

//S Harshini-185001058

//1. Write a java program to create a generic stack and perform the operations.

```
import java.util.Scanner;
import java.lang.*;
class Stack <Y>
{
    int maxsize=10;
    int top=-1;
    Y[] array;
    Y i;
    Stack(Y[] a)
    {
        array=a;
    }
    public void display()
    {
        for(int j=0;j<=top;j++)
            System.out.println("E:" +array[j]);
    }
    public boolean isFull()
    {
        if(top+1==maxsize)
            return true;
        else
            return false;
    }
    public boolean isEmpty()
    {
        if(top==-1)
            return true;
        else
            return false;
    }
    public void push(Y element)
    {
        if(isFull())
            System.out.println("the stack is full");
        else
            array[++top]=element;
    }
    public void pop()
    {

```

```

        if(isEmpty())
        {
            System.out.println("the stack is empty");
        }
        else
        {
            i=array[top];
            top--;
            System.out.println("the popped element is "+i);
        }
    }
}

class Generics
{
    public static void main(String arg[])
    {

        int ch,ele,n;
        boolean wh=true;
        Scanner in=new Scanner(System.in);
        Integer[] a=new Integer[6];
        Stack<Integer> s=new Stack<Integer>(a);
        wh=true;
        System.out.println("Integer stack");
        while(wh)
        {
            System.out.println("enter choice 1.push 2.pop 3.display 4.exit");
            ch=in.nextInt();
            switch(ch)
            {
                case 1: System.out.println("Enter element");
                        ele=in.nextInt();
                        s.push(ele);
                        break;
                case 2: s.pop();

                        break;
                case 3: s.display();
                        break;
                case 4: wh=false;
                        break;
            }
        }
    }
}

```

```

        System.out.println("enter strings for stacks");
        wh=true;
        String e;
        String[] sa=new String[6];
        System.out.println("String stack");
        Stack<String> s1=new Stack<String>(sa);
        wh=true;
        while(wh)
        {
            System.out.println("enter choice 1.push 2.pop 3.display 4.exit");
            ch=in.nextInt();
            in.nextLine();
            switch(ch)
            {
                case 1: System.out.println("Enter element");
                        e=in.nextLine();
                        s1.push(e);
                        break;
                case 2: s1.pop();

                        break;
                case 3: s1.display();
                        break;
                case 4: wh=false;
                        break;
            }
        }
    }
}

```

/*SAMPLE INPUT/OUTPUT

cs1058@wtl10:~/Desktop\$ java Generics

Integer stack

enter choice 1.push 2.pop 3.display 4.exit

1

Enter element

1

enter choice 1.push 2.pop 3.display 4.exit

1

Enter element

2

enter choice 1.push 2.pop 3.display 4.exit

```

1
Enter element
3
enter choice 1.push 2.pop 3.display 4.exit
3
E:1
E:2
E:3
enter choice 1.push 2.pop 3.display 4.exit
2
the popped element is 3
enter choice 1.push 2.pop 3.display 4.exit
4
enter strings for stacks
String stack
enter choice 1.push 2.pop 3.display 4.exit
1
Enter element
hii
enter choice 1.push 2.pop 3.display 4.exit
1
Enter element
nice
enter choice 1.push 2.pop 3.display 4.exit
1
Enter element
meeting
enter choice 1.push 2.pop 3.display 4.exit
3
E:hii
E:nice
E:meeting
enter choice 1.push 2.pop 3.display 4.exit
4

*/

```

//2. Write a java program to find the maximum value from the given type of elements using a generic function.

```
import java.util.Scanner;
```

```

import java.lang.*;
class Max
{
    <T extends Comparable> T findMax(T[] a,int n)
    {
        T max=a[0];
        for(int i=0;i<n;i++)
        {
            if(a[i].compareTo(max)>0)
                max=a[i];
        }
        return max;
    }
}
class Gen2
{
    public static void main(String arg[])
    {
        System.out.println("Integer array");
        Scanner in=new Scanner(System.in);
        System.out.println("enter no of elements");
        int n=in.nextInt();
        int ele,j;
        Integer[] array=new Integer[n];
        System.out.println("enter elements ");
        for(j=0;j<n;j++)
        {
            ele=in.nextInt();
            array[j]=ele;
        }
        Max m=new Max();
        System.out.println("the max value is "+m.findMax(array,n));
    }
}

```

```

System.out.println("\nstring array");
System.out.println("enter no of elements");
n=in.nextInt();
in.nextLine();
String e;
String[] arr=new String[n];
System.out.println("enter strings ");
for(j=0;j<n;j++)

```

```

{
    e=in.nextLine();
    arr[j]=e;
}
Max m1=new Max();
System.out.println("the max value is "+m1.findMax(arr,n));}
}

```

/*SAMPLE INPUT/OUTPUT

cs1058@wtl10:~/Desktop\$ java Gen2

Integer array

enter no of elements

5

enter elements

2

4

6

9

0

the max value is 9

string array

enter no of elements

4

enter strings

hii

hiii

hey

good

the max value is hiii

*/

//3. Perform a sorting operation on various types of elements using generic method.

```
import java.util.Scanner;
```

```
import java.lang.*;
```

```
class Max
```

```
{
```

```
    <T extends Comparable> void findMax(T[] a,int n)
```

```
    {
```

```
        T t;
```

```
        for(int i=0;i<n-1;i++)
```

```

        for(int j=i+1;j<n;j++)
        {
            if(a[i].compareTo(a[j])>0)
            {
                t=a[i];
                a[i]=a[j];
                a[j]=t;
            }
        }
        System.out.println("the sorted array is");
        for(int k=0;k<n;k++)
            System.out.println(a[k]);
    }
}

```

```

class Gen2
{
    public static void main(String arg[])
    {
        System.out.println("Integer array");
        Scanner in=new Scanner(System.in);
        System.out.println("enter no of elements");
        int n=in.nextInt();
        int ele,j;
        Integer[] array=new Integer[n];
        System.out.println("enter elements ");
        for(j=0;j<n;j++)
        {
            ele=in.nextInt();
            array[j]=ele;
        }
        Max m=new Max();
        m.findMax(array,n);
    }
}

```

```

System.out.println("\nstring array");
System.out.println("enter no of elements");
n=in.nextInt();
in.nextLine();
String e;
String[] arr=new String[n];
System.out.println("enter strings ");
for(j=0;j<n;j++)
{

```

```
        e=in.nextLine();
        arr[j]=e;
    }
    Max m1=new Max();
    m1.findMax(arr,n);}
}
```

/*SAMPLE INPUT/OUTPUT

C:\Users\Harshini\Desktop>java Gen2

Integer array

enter no of elements

5

enter elements

5

4

3

2

1

the sorted array is

1

2

3

4

5

string array

enter no of elements

5

enter strings

h

hii

hiii

hiii

hello

the sorted array is

h

hello

hii

hiii

hiii

*/