Note:-in question 1 and 2 we are using scanner class to take input from the user.

Ques 1. Write a java program Add two Numbers.

Explanation:-

To solve this question we will first take two random number and store them in variables. We can either take our own random numbers or take user input using scanner class of java util.

For example, we took one variable "a" and assigned it value 4 and took another variable "b" and assigned it value 6.

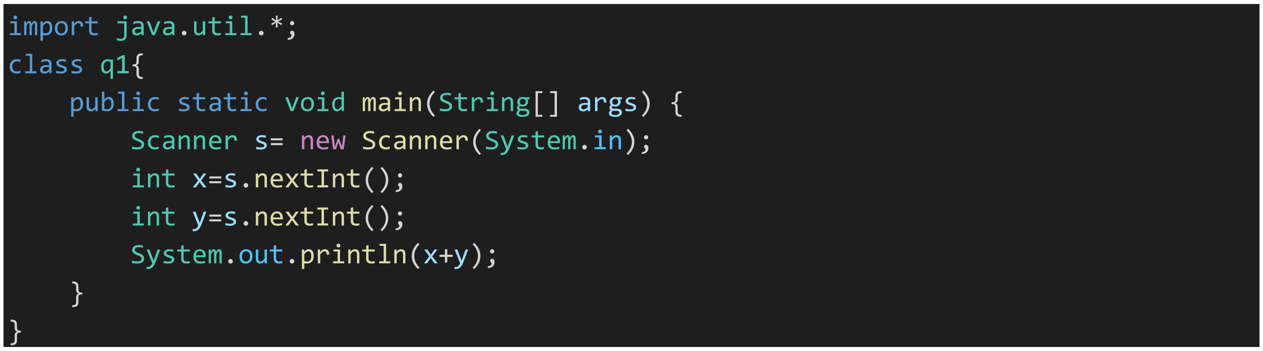
Now with the use of arithmetic operator we can easily print the sum of two numbers on screen or store it in another variable so that we can use it another time.

Sum= a + b

Where "+" is arithmetic operator used to add two values.

We use int datatype to store integer values in our variables.

Code:-



```
import java.util.*;
class q1{
    public static void main(String[] args) {
        Scanner s= new Scanner(System.in);
        int x=s.nextInt();
        int y=s.nextInt();
        System.out.println(x+y);
    }
}
```

Ques 2. Write a java program Check Whether a Number is Even or Odd.

Explanation:-

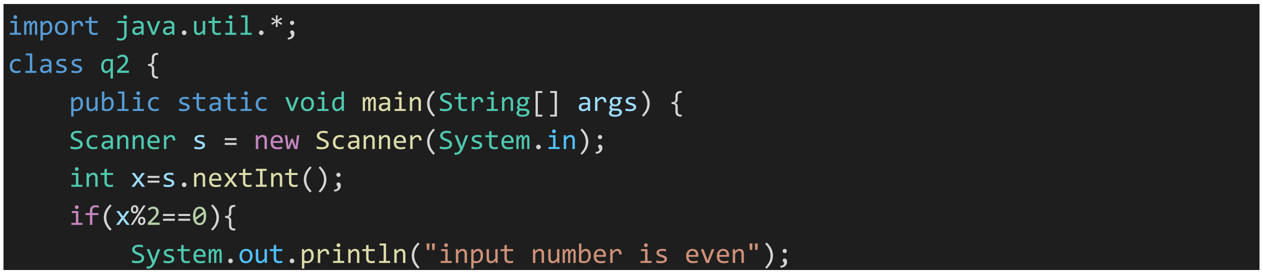
As we know that an even number is always divisible by 2 and gives the remainder 0 and odd number does not give the remainder 0 while dividing with 2. By using this logic we can create our program that can check if any number is odd or even.

If(x%2==0),then print number is even.

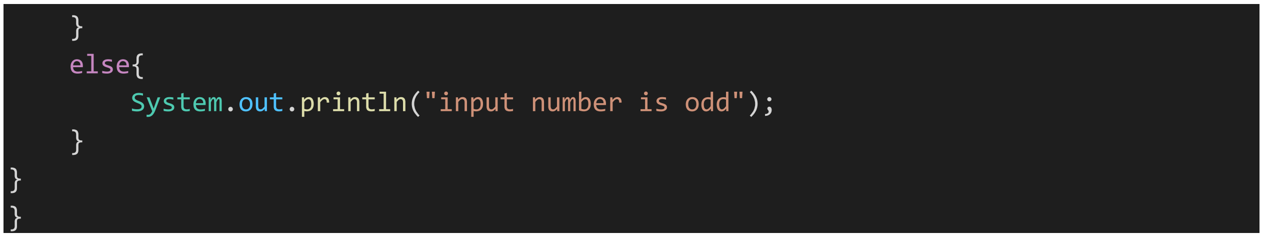
Else,print number is odd.

Here we use if else condition that provides an suitable condition to check if our number is gives 0 or not after dividing with 2.

Code:-



```
import java.util.*;
class q2 {
    public static void main(String[] args) {
    Scanner s = new Scanner(System.in);
    int x=s.nextInt();
    if(x%2==0){
        System.out.println("input number is even");
```



```
    }
    else{
        System.out.println("input number is odd");
    }
}
}
```

Ques 3. Write a java program Check if a given number is palindrome or not.

Explanation:-

To check if a number is palindrome or not we need to first know how to reverse an integer number in java.

We can use an different variable to store that reverse value.

Now we first need to make a logic to reverse our int value. For that we will use while loop.lets take an **integer n=696, temp=n, rev=0, rem**.

We will run while loop with condition temp!=0 means our loop will run until this condition becomes false i.e., when becomes equals to 0.

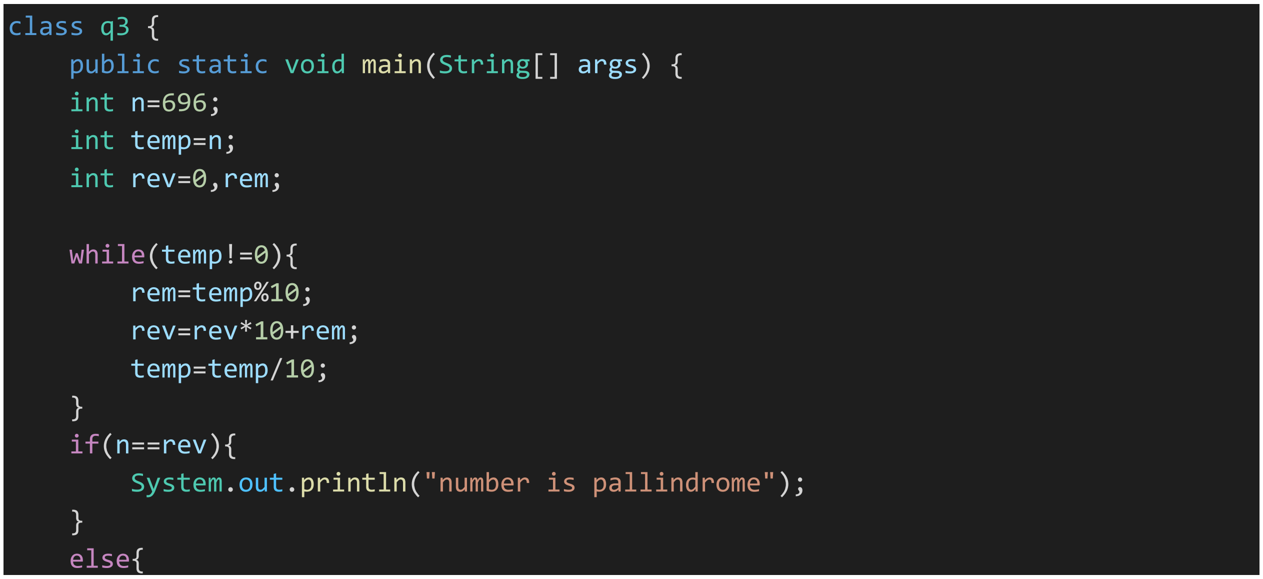
Inside this loop we will use our other variables as, rem=temp%10 , which will give the last digit of our number stored in n (as temp=n) as the last digit will come as modulo(%) or remainder which will be stored inside rem.

rev=rev*10+rem , the value of rev we declared earlier was 0 and multiplying it with 10 will give 0 and we are also adding the value of rem to it after that, this variable will store our reversed value. temp=temp/10, the final thing we will write in our loop will be this equation which means that everytime the value of temp will get divided by 10 and as the datatype of variables we use here is int, the decimal value will not get stored in the temp variable, hence each time last digit of temp will be remove using this.

We can close our loop now.

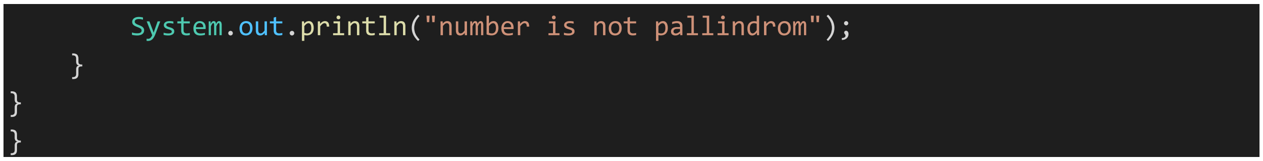
Now we will use the if else condition n==rev , as the the logic we built in the loop will reverse the value of n and store it in rev.

Code:-



```
class q3 {
    public static void main(String[] args) {
    int n=696;
    int temp=n;
    int rev=0,rem;

    while(temp!=0){
        rem=temp%10;
        rev=rev*10+rem;
        temp=temp/10;
    }
    if(n==rev){
        System.out.println("number is pallindrome");
    }
    else{
```



```
        System.out.println("number is not pallindrom");
    }
}
}
```

Ques 4. Write a java program to find the sum of n natural numbers.

Explanation:-

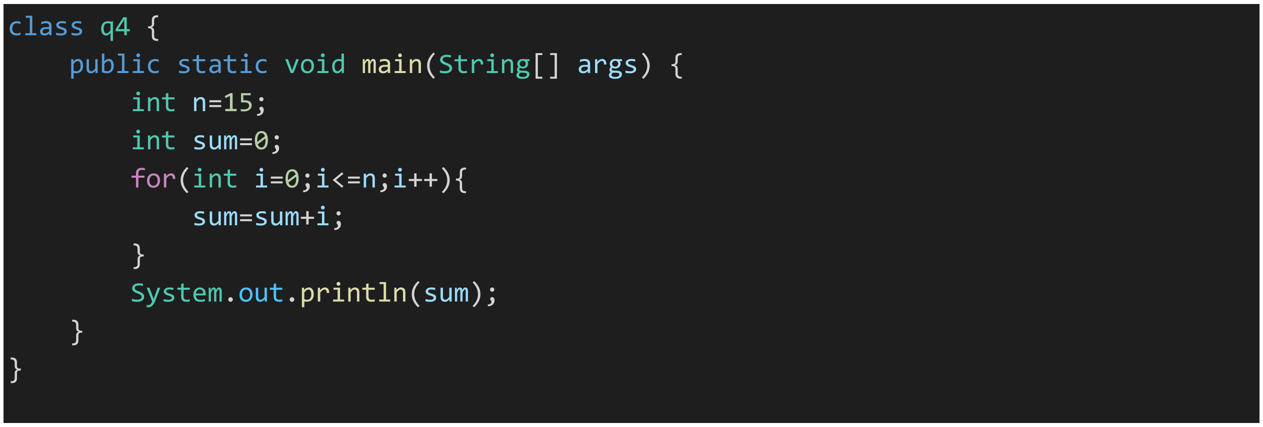
In this problem we will simply run our for loop and will declare two variables, n and sum where n will be the number that will tell from upto which number we need addition, sum variable will store the addition of all the numbers.

For(int i=0;i<=n;i++),this condition of for loop tells that the value of int i that is declared in the loop is 0,and i must be smaller than or equal to n so that it will no go beyond our desired value, and in the end of every loop the value of i will be increased because of i++.

Inside this loop we will run simple equation that is sum=sum+1,this equation will add each and every number till we get the end number that will terminate our loop.

After end of the loop we will get our desired value stored in the variable sum. And can see the value by printing it.

Code:-



```
class q4 {
    public static void main(String[] args) {
        int n=15;
        int sum=0;
        for(int i=0;i<=n;i++){
            sum=sum+i;
        }
        System.out.println(sum);
    }
}
```

Ques 5. Write a java program to Check Prime Number or not.

Explanation:-

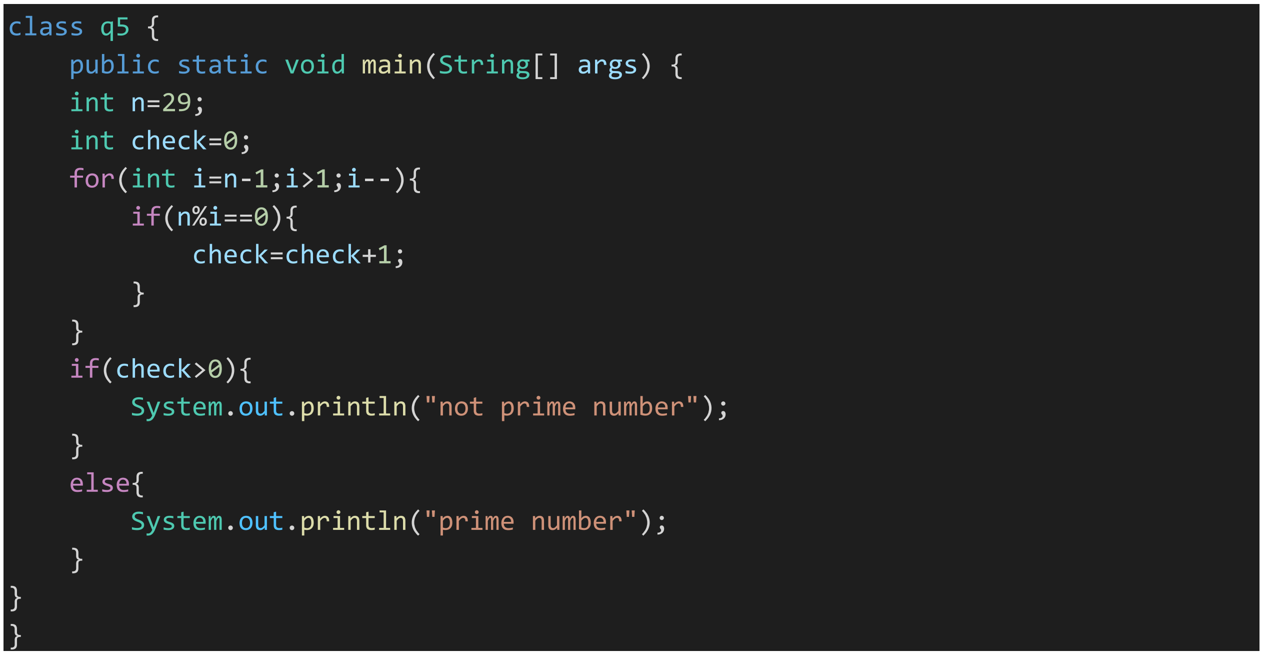
A prime number is the number that is only be divisible by 1 and itself. Which means our logic for solving this question should be around this statement.

Here we will declare two integer variables i.e.,
int n= 29,check=0

n is the value that we need to check if its prime or not and check is the variable which we will use so that we can check our number is prime or not.
For this problem we will use both for loop as well as if else condition to get our desired solution.
For(int i=n-1;i>1;i--), the i variables declared in the loop is equal to n-1 because the value of I must not be equal to or greater the n, as all the numbers are divisible by themselves.
i>1, the value of I should be greater than 1 as every number is divisible by 1 also means we are checking the numbers between n and 1 whether they are dividing our n or not.
i--,afer every run of loop our value of i will get decreased.
Inside this loop we will use our first if condition,
If(n%i==0),this condition checks whether any number in the loop is dividing our number and giving zero as a remainder or not. Inside this condition we will mention our other variable that is check,
check=check+1,the value of check declared earlier was 0 and if the condition is true it will increase and its value will increase more if our number is getting divided by multiple numbers.

We will end our loop here.
And after that we will give another condition that is,
If(check>0), this condition here will see if the value of variable check is greater than 0 or not if its greater than 0 that means our number was divided by another number and gave 0 as a remainder, hence means it is not a prime number and we can print it as the number is “not prime number”.
else, if the value of check is still 0 means our number didn’t got divided by any number as the program didn’t entered our if condition inside our loop. Hence we will print “prime number” .

code:-



```
class q5 {
    public static void main(String[] args) {
    int n=29;
    int check=0;
    for(int i=n-1;i>1;i--){
        if(n%i==0){
            check=check+1;
        }
    }
    if(check>0){
        System.out.println("not prime number");
    }
    else{
        System.out.println("prime number");
    }
}
}
```