

Conditional Statements and LOOPS

CONDITIONAL STATEMENTS

1. If-else

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

    public static void main(String[] args){
        Scanner input=new Scanner(System.in);
        System.out.print("Enter your salary/month:");
        float salary=input.nextFloat();
        if(salary>10000){
            salary+=2000;
        }
        else{
            salary+=1000;
        }
        System.out.println(salary);
    }
}
```

2. If-elseif

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

    public static void main(String[] args){
        Scanner input=new Scanner(System.in);
        System.out.print("Enter your salary/month:");
        float salary=input.nextFloat();
```

```

        if(salary>10000){
            salary+=2000;
        }
        else if(salary>20000){
            salary+=3000;
        }
        else{
            salary+=1000;
        }
        System.out.println(salary);
    }
}

```

LOOPS

- For - number of loops are known

```

//Printing num from 1-5
public class SalaryExample{

    public static void main(String[] args){
        for(int i=1;i<=5;i++)
        {
            System.out.print(i+"\t");
            // System.out.println(i);
        }
    }
}

```

- While : this execute only when the while cond is getting satisfied

```

//Printing num from 1-5
public class SalaryExample{

    public static void main(String[] args){
        int num=1;
        while(num<=5){
            System.out.Print(num);
        }
    }
}

```

```

        num++
    }
}

```

- do while: Loop will be executed atleast once

```

//Printing num from 1-5
public class SalaryExample{

    public static void main(String[] args){
        int num=1;
        do{
            System.out.print(num);
        }while(num<=5);

    }
}

```

Consider,

```

//Printing num from 1-5
public class SalaryExample{

    public static void main(String[] args){
        int num=1;
        do{
            System.out.print(num);
        }while(num!=5);

    }
}

```

Here, 'do' first gets executed and then checks for the cond, again continues to further cont the loop.

Hence, this is how the loop will atleast execute once (here, even if the num is initialised to 1 and cond says num≠)

QUESTION

▼ Q1 : Largest of 3 numbers

```
import java.util.Scanner;
public class LargestAmongNumbers{
    public static void main(String[] args) {
        Scanner input=new Scanner(System.in);
        System.out.print("Enter the first number:");
        int a=input.nextInt();
        System.out.print("Enter the Second number:");
        int b=input.nextInt();
        System.out.print("Enter the Third number:");
        int c=input.nextInt();

        if(a>b && a>c){
            System.out.println("Largest number is:"+a);
        }
        else if(b>a && b>c){
            System.out.println("Largest number is:"+b);
        }
        else{
            System.out.println("Largest number is:"+c);
        }
    }
}
```

Approach 2

```
import java.util.Scanner;
public class LargestAmongNumbers{
    public static void main(String[] args) {
        Scanner input=new Scanner(System.in);
        System.out.print("Enter the first number:");
        int a=input.nextInt();
        System.out.print("Enter the Second number:");
```

```

        int b=input.nextInt();
        System.out.print("Enter the Third number:");
        int c=input.nextInt();

        int max=a;
        if(b>max){
            max=b;
        }

        if(c>max){
            max=c;
        }
        System.out.println("Largest number is :"+max);
    }
}

```

Approach 3

```

import java.util.Scanner;
public class LargestAmongNumbers{
    public static void main(String[] args) {
        Scanner input=new Scanner(System.in);
        System.out.print("Enter the first number:");
        int a=input.nextInt();
        System.out.print("Enter the Second number:");
        int b=input.nextInt();
        System.out.print("Enter the Third number:");
        int c=input.nextInt();

        int max=0;
        if(a<b){
            max=a;
        }
        else{
            max=b;
        }
    }
}

```

```

        if(c>max){
            max=c;
        }

        System.out.println("Largest number is:"+max);
    }
}

```

Approach 4

```

import java.util.Scanner;
public class LargestAmongNumbers{
    public static void main(String[] args) {
        Scanner input=new Scanner(System.in);
        System.out.print("Enter the first number:");
        int a=input.nextInt();
        System.out.print("Enter the Second number:");
        int b=input.nextInt();
        System.out.print("Enter the Third number:");
        int c=input.nextInt();

        // int Max=Math.max(a, b);
        int Max =Math.max(c,Math.max(a,b));
        System.out.println("Largest number is :"+ Max);
    }
}

```

▼ Q2 : Uppercase / Lowercase

```

import java.util.Scanner;

public class Casecheck{
    public static void main(String[] args) {
        Scanner in=new Scanner(System.in);
        System.out.println("Type something");
    }
}

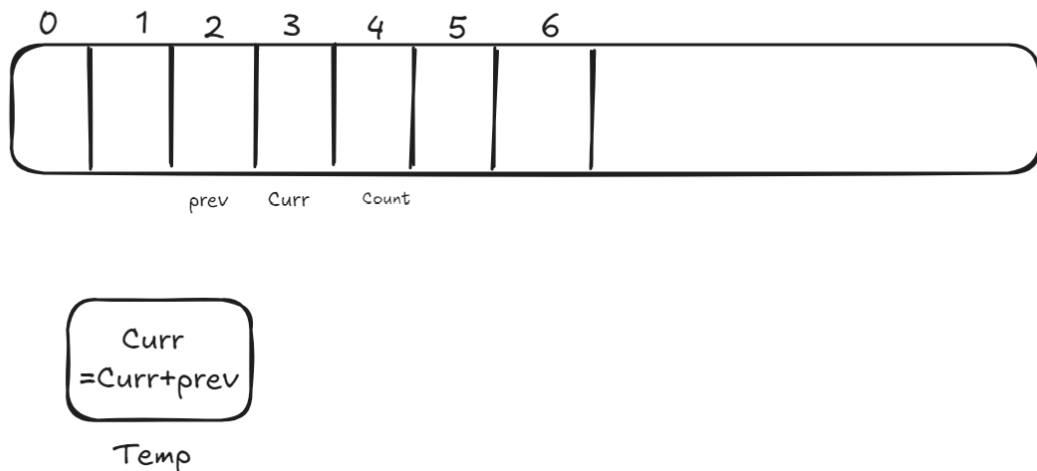
```

```

char ch=in.next().trim().charAt(0);
//trim() --> extra speaces are removed
// CharAt(0)--> charcater at the specified index
if (ch >= 'a' && ch<= 'z'){
    System.out.println("input is in LOWERCASE");
}
else{
    System.out.println("input is in UPPERCASE");
}
}
}

```

▼ Q3 : Find Nth Fibbanocci Number



```

import java.util.Scanner;

public class Fibonacci{
    public static void main(String[] args) {
        Scanner in=new Scanner(System.in);
        System.out.println("Enter the number:");
        int number=in.nextInt();
        int prev=0;
        int curr=1;
    }
}

```

```

        int count=2;

        while(count <= number){
            int temp=curr; //this will be changed, else ev
            curr=curr+prev;
            prev=temp;
            count++;
        }

        System.out.println(curr);
        in.close();
    }
}

```

▼ Q4 : Find the number of times a digit is REPEATED

```

public class RepeatedDigits{
    public static void main(String[] args) {
        int n=45536;

        int count=0;
        while(n>=0){
            int rem=n%10;
            if (rem==5){
                count++
            }
        }
        n=n/10;

    }

}

```

▼ Q5 : Reverse the number

```

public class ReverseNum {
    public static void main(String[] args) {
        int n=1234;
    }
}

```



```

        int ans=0;
        while(n>0){
            int rem=n%10;
            ans=ans*10+rem;
            n=n/10;
        }

        System.out.println(ans);
    }
}

```

▼ Prime Number or not

```

import java.util.Scanner;

public class Prime {
    public static void main(String[] args) {
        Scanner in=new Scanner(System.in);
        System.out.print("Enter a number:");
        int n=in.nextInt();
        if(n%1==0 && n%n==0){
            System.out.println(n+ " " + "is a Prime number"
        }
        else{
            System.out.println(n+ " " + "is not a Prime num
        }

    }

}

```