PP Lab Name: PRN:

Practical No 4 **Date:** 14 /09/2023

Title: Write a C++ program using class to process shopping list for a departmental store. The list includes details such as item code no, item name and price of each item and perform the operations like adding, deleting items to the list and printing the total value of an order.

Description:

A class in C++ is the building block, that leads to Object-Oriented programming. It is a user-defined data type, which holds its own data members and member functions, which can be accessed and used by creating an instance of that class.

Classes are like templates that define common properties or attributes. Objects are instances of a class. Once a class has been defined, we can create any number of objects belonging to that class. Each object is associated with the data or type class with which they are created.

Declaration of class:

```
class <class_name>
 {
    data members;
    member functions;
 };
```

Syntax for creating object:

```
classname objectname;
```

Example: fruit mango;

Hint:
Menu:
Add an item and Quantity
Display total value
Delete an item
Display all items
Exit

```
class item
{
int itemcode[50];
float itemprice[50];
int count;
public:
void cntzero();        //initialize count to zero
void getitem();  (hint: cin>>itemcode[count]; count++;)  //assigns value to data members of class item
void displaysum();      //display total value of all items
void remove();          //ask the user to enter item code and delete the specified item
void displayitems();    // displaying items
};
```

PP Lab Name: PRN:

Display all items

Code	Quantity	price	Total
567	**2**	**100**	**200**
678	**5**	**50**	**250**

Grand Total: Rs.450

Total value of inventory: 150

Program Code:

shopping.cpp

```cpp
#include <iostream>
#include<iomanip>
using namespace std;
class shoppinglist
{
    int itemcode[20];
    string itemname[20];
    int itemqty[20];
    int itemprice[20];
    int count;
public:
    void cnt()
    {
        count=0;
    }
    void add();
    void changeqty();
    void totalvalue();
    void allitems();
};
void shoppinglist::add()
{
    cout<<"enter item code:"<<endl;
    cin>>itemcode[count];
    cout<<"enter item name:"<<endl;
    cin>>itemname[count];
    cout<<"enter quantity:"<<endl;
    cin>>itemqty[count];
    cout<<"enter price of item:"<<endl;
    cin>>itemprice[count];
    count++;
    cout<<"\nitem added successfully!!!";
}
```

```cpp
void shoppinglist::changeqty() {
    int c;
    cout << "Enter item code to change quantity: ";
    cin >> c;

    for (int i = 0; i < count; i++) {
        if (itemcode[i] == c) {
            int newqty;
            cout << "Enter new quantity: ";
            cin >> newqty;
            itemqty[i] = newqty;
            cout << "Quantity updated successfully!!!" << endl;
            return;
        }
    }

}
void shoppinglist:: totalvalue()
{
    int sum=0;
    for(int i=0;i<count;i++)
    {
        sum=sum+(itemqty[i]*itemprice[i]);
    }
    cout<<"\nThe total value of all items in the list is "<<sum<<endl;
}
void shoppinglist::allitems()
{
    cout<<"\nDisplaying all items....."<<endl;
    cout<<setw(12)<<"code"<<"|"<<setw(12)<<"name"<<"|"<<setw(12)<<"quantity"<<"|"<<setw(12)<<"price"<<"|"<<endl;;
    for(int i=0;i<count;i++)
    {
        cout<<setw(12)<<itemcode[i]<<"|"<<setw(12)<<itemname[i]<<"|"<<setw(12)<<itemqty[i]<<"|"<<setw(12)<<itemprice[i];
        cout<<"\n";
    }
}
```

PP Lab Name: PRN:

```cpp
int main()
{
    shoppinglist order;
    order.cnt();
    int x;
    while (true)
    {
        cout<<"\nwelcome to shopping list"<<endl;
        cout<<"************************************************"<<endl;
        cout<<"you can perform all the below operations";
        cout<<"\n1.Add an item";
        cout<<"\n2.Increase/Decrease quantity";
        cout<<"\n3.Display total value of all items in the list";
        cout<<"\n4.Display list";
        cout<<"\n5.Exit";
        cout<<"\nEnter an option:"<<endl;
        cin>>x;

        switch(x)
        {
            case 1:
                order.add();
                break;
            case 2:
                order.changeqty();
                break;
            case 3:
                order.totalvalue();
                break;
            case 4:
                order.allitems();
                break;
            case 5:
                cout<<"Thank you for visiting!!!"<<endl;
                return 0;
            default:
                cout<<"Please enter a valid option";
        }
    }
}
```

Input and Output

```
C:\Program Files (x86)\Dev-C

welcome to shopping list
************************************************
you can perform all the below operations
1.Add an item
2.Increase/Decrease quantity
3.Display total value of all items in the list
4.Display list
5.Exit
Enter an option:
1
enter item code:
1
enter item name:
chips
enter quantity:
2
enter price of item:
5

item added successfully!!!
welcome to shopping list
************************************************
you can perform all the below operations
1.Add an item
2.Increase/Decrease quantity
3.Display total value of all items in the list
4.Display list
5.Exit
Enter an option:
1
enter item code:
2
enter item name:
soap
enter quantity:
4
enter price of item:
20

item added successfully!!!
welcome to shopping list
************************************************
you can perform all the below operations
1.Add an item
2.Increase/Decrease quantity
3.Display total value of all items in the list
4.Display list
5.Exit
Enter an option:
1
```

```
enter price of item:
2

item added successfully!!!
welcome to shopping list
************************************************
you can perform all the below operations
1.Add an item
2.Increase/Decrease quantity
3.Display total value of all items in the list
4.Display list
5.Exit
Enter an option:
2
Enter item code to change quantity: 3
Enter new quantity: 15
Quantity updated successfully!!!

welcome to shopping list
************************************************
you can perform all the below operations
1.Add an item
2.Increase/Decrease quantity
3.Display total value of all items in the list
4.Display list
5.Exit
Enter an option:
3

The total value of all items in the list is 120

welcome to shopping list
************************************************
you can perform all the below operations
1.Add an item
2.Increase/Decrease quantity
3.Display total value of all items in the list
4.Display list
5.Exit
Enter an option:
4

Displaying all items.....
        code|        name|    quantity|       price|
           1|       chips|           2|           5
           2|        soap|           4|          20
           3|      coffee|          15|           2

the grand total is : 120
```

```
welcome to shopping list
*********************************************
you can perform all the below operations
1.Add an item
2.Increase/Decrease quantity
3.Display total value of all items in the list
4.Display list
5.Exit
Enter an option:
5
Thank you for visiting!!!

--------------------------------
Process exited with return value 0
Press any key to continue . . .
```

Conclusion: Thus we have implemented the concept of classes in C++

Practice programs: Write a program that would print the information (name, year of joining, salary, address) of three employees by creating a class named 'Employee'. The output should be as follows:

Name	Year of joining	Address
Rahul	2010	64C- WallsStreat
Samar	2000	68D- WallsStreat
Ishita	2018	26B- WallsStreat

CODE

main.cpp Run

```cpp
#include <iostream>
#include <iomanip>
#include <string>
using namespace std;

class Employee {
private:
    string name;
    int joining;
    string address;

public:

    void display() {
        cout << left << setw(10) << name << setw(6) << joining << setw(20)
             << address << endl;
    }


    void empname(string n) {
        name = n;
    }

    void empjoin(int year) {
        joining = year;
    }

    void empaddress(string addr) {
        address = addr;
    }
```

main.cpp Run

```
31
32  int main() {
33
34      Employee emp1;
35      Employee emp2;
36      Employee emp3;
37
38
39      emp1.empname("Rahul");
40      emp1.empjoin(2010);
41      emp1.empaddress("64C- WallsStreat");
42
43      emp2.empname("Samar");
44      emp2.empjoin(2000);
45      emp2.empaddress("68D- WallsStreat");
46
47      emp3.empname("Ishita");
48      emp3.empjoin(2018);
49      emp3.empaddress("26B- WallsStreat");
50
51
52      cout << left << setw(10) << "Name" << setw(6) << "Year" << setw(20) <<
            "Address" << endl;
53      emp1.display();
54      emp2.display();
55      emp3.display();
56
57      return 0;
58  }
59
```

OUTPUT

```
/tmp/PWZdMbjLEr.o
Name      Year  Address
Rahul     2010  64C- WallsStreat
Samar     2000  68D- WallsStreat
Ishita    2018  26B- WallsStreat
```