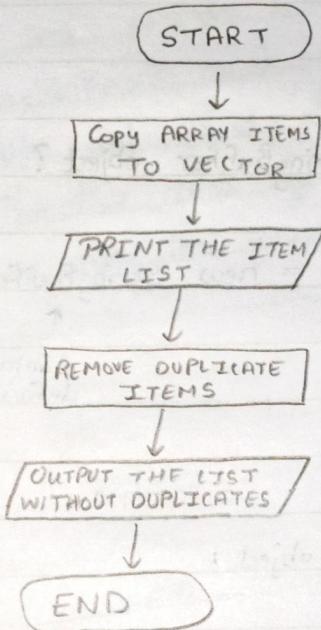


Aim : Create , debug and run java programs based on Wrapper class and Vectors

Flow chart :



code i : VECTOR

Practical No. 05

Aim: Create, debug and run java programs based on wrapper class and vectors.

Theory :

What are wrapper classes?

- • As we know, vectors cannot handle primitive data types like int, float, long, char and double.
- Primitive data types may be converted into object types by using the wrapper classes contained in the java.lang package.

Simple Type	Wrapper class
boolean	Boolean
char	Character
double	Double
float	Float
int	Integer

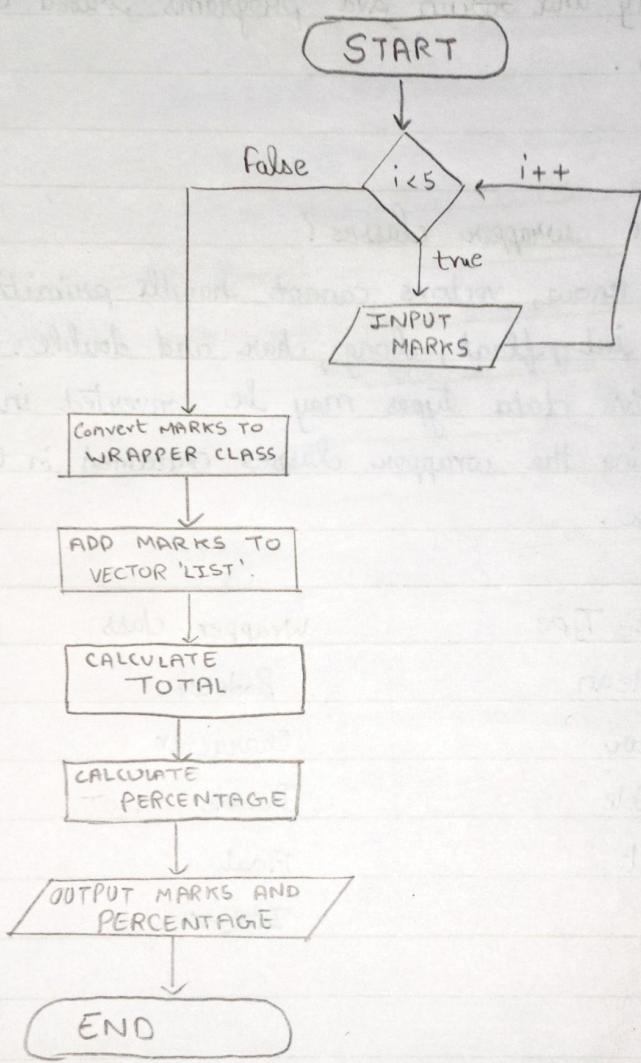
Autoboxing:

- The automatic conversion of primitive data type into its corresponding wrapper class is known as autoboxing.
- Since Java 5, we do not need to use the valueOf() method.

example,

int a = 25;

Integer b = new Integer(a); Integer.valueOf()



Code 2: Wrapper Class

Unboxing:

- Automatic conversion of wrapper class type into its corresponding primitive type is known as unboxing.
- This process is reverse of auto-boxing.
- Example:

```
Integer a = new Integer(35);
```

int b = a; // now, since Java 5, compiler will write
// a.intValue(); internally.

Vectors in Java:

- The Vector class in Java implements a growable array of objects.
- To use Vector class, we need to import, 'java.util.Vector'.
- Vector implements a dynamic array, i.e. it can grow and shrink size according to the need.

Declaration:

```
Vector<e> v = new Vector<e>();
```

ex. Vector<String> v = new Vector<String>();

Vector methods:

- v.addElement(item);
- v.elementAt(index);
- v.size();
- v.removeElement(item);
- v.removeElementAt(index);

: PAGE NO. 4

Conclusion:

Hence, by performing this practical I learnt about the concepts of Wrapper classes and Vector. I also coded, debugged and executed java programs based on the concepts of Wrapper classes and Vectors.

Code:

```
import java.util.Vector;

class Practical5A{

    public static void main(String[] args){

        Vector<String> list = new Vector<String>();
//        String list[] = new String[10];
        String items[] =
{"Bread","Bread","Jam","Butter","Chips","Milk","Butter","Chips","Cookies","Bre
ad","Donuts","Jam"};
        //copying items into the vector 'list'
        for(String a : items){
            list.add(a);
        }

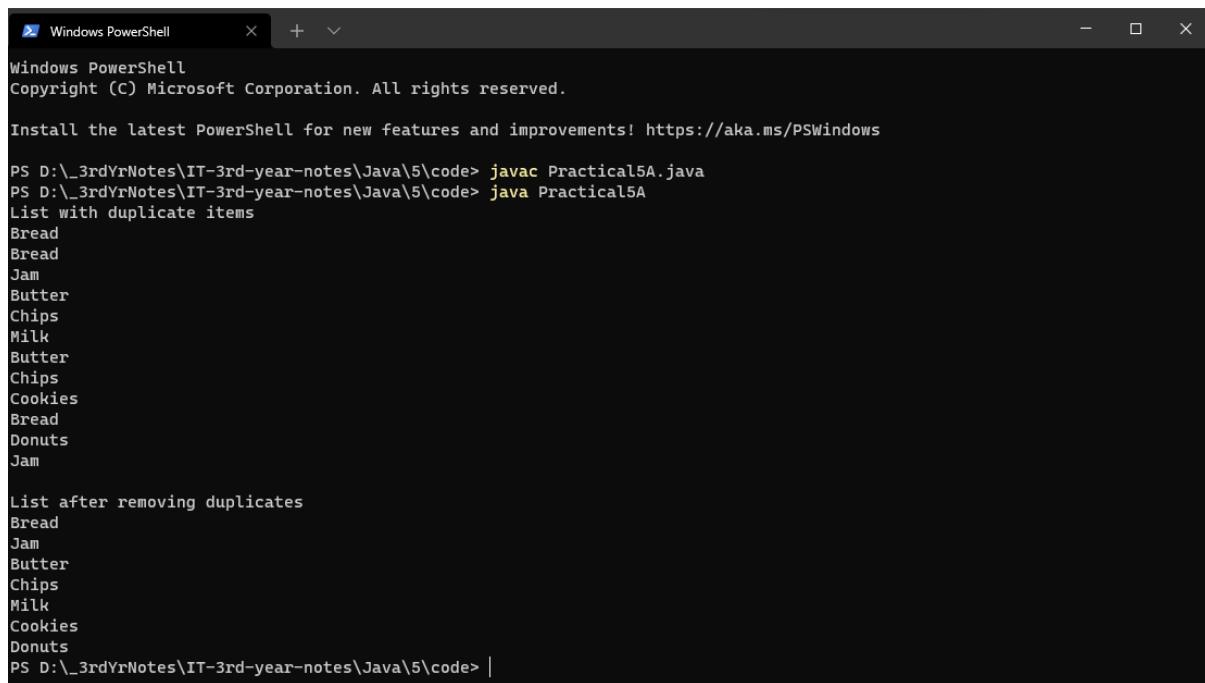
        System.out.println("List with duplicate items");
        for(String a : list){
            System.out.println(a);
        }
        System.out.println();

        for(int i = 0;i<list.size()-1;i++){
            for(int j = i+1;j<list.size();){
                if(list.elementAt(i) == list.elementAt(j)){
                    list.remove(j);
                }else{
                    j++;
                }
            }
        }

        System.out.println("List after removing duplicates");
        for(String a : list){
            System.out.println(a);
        }

    }
}
```

Output:



A screenshot of a Windows PowerShell window titled "Windows PowerShell". The window shows the output of a Java application named "Practical5A". The application lists grocery items with duplicates, then prints a list after removing duplicates.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS D:\_3rdYrNotes\IT-3rd-year-notes\Java\5\code> javac Practical5A.java
PS D:\_3rdYrNotes\IT-3rd-year-notes\Java\5\code> java Practical5A
List with duplicate items
Bread
Bread
Jam
Butter
Chips
Milk
Butter
Chips
Cookies
Bread
Donuts
Jam

List after removing duplicates
Bread
Jam
Butter
Chips
Milk
Cookies
Donuts
PS D:\_3rdYrNotes\IT-3rd-year-notes\Java\5\code> |
```

Code:

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

//@SuppressWarnings("unchecked")
class Practical5B{

    public static void main(String[] args){
        Scanner sc = new Scanner(System.in);
        Vector<Integer> list = new Vector<Integer>();
        int marks;
        for(int i = 0;i < 5 ; i++){
            System.out.print("Enter Marks in subject " + (i+1) + " : ");
            marks = sc.nextInt();
            Integer wrappedMarks = marks;
            list.add(wrappedMarks);
        }

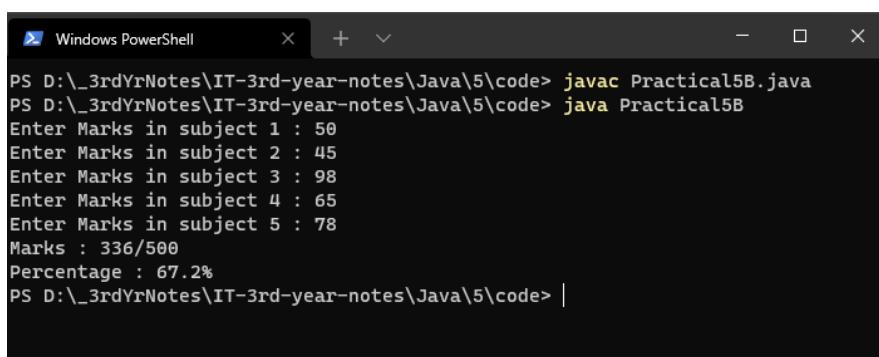
        Integer total = 0;
        Float percent = 0.0f;

        for(Integer a : list){
            total += a;
        }

        percent = total/(float) list.size();

        System.out.println("Marks : " + total + "/" + 500);
        System.out.println("Percentage : " + percent + "%");
    }
}
```

Output:



A screenshot of a Windows PowerShell window titled "Windows PowerShell". The command "javac Practical5B.java" is run, followed by "java Practical5B". The application prompts for five integer inputs: 50, 45, 98, 65, and 78. It then calculates the total marks as 336 and the percentage as 67.2%.

```
PS D:\_3rdYrNotes\IT-3rd-year-notes\Java\5\code> javac Practical5B.java
PS D:\_3rdYrNotes\IT-3rd-year-notes\Java\5\code> java Practical5B
Enter Marks in subject 1 : 50
Enter Marks in subject 2 : 45
Enter Marks in subject 3 : 98
Enter Marks in subject 4 : 65
Enter Marks in subject 5 : 78
Marks : 336/500
Percentage : 67.2%
PS D:\_3rdYrNotes\IT-3rd-year-notes\Java\5\code> |
```

Conclusion:

Hence, by performing this practical I learnt about the concepts of Wrapper classes and Vector. I also coded, debugged and executed java programs based on the concepts of wrapper classes and vectors.