

Name: saad jamil

ID:15889

Q.1 How to check Even and Odd numbers in java using object oriented approach?

ANS. To check Even and Odd number in java we use (IF , ELSE) for Example

To make the program we will take these important steps:

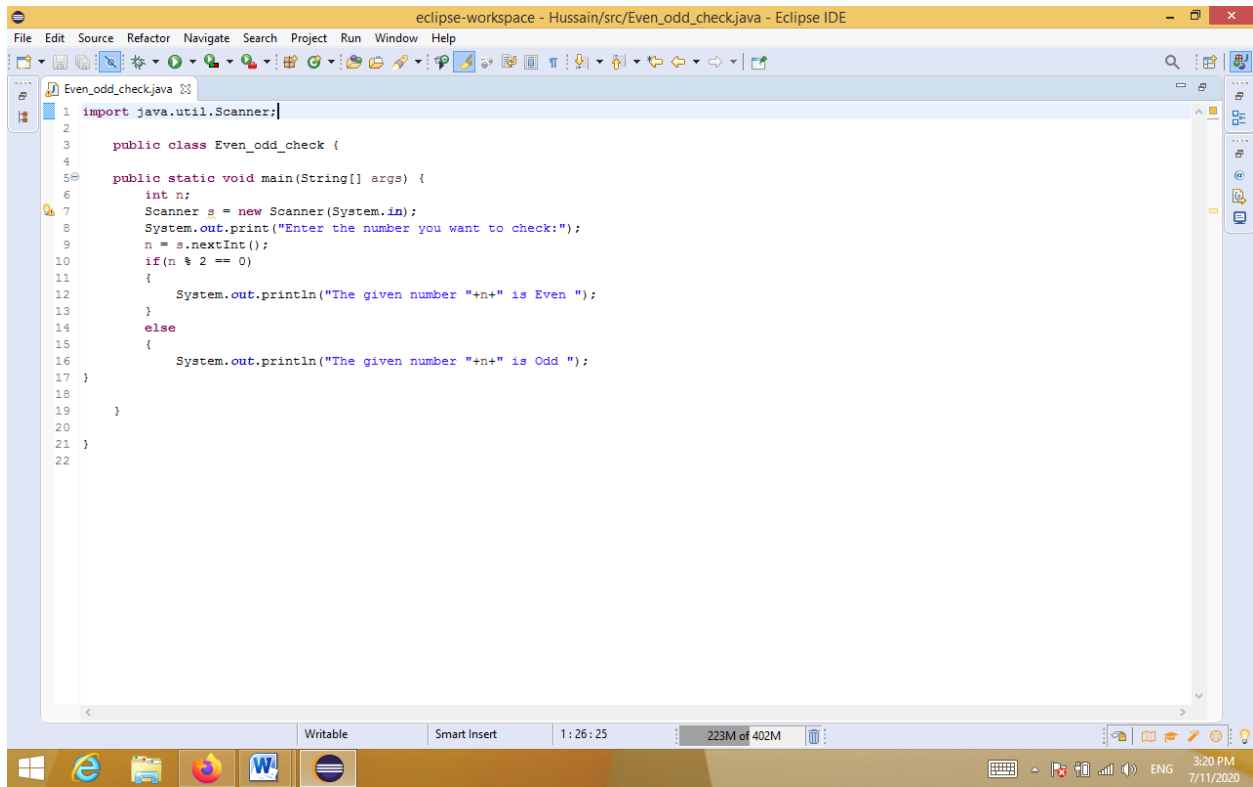
we use **IF statement**

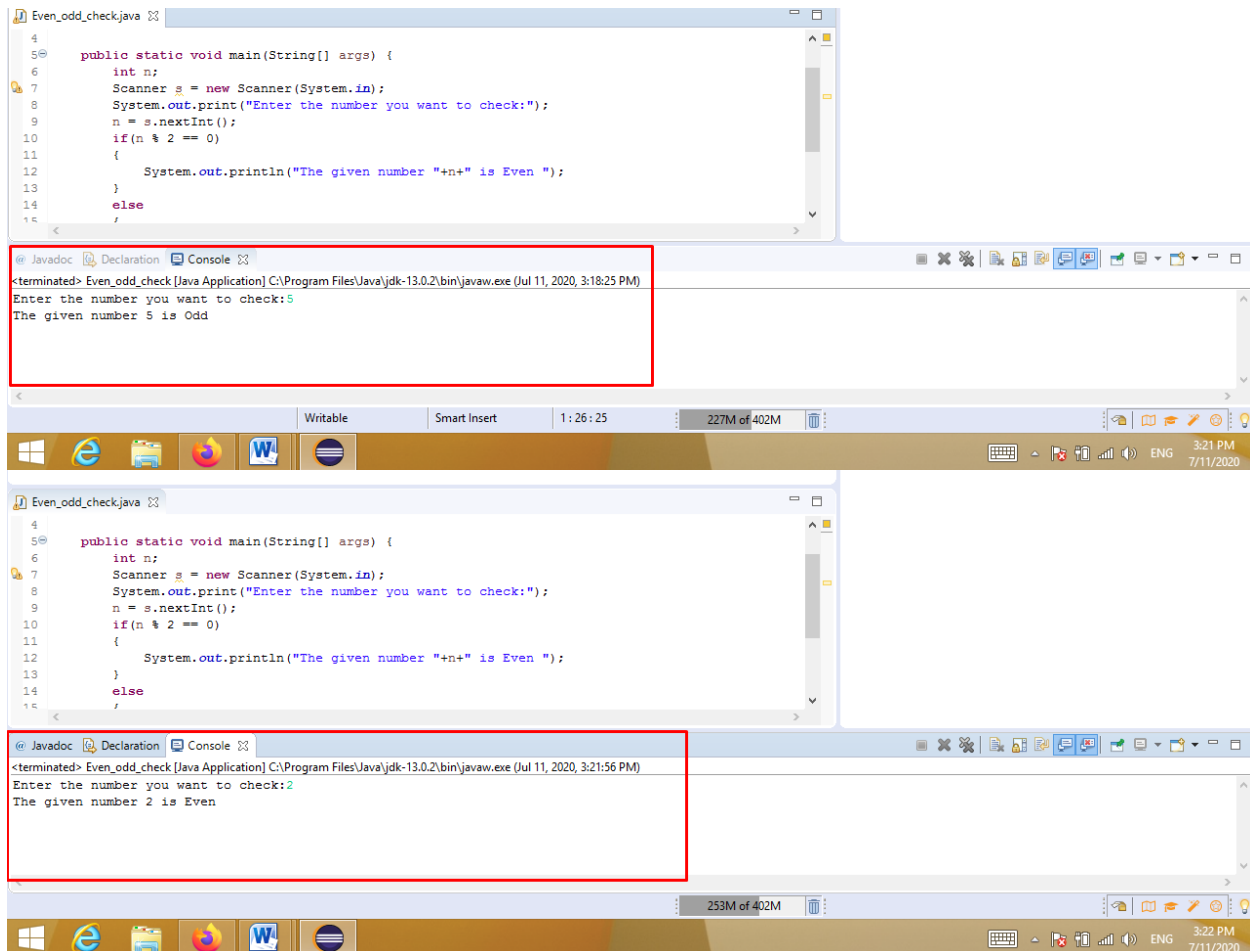
```
if(n % 2 == 0)
{
    System.out.println("The given number is Even ");
}
```

Else Statement

```
else
{
    System.out.println("The given number is Odd ");
}
```

Program in Eclipse with output





Q.2 How to add 2 complex numbers in java using object oriented approach?

ANS. **Complex numbers** have two parts – real part and imaginary part. When adding complex numbers we add real parts together and imaginary parts together. In this program we have a class `ComplexNumber`. In this class we have two instance variables `real` and `imaginary` to hold the real and imaginary parts of complex numbers.

We have declared a method `sum()` to add the two numbers by adding their real and imaginary parts together.

The constructor of this class is used for initializing the complex numbers. For e.g. when we create an instance of this class like this `ComplexNumber NUMBER = new ComplexNumber(0, 0);`, it actually creates a complex number $0 + 0i$.

Program in Eclipse:

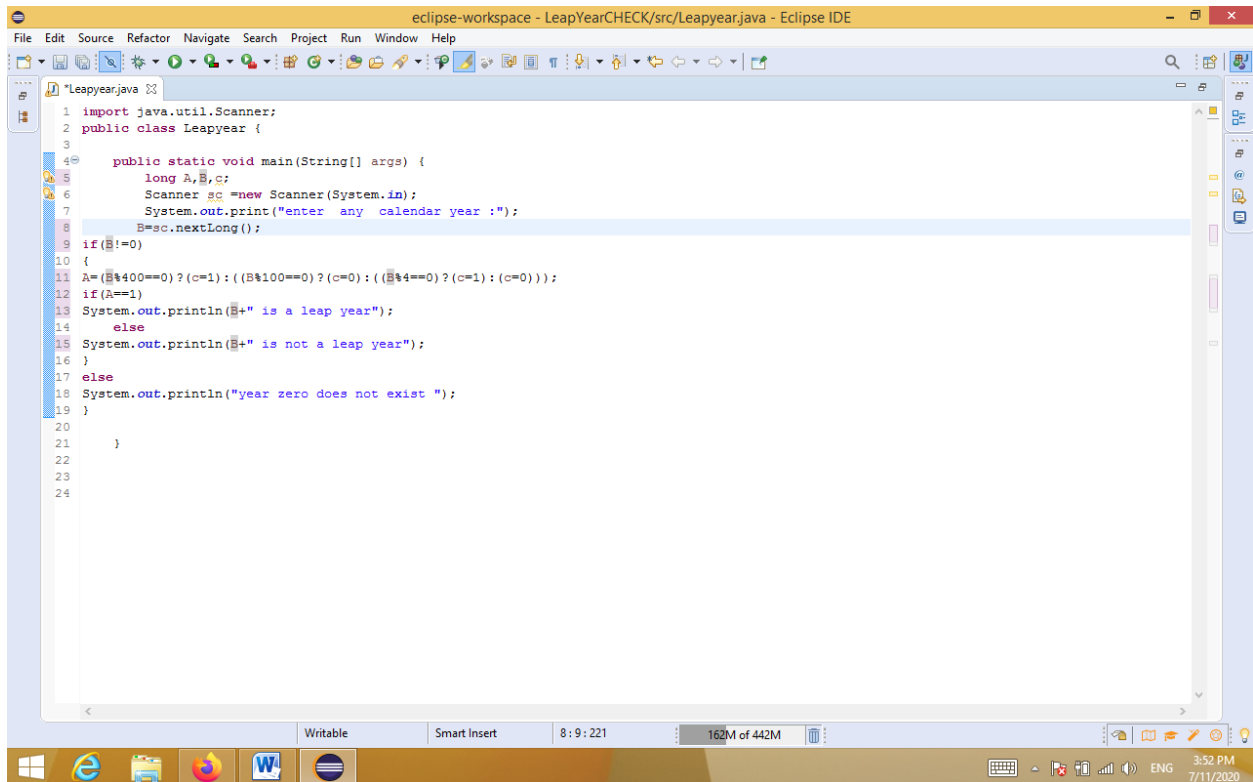
```
eclipse-workspace - ComplexNumber/src/ComplexNumber.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
ComplexNumber.java
1
2 public class ComplexNumber {
3     double real, img;
4     ComplexNumber(double r, double i){
5         this.real = r;
6         this.img = i;
7     }
8     public static ComplexNumber sum(ComplexNumber c1, ComplexNumber c2)
9     {
10        ComplexNumber NUMBER = new ComplexNumber(0, 0);
11        NUMBER.real = c1.real + c2.real;
12        NUMBER.img = c1.img + c2.img;
13        return NUMBER;
14    }
15    public static void main(String[] args) {
16        ComplexNumber c1 = new ComplexNumber(10.9, 4.4);
17        ComplexNumber c2 = new ComplexNumber(11.2, 32.15);
18        ComplexNumber temp = sum(c1, c2);
19        System.out.printf("Sum is: "+ temp.real+" "+ temp.img +"i");
20    }
21 }
22
23
24
```

```
ComplexNumber.java
11        NUMBER.real = c1.real + c2.real;
12        NUMBER.img = c1.img + c2.img;
13        return NUMBER;
14    }
15    public static void main(String[] args) {
16        ComplexNumber c1 = new ComplexNumber(10.9, 4.4);
17        ComplexNumber c2 = new ComplexNumber(11.2, 32.15);
18        ComplexNumber temp = sum(c1, c2);
19        System.out.printf("Sum is: "+ temp.real+" "+ temp.img +"i");
20    }
21 }
22
```

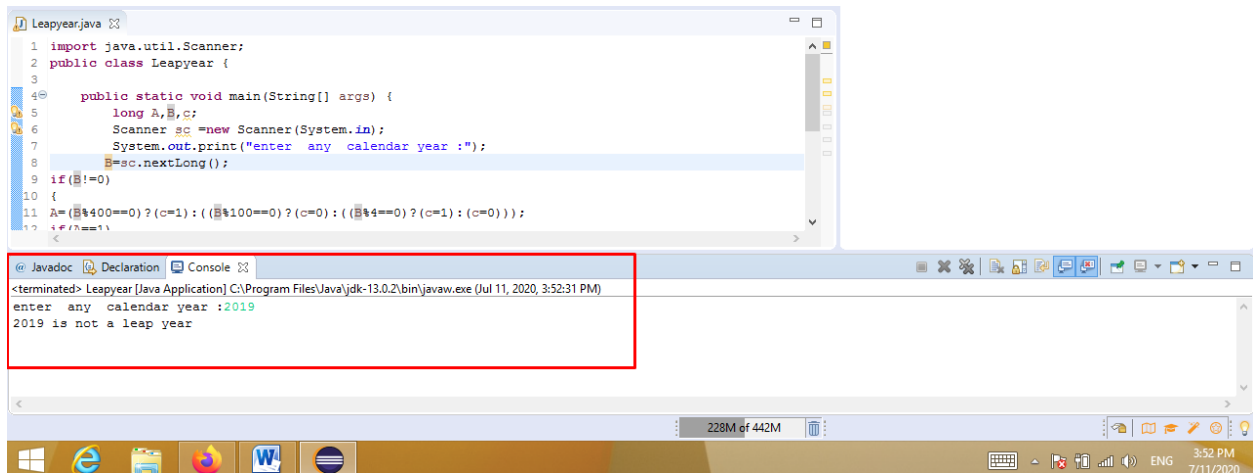
```
@ Javadoc Declaration Console
<terminated> ComplexNumber [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Jul 11, 2020, 3:39:52 PM)
Sum is: 22.1 + 36.55i
```

q.3 How to check Leap year in java using object oriented approach

ANS.



```
1 import java.util.Scanner;
2 public class Leapyear {
3
4     public static void main(String[] args) {
5         long A,B,C;
6         Scanner sc =new Scanner(System.in);
7         System.out.print("enter any calendar year :");
8         B=sc.nextLong();
9         if(B!=0)
10        {
11            A=(B%400==0)?(c=1):((B%100==0)?(c=0):((B%4==0)?(c=1):(c=0)));
12            if(A==1)
13                System.out.println(B+" is a leap year");
14            else
15                System.out.println(B+" is not a leap year");
16        }
17        else
18            System.out.println("year zero does not exist ");
19    }
20
21    }
22
23
24
```



```
1 import java.util.Scanner;
2 public class Leapyear {
3
4     public static void main(String[] args) {
5         long A,B,C;
6         Scanner sc =new Scanner(System.in);
7         System.out.print("enter any calendar year :");
8         B=sc.nextLong();
9         if(B!=0)
10        {
11            A=(B%400==0)?(c=1):((B%100==0)?(c=0):((B%4==0)?(c=1):(c=0)));
12            if(A==1)
13                System.out.println(B+" is a leap year");
14            else
15                System.out.println(B+" is not a leap year");
16        }
17        else
18            System.out.println("year zero does not exist ");
19    }
20
21    }
22
23
24
```

@ Javadoc Declaration Console

```
<terminated> Leapyear [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Jul 11, 2020, 3:52:31 PM)
enter any calendar year :2019
2019 is not a leap year
```

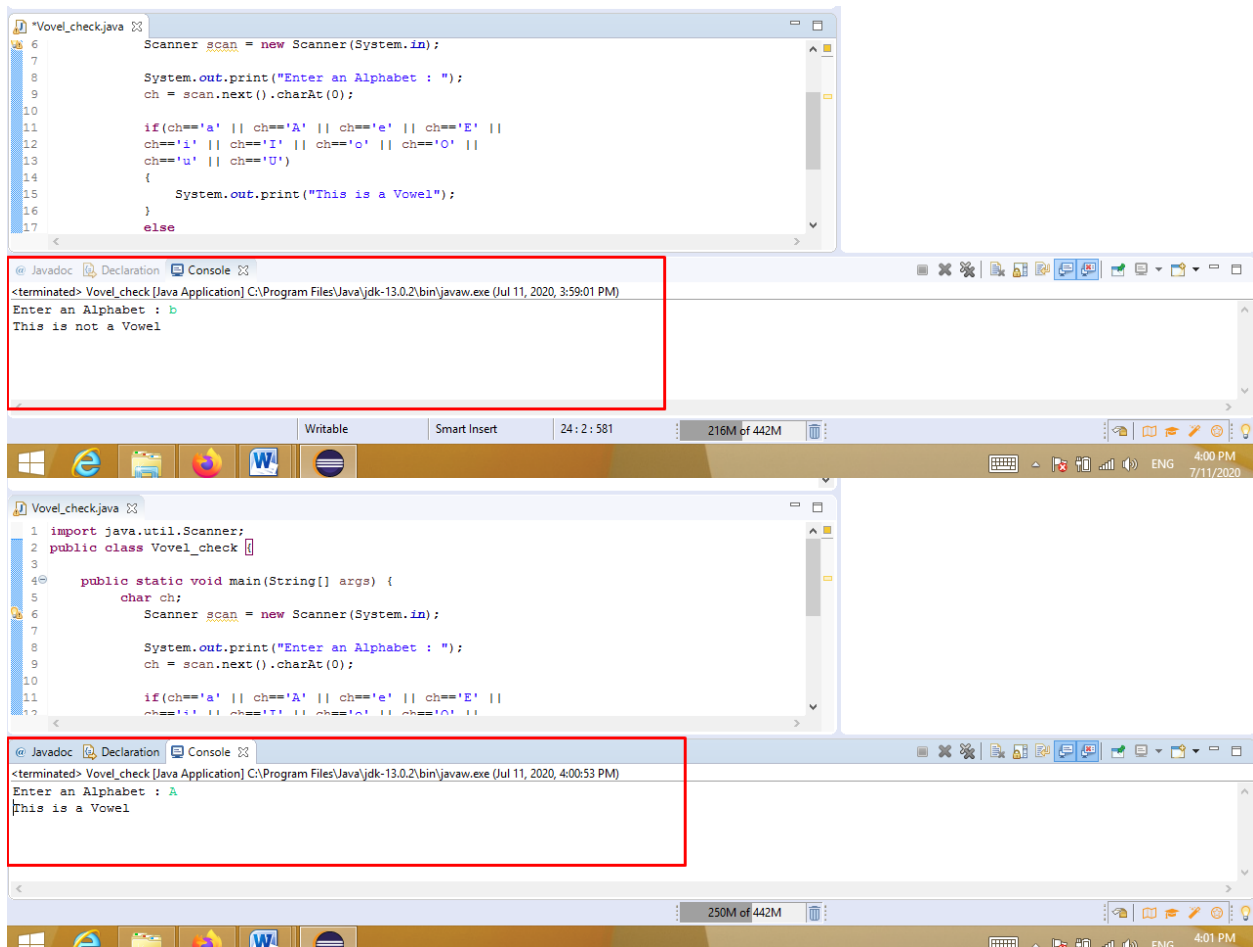
```
Leapyear.java
1 import java.util.Scanner;
2 public class Leapyear {
3
4     public static void main(String[] args) {
5         long A,B,C;
6         Scanner sc =new Scanner(System.in);
7         System.out.print("enter any calendar year :");
8         B=sc.nextLong();
9         if(B!=0)
10        {
11            A=(B%400==0)?(C=1):((B%100==0)?(C=0):((B%4==0)?(C=1):(C=0)));
12        }
13    }
14 }
```

@ Javadoc Declaration Console
<terminated> Leapyear [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Jul 11, 2020, 3:54:04 PM)
enter any calendar year :2020
2020 is a leap year

Q.4 How to check that the input from the user is the vowel or not in java using object oriented approach?

ANS. Same procedure as Question number 1

```
eclipse-workspace - Vowel check/src/Vowel_check.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Vowel_check.java
1 import java.util.Scanner;
2 public class Vowel_check {
3
4     public static void main(String[] args) {
5         char ch;
6         Scanner scan = new Scanner(System.in);
7
8         System.out.print("Enter an Alphabet : ");
9         ch = scan.next().charAt(0);
10
11         if(ch=='a' || ch=='A' || ch=='e' || ch=='E' ||
12            ch=='i' || ch=='I' || ch=='o' || ch=='O' ||
13            ch=='u' || ch=='U')
14         {
15             System.out.print("This is a Vowel");
16         }
17         else
18         {
19             System.out.print("This is not a Vowel");
20         }
21     }
22 }
23
24 }
25 }
```



q.5 How to use power of a number in java using object oriented approach?

Ans.

Read the base and exponent values from the user. Multiply the base number by itself and multiply the resultant with base (again) repeat this n times where n is the exponent value.

```
eclipse-workspace - POWER of NUMVER/src/Powerofnumber.java - Eclipse IDE
File Edit Source Refactor Navigate Search Project Run Window Help
Powerofnumber.java
1 import java.util.Scanner;
2 public class Powerofnumber {
3
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         System.out.println("Enter the base number ::");
7         int base = sc.nextInt();
8         int temp = base;
9         System.out.println("Enter the exponent number ::");
10        int exp = sc.nextInt();
11
12        for (int i=1; i<exp; i++){
13            temp = temp*temp;
14        }
15        System.out.println("Result of "+base+" power "+exp+" is "+temp);
16    }
17
18 }
19
20
21
```

```
*Powerofnumber.java
1 import java.util.Scanner;
2 public class Powerofnumber {
3
4     public static void main(String[] args) {
5         Scanner sc = new Scanner(System.in);
6         System.out.println("Enter the base number ::");
7         int base = sc.nextInt();
8         int temp = base;
9         System.out.println("Enter the exponent number ::");
10        int exp = sc.nextInt();
11
12        for (int i=1; i<exp; i++){
```

```
@ Javadoc Declaration Console
<terminated> Powerofnumber [Java Application] C:\Program Files\Java\jdk-13.0.2\bin\javaw.exe (Jul 11, 2020, 4:05:25 PM)
Enter the base number ::
2
Enter the exponent number ::
2
Result of 2 power 2 is 4
```