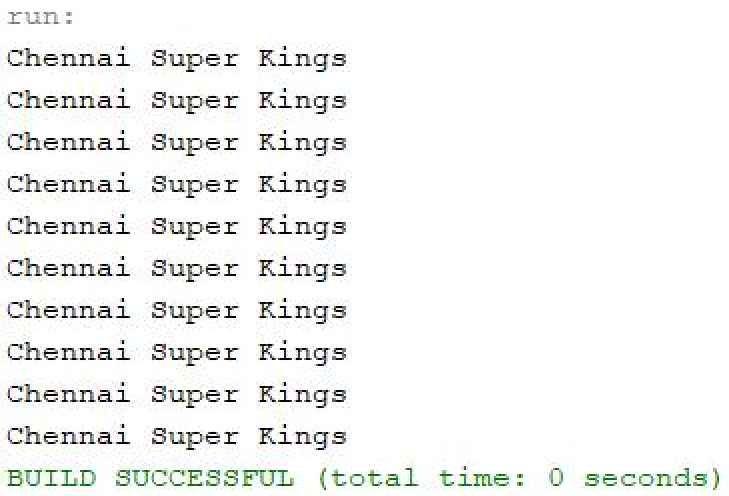
```
Minimum : 2
Absolute vaule of y  : 4
x to the Power y : 16.0
Round of 8.4 : 8
pi : 3.141592653589793
BUILD SUCCESSFUL (total time: 0 seconds)
```

5. Write a java programee to find the name length using for loop.

Input:

```
package begginnerjava;


public class forloop {

    public static void main(String [] args){
        for(int i=1;i<=10;i++){
            System.out.println("Chennai Super Kings ");

        }


    }
}
```

Output:



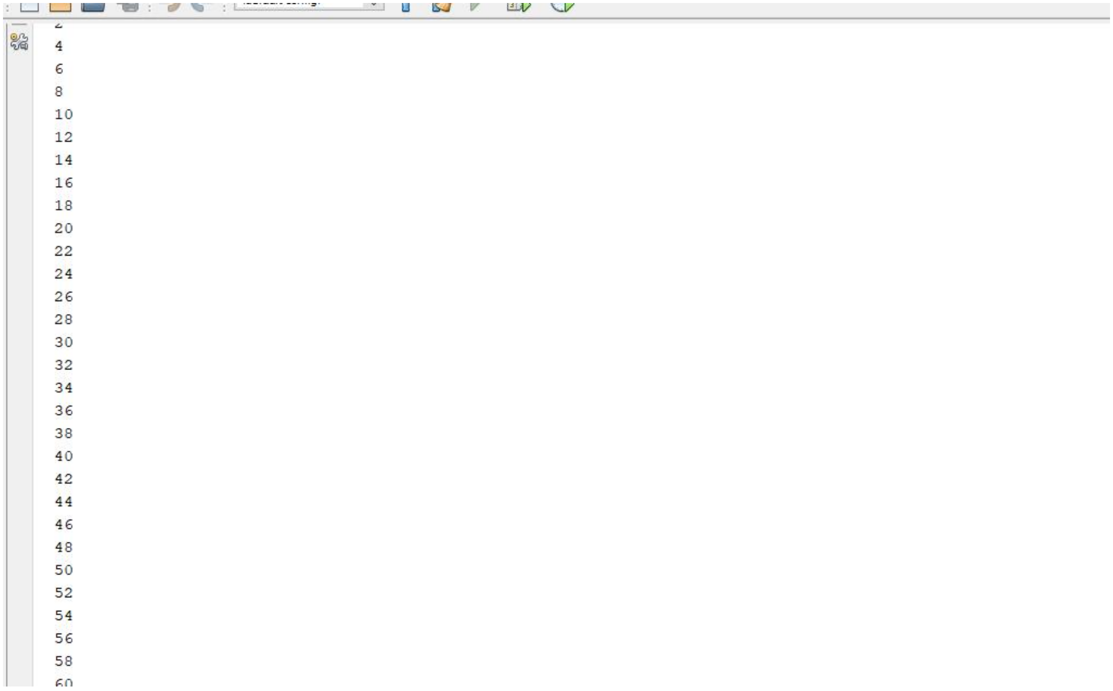
```
run:
Chennai Super Kings
Chennai Super Kings
Chennai Super Kings
Chennai Super Kings
Chennai Super Kings
Chennai Super Kings
Chennai Super Kings
Chennai Super Kings
Chennai Super Kings
Chennai Super Kings
BUILD SUCCESSFUL (total time: 0 seconds)
```

6. Write a Java Programme to find 1 to 100 number using for loop.

Input:

```
package begginnerjava;


public class forloop {

    public static void main(String [] args){

        for(int i=2;i<=100; I=I+2){
            System.out.println(i);

        }


    }
}
```

Output:



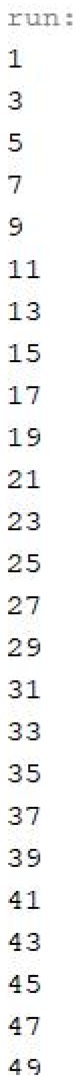
```
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46
48
50
52
54
56
58
60
```

7. Write a Java Programme to find 1 to 100 number using while loop.

Input:

```
package begginnerjava;

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

       int i=1;
       while (i<=100)  {

       System.out.println(i);
       i=i+2;
       }




   }

}
```

Output:



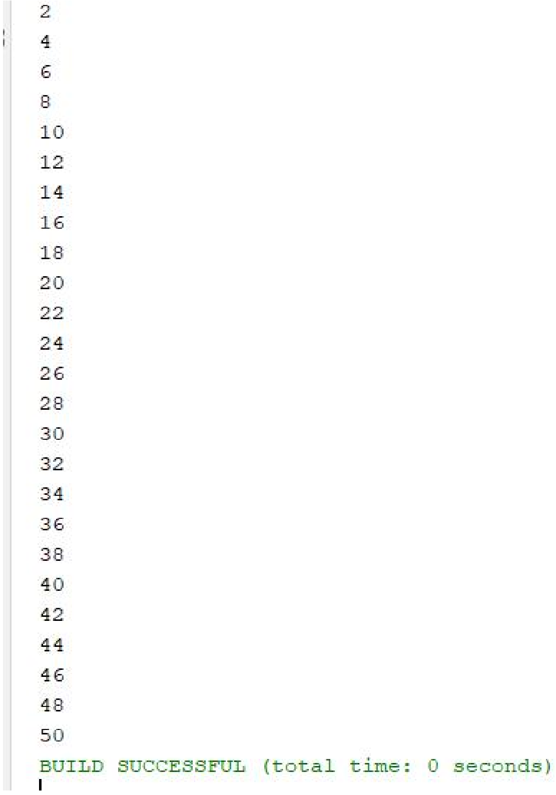
```
run:
1
3
5
7
9
11
13
15
17
19
21
23
25
27
29
31
33
35
37
39
41
43
45
47
49
```

8. Write a Java Programme to find 1 to 100 number using while loop.

Input:

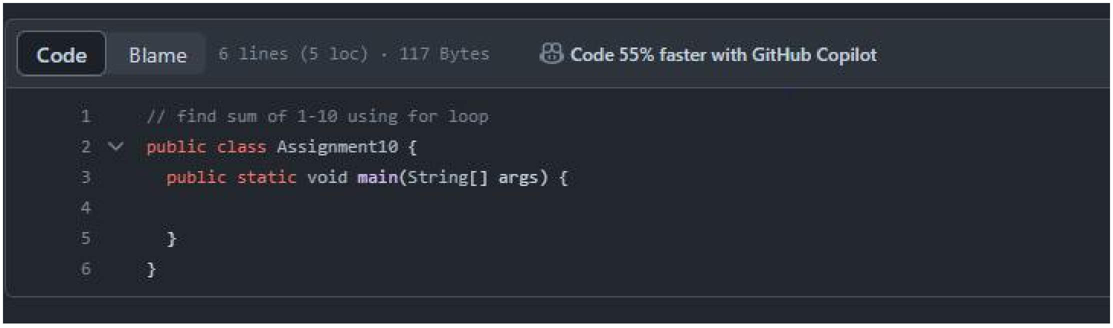
```
package begginnerjava;


public class dowhileloop {

   public static void main(String [] args){

      int i =2;
      do {
            System.out.println(i);

            i=i+2

        }
        while(i<=50);
    }

}
```

Output:



```
2
4
6
8
10
12
14
16
18
20
22
24
26
28
30
32
34
36
38
40
42
44
46
48
50
BUILD SUCCESSFUL (total time: 0 seconds)
```

9.Write a Java Programme to find sum (1-10)



Code
Blame
6 lines (5 loc) · 117 Bytes
Code 55% faster with GitHub Copilot

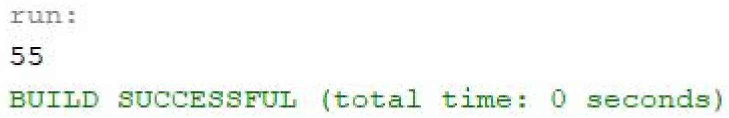
```
// find sum of 1-10 using for loop
public class Assignment10 {
  public static void main(String[] args) {

  }
}
```

Input:

package practisejava;

```
public class Assignment10 {


public static void main(String [] args){

        int i,sum=0;


        for(i=1;i<=10;i++)
        {
            sum+=i;
        }
        System.out.println(sum);
    }

}

}
```

Output:

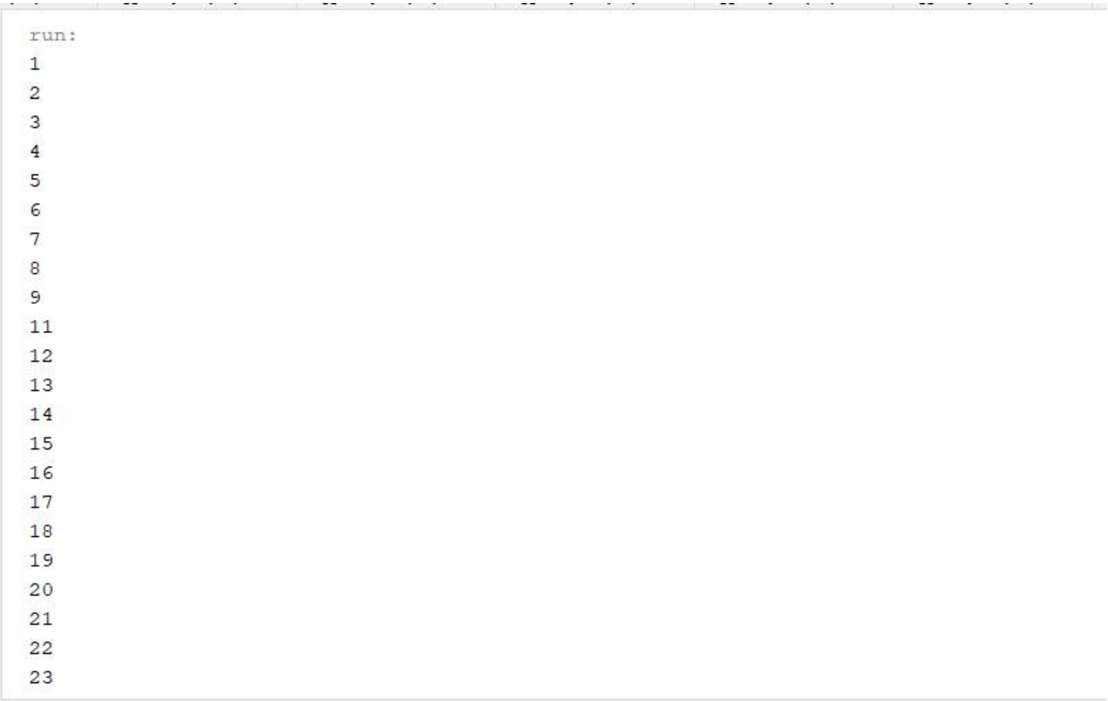


```
run:
55
BUILD SUCCESSFUL (total time: 0 seconds)
```

10.Write a Java Programme to check break statement.
Input:

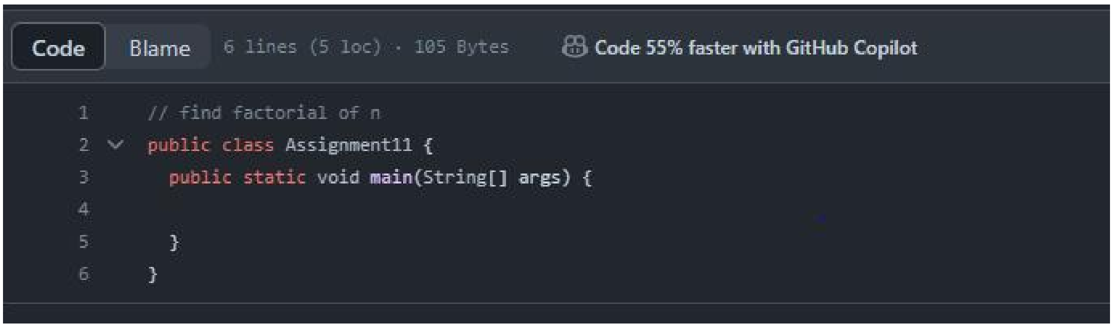
```
package begginnerjava;


public class BreakStatement {

    public static void main(String [] args){

        for( int i=1;i<=100;i++){
            if(i==10){
                continue;

            }

            System.out.println(i);
        }


    }

}
```

Output:



```
run:
1
2
3
4
5
6
7
8
9
11
12
13
14
15
16
17
18
19
20
21
22
23
```

10.Write a Java To find n factorial Number.



Code
Blame
6 lines (5 loc) · 105 Bytes
Code 55% faster with GitHub Copilot

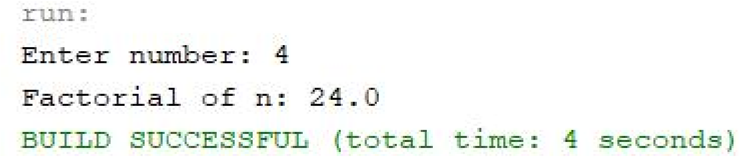
```
// find factorial of n
public class Assignment11 {
  public static void main(String[] args) {

  }
}
```

Input:

package practisejava;

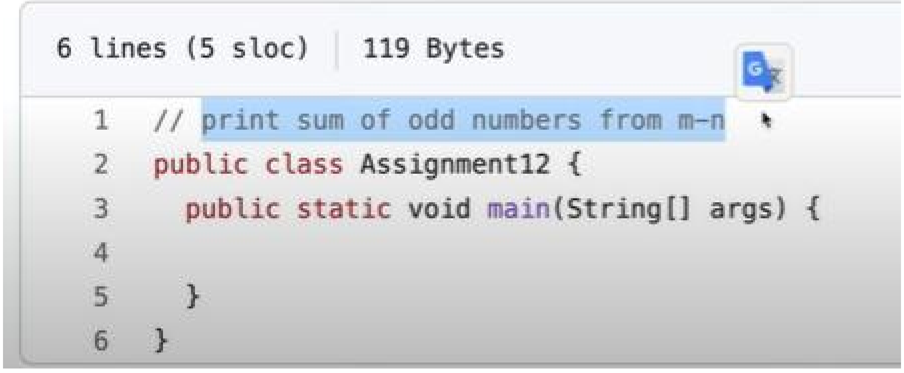
```
import java.util.Scanner;
public class Assignment11 {



   public static void main(String[] args) {

      Scanner sc = new Scanner(System.in);

      System.out.print("Enter number: ");
      int n = sc.nextInt();
      double fact=1;

      for(int i=1;i<=n;i++)
      {
         fact*=i;
      }
      System.out.println("Factorial of n: "+fact);
   }

}

}
```

Output:



```
run:
Enter number: 4
Factorial of n: 24.0
BUILD SUCCESSFUL (total time: 4 seconds)
```

11. Write a java programme to find odd numbers m to n.



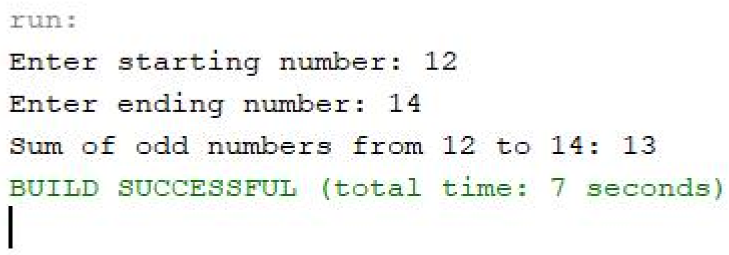
6 lines (5 sloc) | 119 Bytes

```
// print sum of odd numbers from m-n
public class Assignment12 {
  public static void main(String[] args) {

  }
}
```

Input:

```
package practisejava;
import java.util.Scanner;
public class Assignment12 {

    public static void main(String[] args) {

        Scanner sc = new Scanner(System.in);

        System.out.print("Enter starting number: ");
        int m = sc.nextInt();

        System.out.print("Enter ending number: ");
        int n = sc.nextInt();

        int sum = 0;

        for(int i=m;i<=n;i++)
        {
            if(i%2!=0)
            {
                sum+=i;
            }
        }
        System.out.println("Sum of odd numbers from "+m+" to "+n+":
"+sum);
    }

}

}
```

Output:

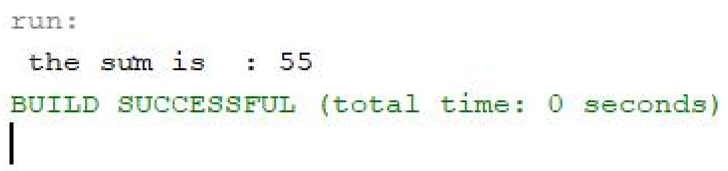


```
run:
Enter starting number: 12
Enter ending number: 14
Sum of odd numbers from 12 to 14: 13
BUILD SUCCESSFUL (total time: 7 seconds)
```

12. Write a java programme to find even numbers m to n.
Input:

```
package begginnerjava;


public class forloop1 {

    public static void main(String [] args){

        int sum=0;
        for( int i=1;i<=10; i++){

            sum = sum + i;

        }

        System.out.println(" the sum is  : "+sum);


}
}
```

Output:

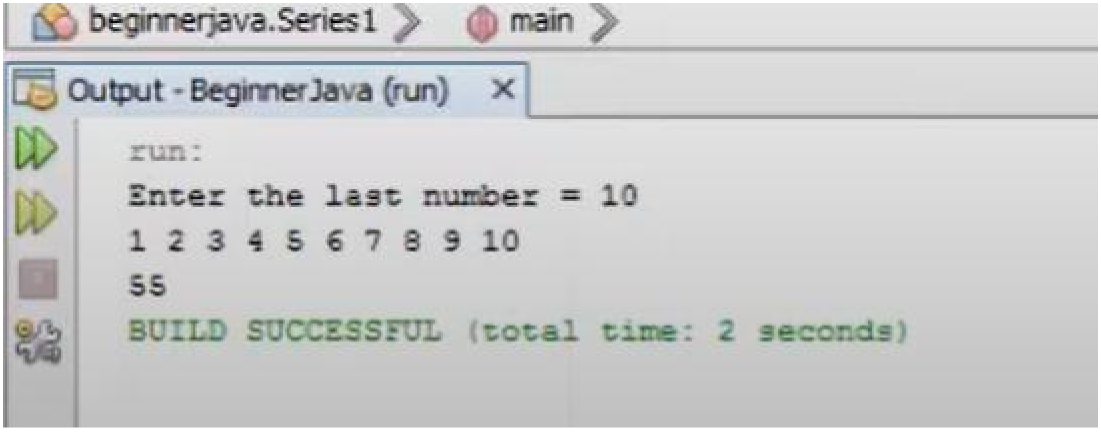


```
run:
 the sum is  : 55
BUILD SUCCESSFUL (total time: 0 seconds)
```

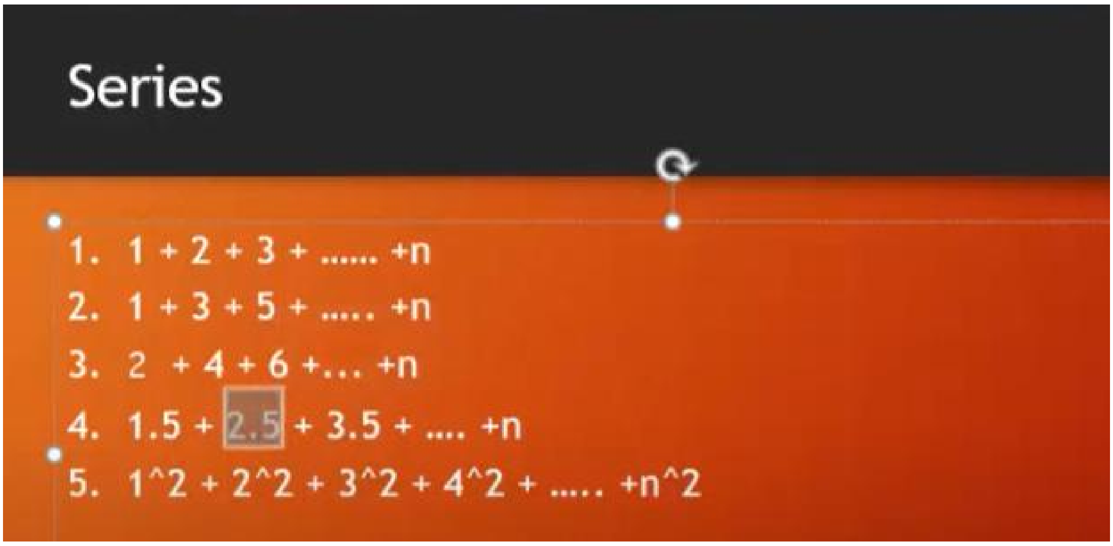
13. Write a java programme to find sum numbers m to n.
1+2+3+……+n
Input:

```
package begginnerjava;

import java.util.Scanner;
public class Series1 {
   public static void main(String[] args){
      Scanner input = new Scanner(System.in);

      int n,sum=0;
      System.out.print("Enter the last number : ");
      n = input.nextInt();

      for (int i=1;i<=n;i=i+1){
         System.out.print(i+"");
         sum =  sum + i;


      }



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





   }
}
```

Output:



```
run:
Enter the last number = 10
1 2 3 4 5 6 7 8 9 10
55
BUILD SUCCESSFUL (total time: 2 seconds)
```



Series

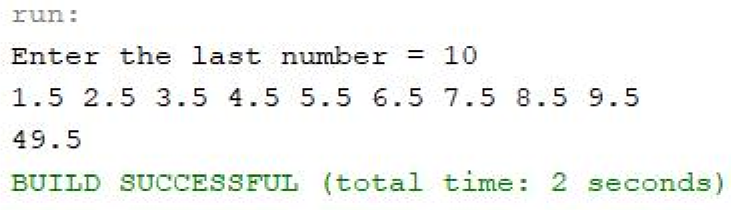
1. 1 + 2 + 3 + …… +n
2. 1 + 3 + 5 + ….. +n
3. 2 + 4 + 6 +… +n
4. 1.5 + 2.5 + 3.5 + …. +n
5. 1^2 + 2^2 + 3^2 + 4^2 + ….. +n^2

14. Write a Java Programme to Find the Series .
1.5+2.5+3.5+………….+n
Input:

```
package begginnerjava;

import java.util.Scanner;
public class Series1 {
    public static void main(String[] args){
        Scanner input = new Scanner(System.in);

       double n ,sum=0;
        System.out.print("Enter the last number = ");
```

```
      n = input.nextDouble();

      for (double i = 1.5; i <=n; i = i+1){
         System.out.print(i+" ");
         sum =  sum + i;
      }


      System.out.println();
      System.out.println(sum);




   }

}
```

Output:



```
run:
Enter the last number = 10
1.5 2.5 3.5 4.5 5.5 6.5 7.5 8.5 9.5
49.5
BUILD SUCCESSFUL (total time: 2 seconds)
```

15. Write a Java Programme to Find the Series .

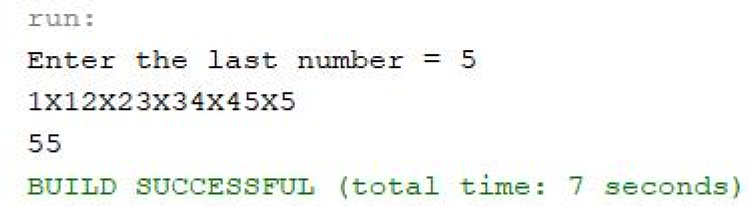
1^2+2^2+3^2+…….n

Input:

```
package begginnerjava;

import java.util.Scanner;
public class Series1 {
   public static void main(String[] args){
      Scanner input = new Scanner(System.in);
```

```
        int n ,sum=0;
        System.out.print("Enter the last number = ");
        n = input.nextInt();

        for (int i = 1; i <=n ; i = i+1){
            System.out.print(i+"X"+i);
            sum =  sum + i*i;
        }


        System.out.println();
        System.out.println(sum);




    }

}
```

Output:



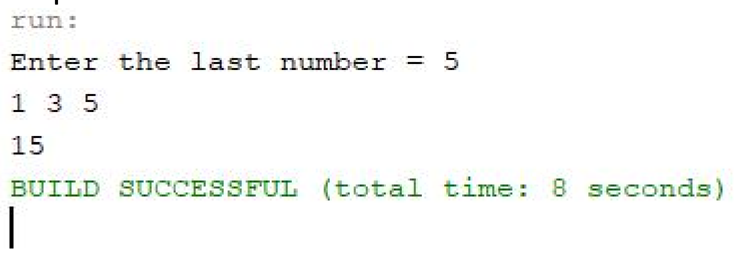
```
run:
Enter the last number = 5
1X12X23X34X45X5
55
BUILD SUCCESSFUL (total time: 7 seconds)
```

16. Write a Java Programme to Find the Series .
1 X 2 X 3 X ………..N
Input:

```
package begginnerjava;

import java.util.Scanner;
public class Series02 {
    public static void main(String [] args){
    Scanner input = new Scanner(System.in);
        int n,result=1;
    System.out.print("Enter the last number = ");
```

```
   n = input.nextInt();
   for (int i=1 i <=n; i= i+2){

       System.out.print(i+" ");
       result = result * i;

   }



   System.out.println();
   System.out.println(result);



}
```

Output:



```
run:
Enter the last number = 5
1 3 5
15
BUILD SUCCESSFUL (total time: 8 seconds)
```

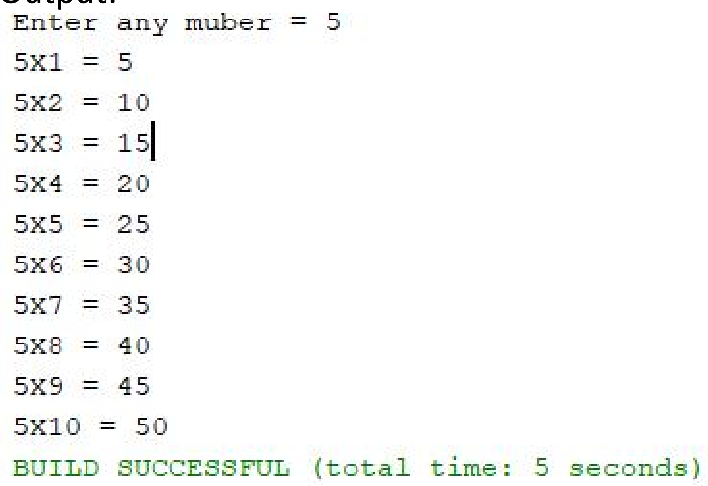
17. Write a Java Programme to Find Multipication Table.

Input:

```
package begginnerjava;

import java.util.Scanner;
public class MultipicationTable {

  public static void main(String[] args){

      Scanner input = new Scanner(System.in);

      int num;

      System.out.print("Enter any muber = ");
```

```
    num = input.nextInt();


    for (int i =1; i<=10 ; i++){

       System.out.println(num+ "X"+i + " = "+num*i);


    }




  }

}
```

Output:



```
Enter any muber = 5
5X1 = 5
5X2 = 10
5X3 = 15
5X4 = 20
5X5 = 25
5X6 = 30
5X7 = 35
5X8 = 40
5X9 = 45
5X10 = 50
BUILD SUCCESSFUL (total time: 5 seconds)
```

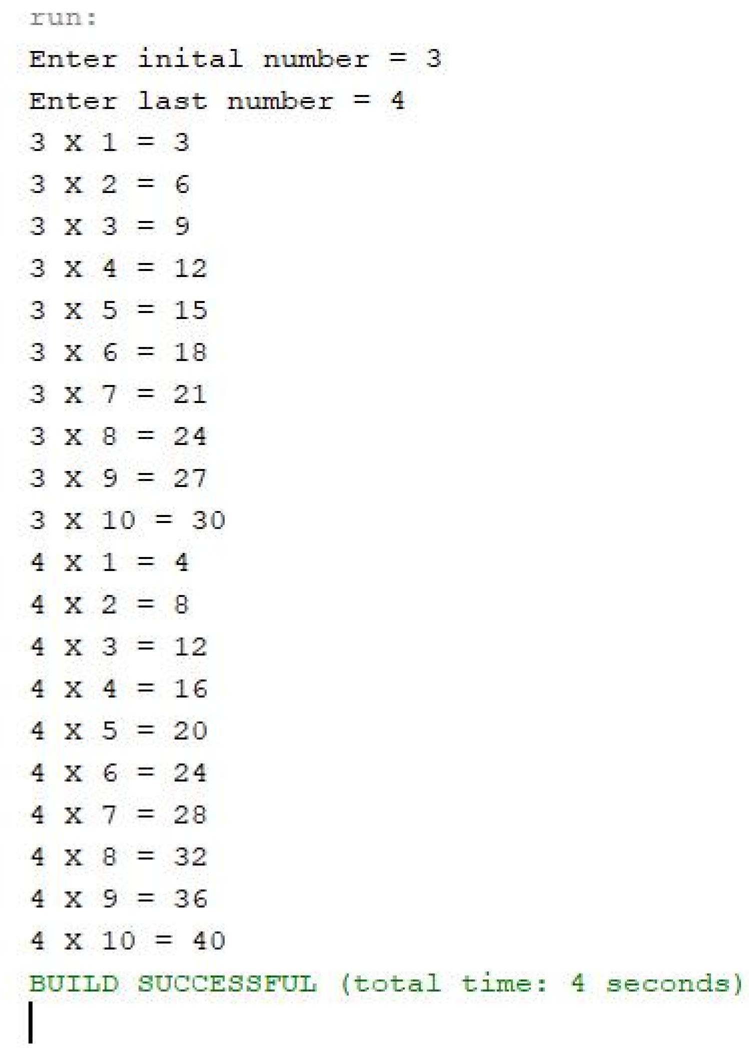
18. Write a Java Programme to Find Multipication Table.

Input:

```
package begginnerjava;

import java.util.Scanner;
```

```
public class MultipicationTable {

  public static void main(String[] args){

     Scanner input = new Scanner(System.in);

     int m,n

     System.out.print("Enter inital number = ");
     m = input.nextInt();

     System.out.print("Enter last number = ");
     n = input.nextInt();


     for (int i = m; i <=n; i++){

              for (int j = 1; j <=10; j++){

      Syst3em.out.println(i+ " X "+j + " = "+i*j);




  }

}
```

Output:



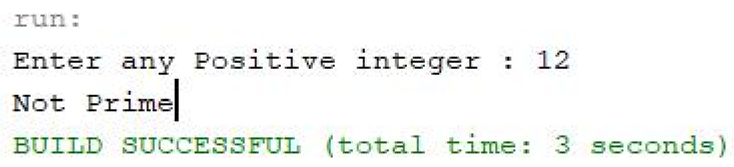
```
run:
Enter inital number = 3
Enter last number = 4
3 X 1 = 3
3 X 2 = 6
3 X 3 = 9
3 X 4 = 12
3 X 5 = 15
3 X 6 = 18
3 X 7 = 21
3 X 8 = 24
3 X 9 = 27
3 X 10 = 30
4 X 1 = 4
4 X 2 = 8
4 X 3 = 12
4 X 4 = 16
4 X 5 = 20
4 X 6 = 24
4 X 7 = 28
4 X 8 = 32
4 X 9 = 36
4 X 10 = 40
BUILD SUCCESSFUL (total time: 4 seconds)
```

19. Write a Java Programme to check prime number or not.

Input:

```
package begginnerjava;

import java.util.Scanner;
public class primenumber {
   public static void main(String [] args){

      Scanner input = new Scanner (System.in);
      System.out.print("Enter any Positive integer : ");
      int num = input.nextInt();
      int count = 0;

      for (int i = 2; i<num; i++){
           if(num%i==0){
               count++;
               break;

           }


           }


         if (count==0){

             System.out.println("Prime Number");

         }

         else {
             System.out.println("Not Prime");

         }


       }






    }




}
```

Output:



```
run:
Enter any Positive integer : 12
Not Prime
BUILD SUCCESSFUL (total time: 3 seconds)
```

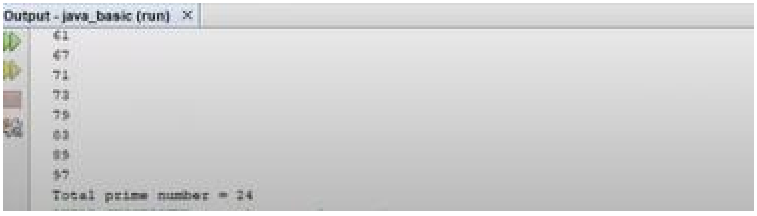
20. Write a Java Programme to check prime number m to n.

Input:

```
package begginnerjava;

import java.util.Scanner;
public class primetest {
   public static void main(String [] args){

     int m,n,count=0,total Prime=0;
     Scanner input =  new Scanner(System.in);
     System.out.print("Enter Initial Number  : ");
     m = input.nextInt();

     System.out.print("Enter last Number  : ");
     n = input.nextInt();

     for(int i=m; i<=n; i++)
   {
       for(int j=2; j<=i-1; j++){
        if(i%j==0){
            count++;
          break;
        }
}
        if(count==0){

        System.out.println(i);
        total Prime++;

   }
          count=0;

   }
```

```
            System.out.println("Total Prime : "+total prime);

    }

}
```

Output:



```
61
67
71
73
79
83
89
97
Total prime number = 24
```

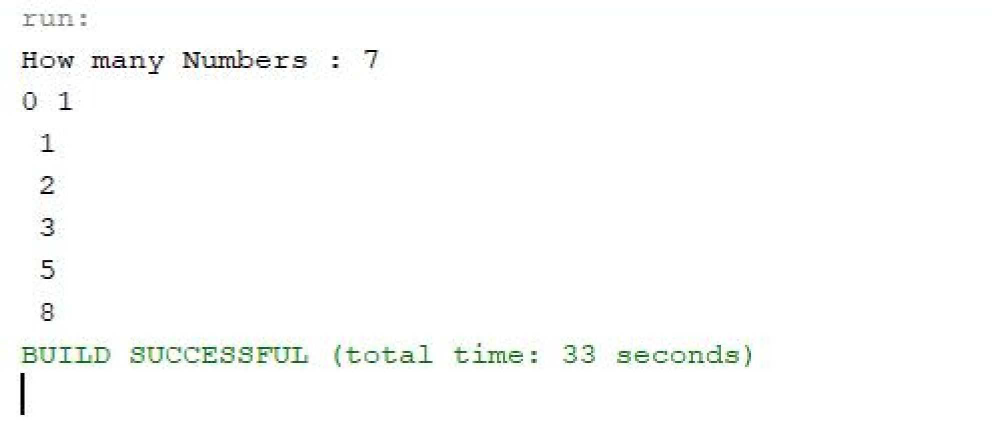
21. Write a Java Programme to check fibonaaci series.

Input:

```
package begginnerjava;
import java.util.Scanner;

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

    Scanner input = new Scanner(System.in);
    System.out.print("How many Numbers : ");
    int n = input.nextInt();

    int first = 0;
    int second = 1;
    int fibo;


    System.out.println(first+ " " +second);

      for(int i =3; i<=n; i++){

         fibo = first + second;
         System.out.println(" "+fibo);
         first = second;
         second=fibo;
```

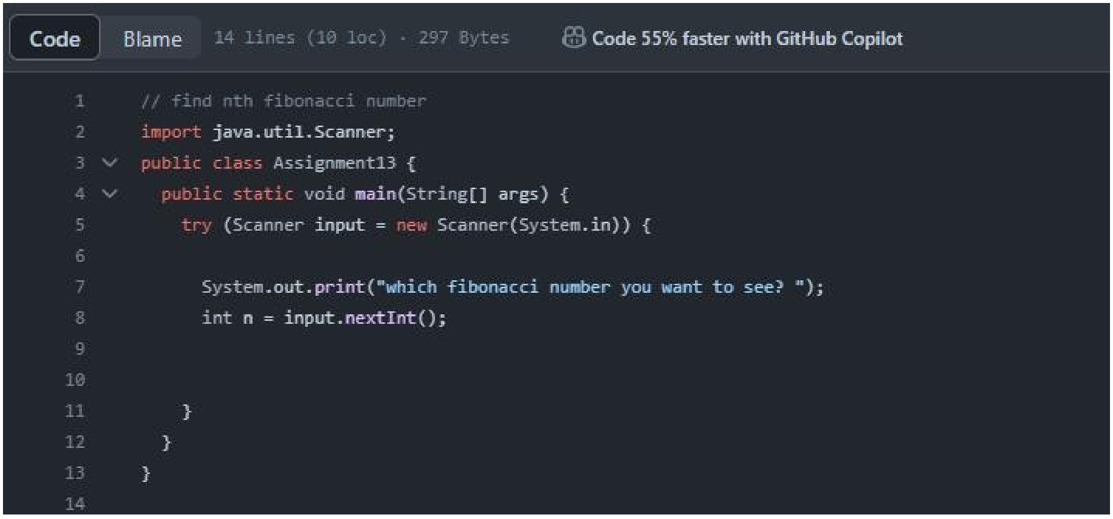
```
        }


    }

}
```

Output:



```
run:
How many Numbers : 7
0 1
 1
 2
 3
 5
 8
BUILD SUCCESSFUL (total time: 33 seconds)
```

22.   Write a Java Programme to check fibonaaci series.



Code
Blame
14 lines (10 loc) · 297 Bytes
Code 55% faster with GitHub Copilot

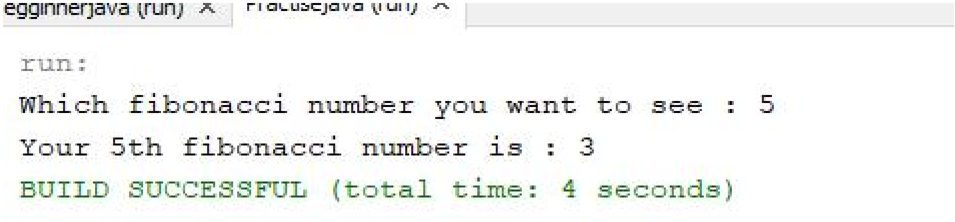
```java
// find nth fibonacci number
import java.util.Scanner;
public class Assignment13 {
  public static void main(String[] args) {
    try (Scanner input = new Scanner(System.in)) {

      System.out.print("which fibonacci number you want to see? ");
      int n = input.nextInt();


    }
  }
}

```

Input:

package practisejava;

```
import java.util.Scanner;
public class Assignment13 {
   public static void main(String [] args){


  Scanner x = new Scanner (System.in);
       System.out.print("Which fibonacci number you want to see : ");
       int n =x.nextInt();

       int first=0;
       int second=1;
       int fibo=0;

       if(n==1){
          System.out.println("Your "+n+"th fibonacci number is : "+first);
       }
       else if(n==2){
          System.out.println("Your "+n+"th fibonacci number is : "+second);
       }
       else{
          for(int i=3; i<=n; i=i+1){
             fibo=first+second;
             first=second;
             second=fibo;
          }
          System.out.println("Your "+n+"th fibonacci number is : "+fibo);
       }

   }
}
```

Output:

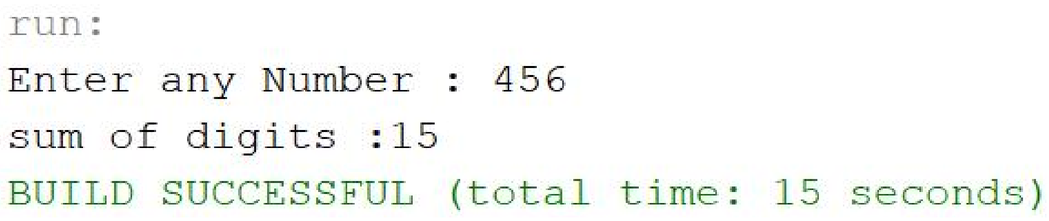


```
run:
Which fibonacci number you want to see : 5
Your 5th fibonacci number is : 3
BUILD SUCCESSFUL (total time: 4 seconds)
```

23.Write a Java Programme to check sum of digits.
Input:

```
package begginnerjava;

import java.util.Scanner;
public class Sumofdigits {
    public static void main(String [] args){

        Scanner input = new Scanner (System.in);
        int sum = 0 , r,temp,num;

        System.out.print("Enter any Number : ");
        num =  input.nextInt();
         temp = num;
          while (temp != 0){
             r = temp % 10;
             sum = sum + r;
             temp = temp /10;

          }

           System.out.println("sum of digits :"+sum);

    }


}
```

Output:



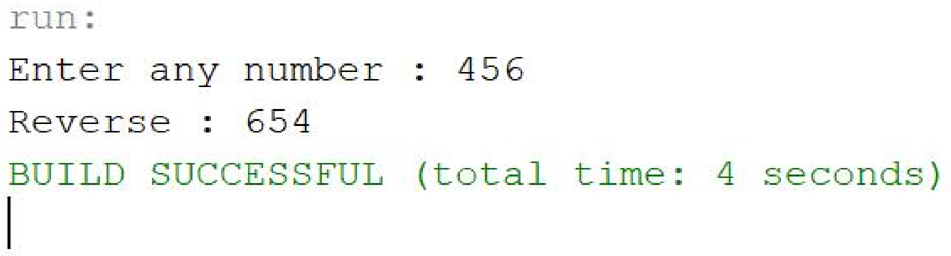
```
run:
Enter any Number : 456
sum of digits :15
BUILD SUCCESSFUL (total time: 15 seconds)
```

24.Write a Java Programme to check Reverse sum of digits

Input:

```
package begginnerjava;

import java.util.Scanner;
public class ReverseNumber {
    public static void main(String [] args){

        Scanner input = new Scanner(System.in);
        int num,sum=0,temp,r;
        System.out.print("Enter any number : ");
        num = input.nextInt();


        temp = num;
        while (temp!=0){

            r = temp % 10;
            sum  = sum *10 + r;
            temp = temp / 10;

        }

        System.out.println("Reverse : "+sum);
    }


}
```

Output:

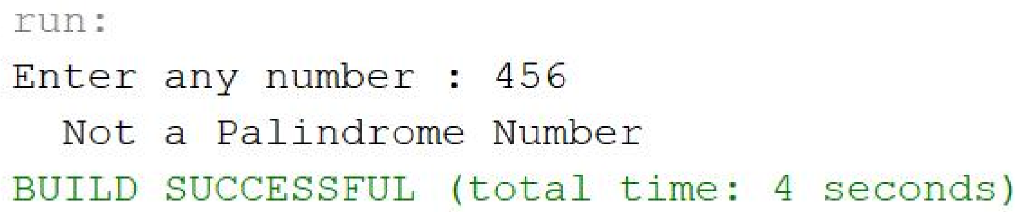


```
run:
Enter any number : 456
Reverse : 654
BUILD SUCCESSFUL (total time: 4 seconds)
```

25. Write a Java Programme to check Palindrome number.
Input:

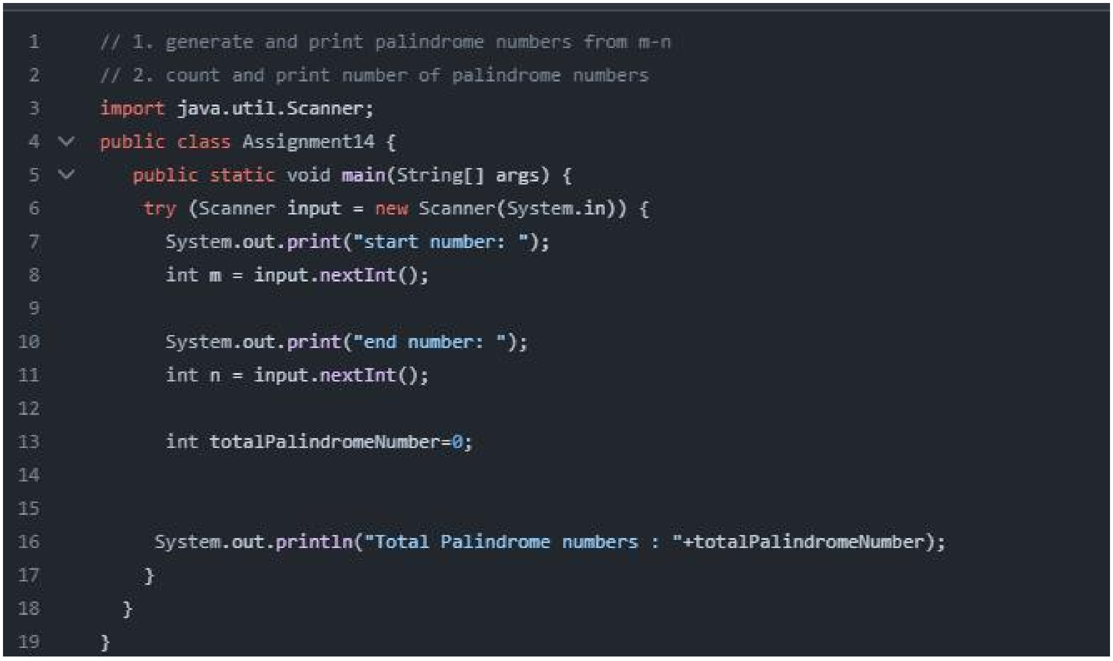
```
package begginnerjava;

import java.util.Scanner;
public class Palindromenumber {
  public static void main(String [] args){

    Scanner input = new Scanner(System.in);
    int num,sum=0,temp,r;
    System.out.print("Enter any number : ");
    num = input.nextInt();


    temp = num;
    while(temp!=0){

      r = temp % 10;
      sum  = sum *10 + r;
      temp = temp / 10;

    }

   if (num==sum){
     System.out.println("Palindrome Number");

   }

   else {

     System.out.println("  Not a Palindrome Number");
   }
  }
}
```

Output:



```
run:
Enter any number : 456
  Not a Palindrome Number
BUILD SUCCESSFUL (total time: 4 seconds)
```

26. Write a Java Programme to check Palindrome number m to n



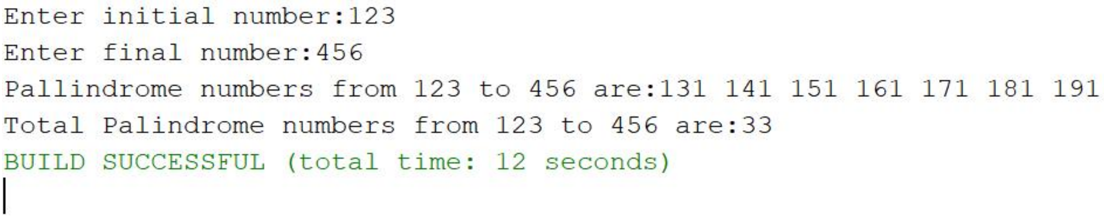
```
// 1. generate and print palindrome numbers from m-n
// 2. count and print number of palindrome numbers
import java.util.Scanner;
public class Assignment14 {
   public static void main(String[] args) {
    try (Scanner input = new Scanner(System.in)) {
      System.out.print("start number: ");
      int m = input.nextInt();

      System.out.print("end number: ");
      int n = input.nextInt();

      int totalPalindromeNumber=0;


     System.out.println("Total Palindrome numbers : "+totalPalindromeNumber);
    }
  }
}
```

Input:

```
package practisejava;

import java.util.Scanner;
public class Assignment14 {

   public static void main(String[] args) {
      Scanner input=new Scanner(System.in);
      int n1, n2, t, sum=0, count=0, r, i;
      System.out.print("Enter initial number:");
```

```
    n1=input.nextInt();
    System.out.print("Enter final number:");
    n2=input.nextInt();
    System.out.print("Pallindrome numbers from "+n1+" to "+n2+"
are:");
    for(i=n1; i<=n2; i++)
    {
      t=i;
      while(t!=0 && i>10)
      {
        r=t%10;
        sum=sum*10+r;
        t=t/10;
      }
      if(sum==i)
      {
        System.out.print(i+" ");
        count++;
      }
      sum=0;
    }
    System.out.println();
    System.out.println("Total Palindrome numbers from "+n1+" to
"+n2+" are:"+count);
  }
}




}
```

Output:



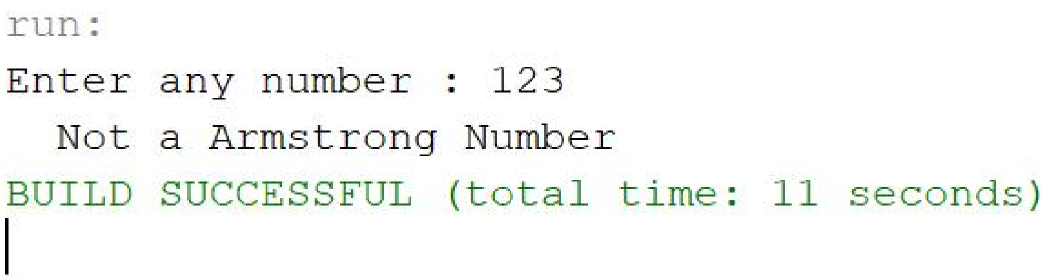
```
Enter initial number:123
Enter final number:456
Pallindrome numbers from 123 to 456 are:131 141 151 161 171 181 191
Total Palindrome numbers from 123 to 456 are:33
BUILD SUCCESSFUL (total time: 12 seconds)
```

27. Write a Java Programme to check Armstone number.

Input:

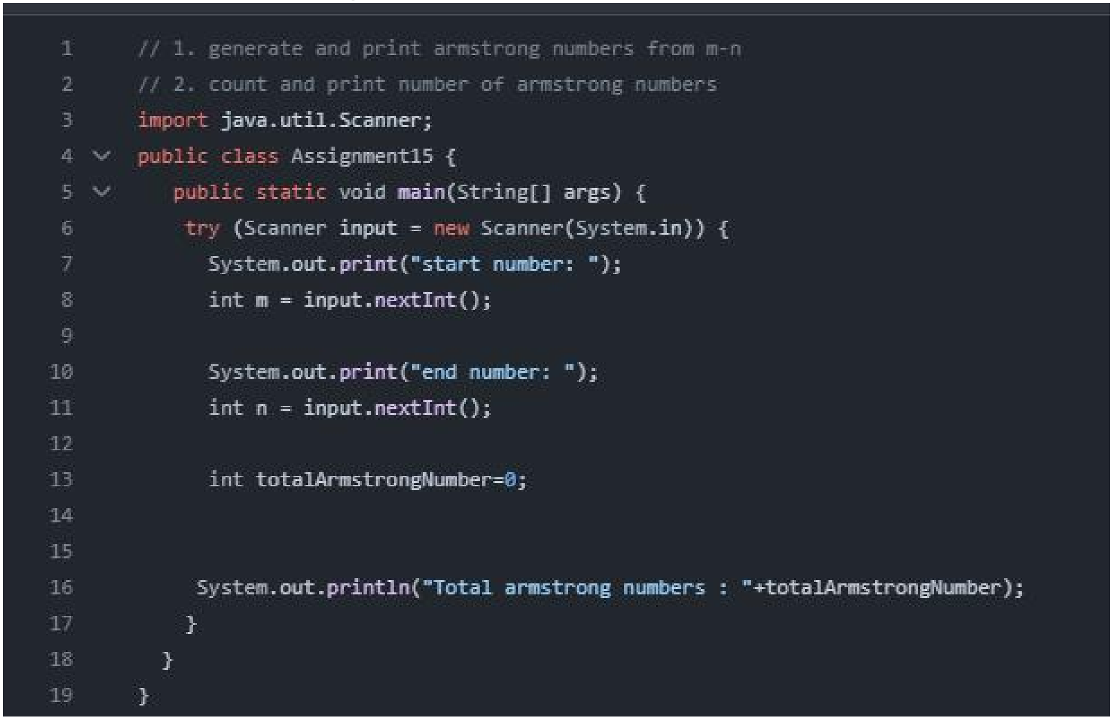
```
package begginnerjava;
import java.util.Scanner;
public class ArmstoneNumber {

    public static void main(String [] args){

        Scanner input = new Scanner(System.in);
        int num,sum=0,temp,r;
        System.out.print("Enter any number : ");
        num = input.nextInt();


        temp = num;
        while(temp!=0){

            r = temp % 10;
            sum  = sum *10 + r * r *r;
            temp = temp / 10;

        }

        if (num==sum){
           System.out.println("Armstrong Number ");

       }

       else {

          System.out.println("  Not a Armstrong Number");
       }


}
```

Output:



```
run:
Enter any number : 123
  Not a Armstrong Number
BUILD SUCCESSFUL (total time: 11 seconds)
```

28. Write a Java Programme to chech m to n Armstone Number.

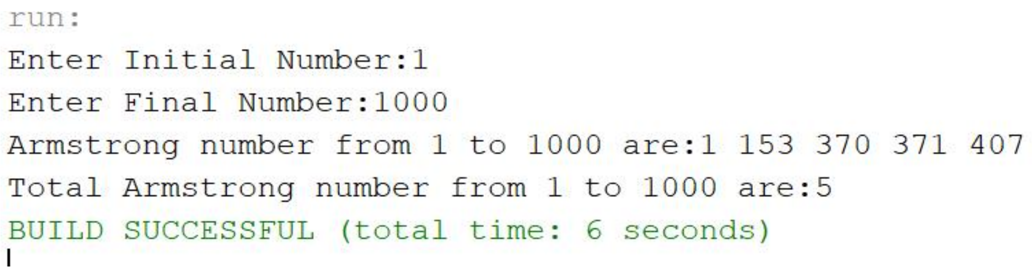


```java
// 1. generate and print armstrong numbers from m-n
// 2. count and print number of armstrong numbers
import java.util.Scanner;
public class Assignment15 {
   public static void main(String[] args) {
    try (Scanner input = new Scanner(System.in)) {
      System.out.print("start number: ");
      int m = input.nextInt();

      System.out.print("end number: ");
      int n = input.nextInt();

      int totalArmstrongNumber=0;


     System.out.println("Total armstrong numbers : "+totalArmstrongNumber);
    }
  }
}
```

Input:

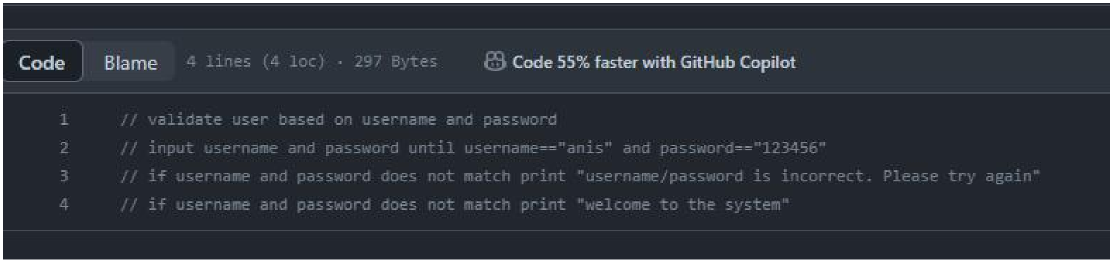
```
package practisejava;
import java.util.Scanner;
public class Assignment15 {
  public static void main(String[] args) {
        Scanner input=new Scanner(System.in);
        int n1, n2, i, temp, r, sum=0, count=0;
        System.out.print("Enter Initial Number:");
        n1=input.nextInt();
        System.out.print("Enter Final Number:");
        n2=input.nextInt();
        System.out.print("Armstrong number from "+n1+" to "+n2+" are:");
        for(i=n1; i<=n2; i++)
        {
            temp=i;
            while(temp!=0)
            {
                r=temp%10;
                sum=sum+r*r*r;
                temp=temp/10;
            }
            if(i==sum)
            {
                System.out.print(i+" ");
                count++;
            }
            sum=0;
        }
        System.out.println();
        System.out.println("Total Armstrong number from "+n1+" to "+n2+"
are:"+count);
    }
}

}
```

Output:



```
run:
Enter Initial Number:1
Enter Final Number:1000
Armstrong number from 1 to 1000 are:1 153 370 371 407
Total Armstrong number from 1 to 1000 are:5
BUILD SUCCESSFUL (total time: 6 seconds)
```

29.   Write a Java Programme Using Loop M to N.



Code
Blame
4 lines (4 loc) · 297 Bytes
Code 55% faster with GitHub Copilot

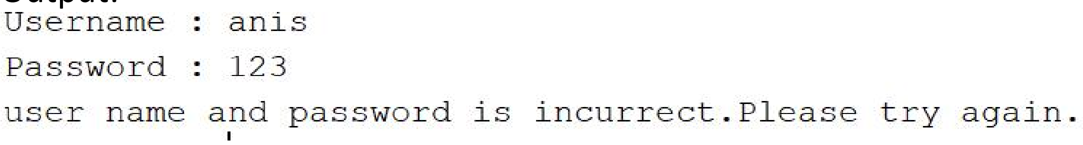
```
// validate user based on username and password
// input username and password until username=="anis" and password=="123456"
// if username and password does not match print "username/password is incorrect. Please try again"
// if username and password does not match print "welcome to the system"
```

Input:

```
package practisejava;

import java.util.Scanner;
public class Assignment16 {
  public static void main(String[] args) {

      Scanner input = new Scanner( System.in);

      for ( int x = 0; x<=10 ; x++){

         System.out.print("Username : ");
         String username = input.next();
         System.out.print("Password : ");
         int pass = input.nextInt();

         if( pass == 123456 && username.contains("anis")){
            System.out.println("Well come to the system");
            break;
         }else{
            System.out.println("user name and password is incurrect.Please
try again.");
         }
      }
```

```
    }
}

}
```

Output:



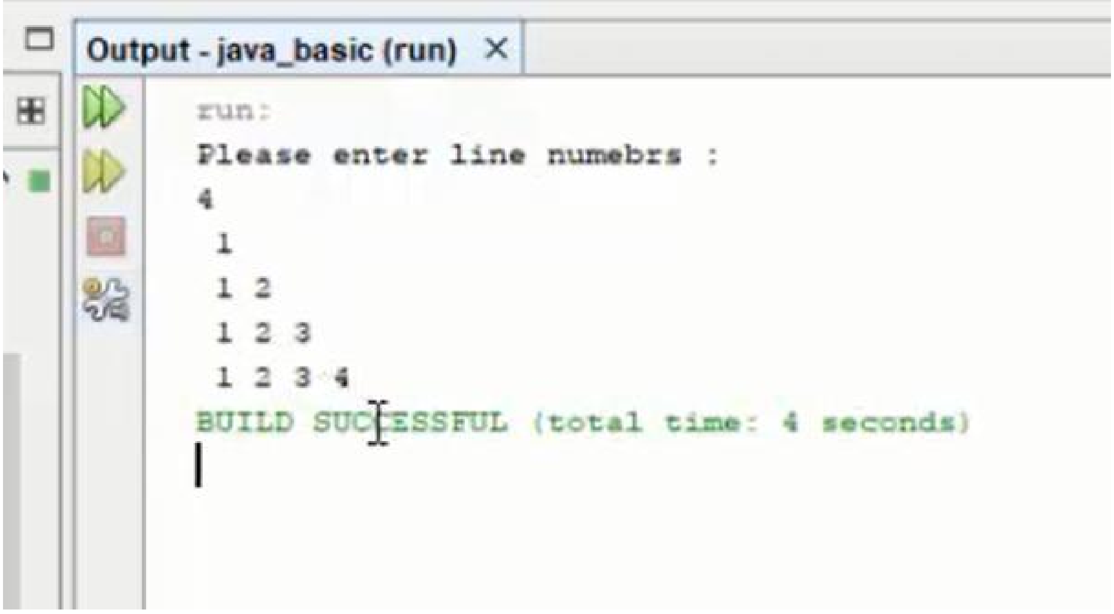
```
Username : anis
Password : 123
user name and password is incurrect.Please try again.
```

30. Write a Java Programme Pattern Using Loop.

Input:

```
package begginnerjava;
import java.util.Scanner;
public class pattern01 {
    public static void main(String [] args){
        Scanner input = new Scanner(System.in);


        System.out.print("Enter the line numbers : ");
        int n = input.nextInt();

        for (int row =1; row<=n; row++){

            for (int col =1; row<=n; col++){

                System.out.println(" "+col);
            }
              System.out.printrln();

        }





    }
```

}
Output:

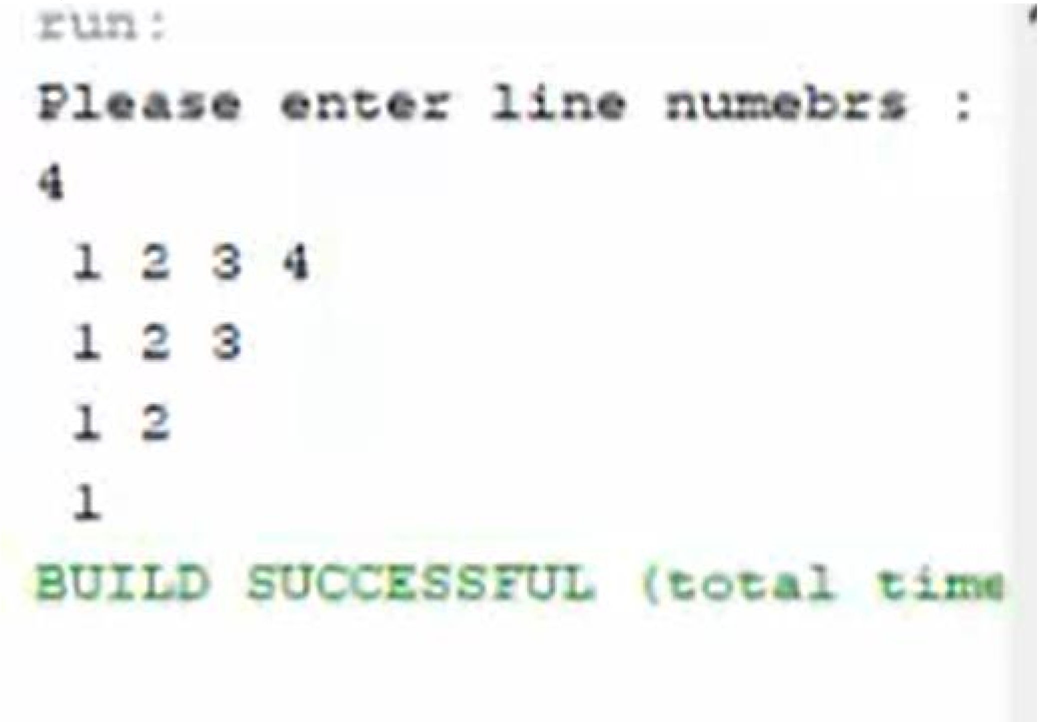


```
run:
Please enter line numebrs :
4
 1
 1 2
 1 2 3
 1 2 3 4
BUILD SUCCESSFUL (total time: 4 seconds)
```

31. Write a Java Programme Pattern Triangle .
Input:

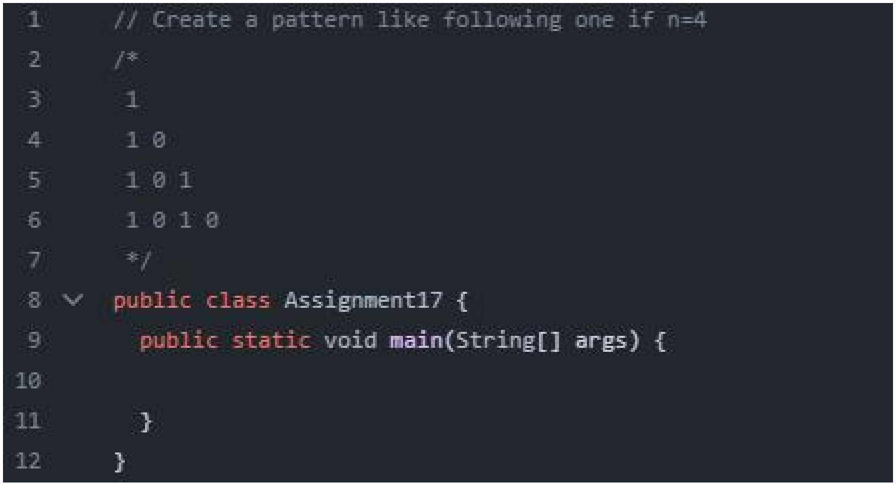
```
package begginnerjava;
import java.util.Scanner;
public class pattern2 {
   public static void main(String [] args){
      Scanner input = new Scanner(System.in);


      System.out.print("Enter the line numbers : ");
      int n = input.nextInt();

      for (int row =n; row>=1; row--){

         for (int col =1; col<=row; col++){

            System.out.print(" "+col);
         }
           System.out.printrln();
      }



   }

}
```

Output:



```
run:
Please enter line numebrs :
4
 1 2 3 4
 1 2 3
 1 2
 1
BUILD SUCCESSFUL (total time
```

32. Write a Java Programme Pattern related .

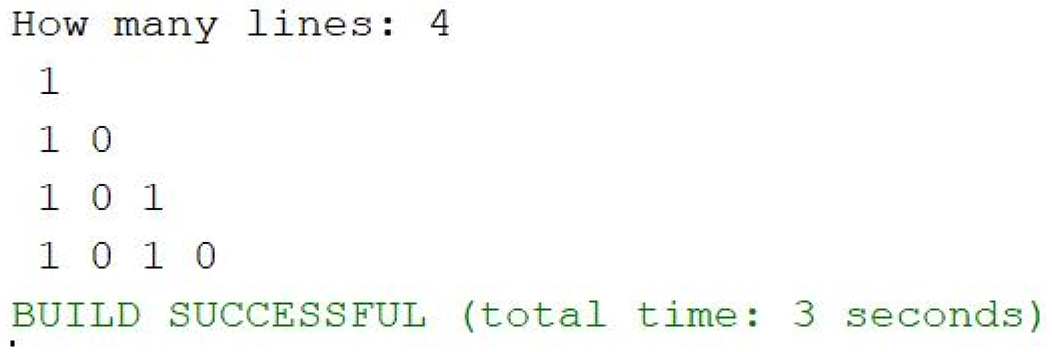


```
// Create a pattern like following one if n=4
/*
 1
 1 0
 1 0 1
 1 0 1 0
 */
public class Assignment17 {
  public static void main(String[] args) {

  }
}
```

Input:

```
package practisejava;
import java.util.Scanner;
public class Assignment17 {

    public static void main(String [] args){

       Scanner in= new Scanner(System.in);
        int n, row, col;
        System.out.print("How many lines: ");
        n = in.nextInt();
        in.close();
        for(row=1; row<=n; row++){
            for(col=1; col<=row; col++){
                System.out.print(" "+col%2);
            }
            System.out.println(" ");
        }



    }



}
```

Output:



```
How many lines: 4
 1
 1 0
 1 0 1
 1 0 1 0
BUILD SUCCESSFUL (total time: 3 seconds)
```