Jaypee University of Engineering and Technology, Guna

Department of Computer Science and Engineering

Object Oriented Programming Lab
(18B17CI271)

Lab Exercise-7

Name : Palash Mishra

Enroll no. : 201B172

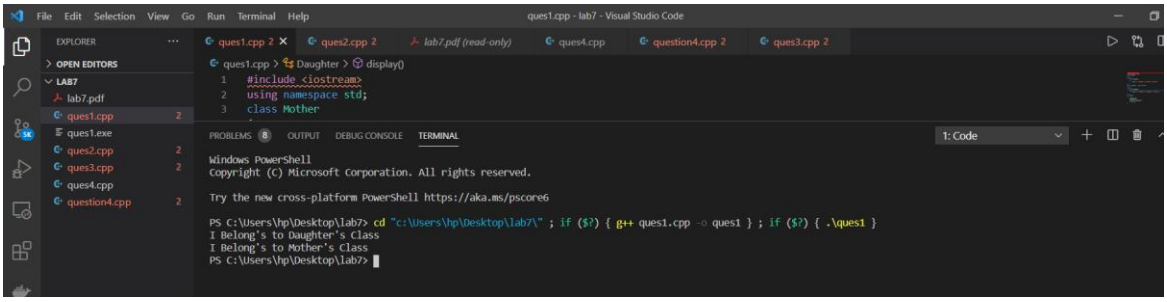
Question 1.

Write a program with a mother class and a derived daughterclass. Bothof them should have a method void display ()that prints a message (different for mother and daughter).In the main function declare an object of class daughter and call the display() method on it.Also suitably invoke the display() function of mother class using this object of class daughter.

Answer :

```
#include <iostream>
using namespace std;
class Mother
{
public:
    void display()
    {
        cout << "I Belong's to Mother's Class\n";
    }
};
class Daughter : public Mother
{
public:
    void display()
    {
        cout << "I Belong's to Daughter's Class\n";
    }
};
int main()
{
    Daughter D;
    D.display();
    D.Mother::display();
    return 0;
}
```

Output:



Question 2.

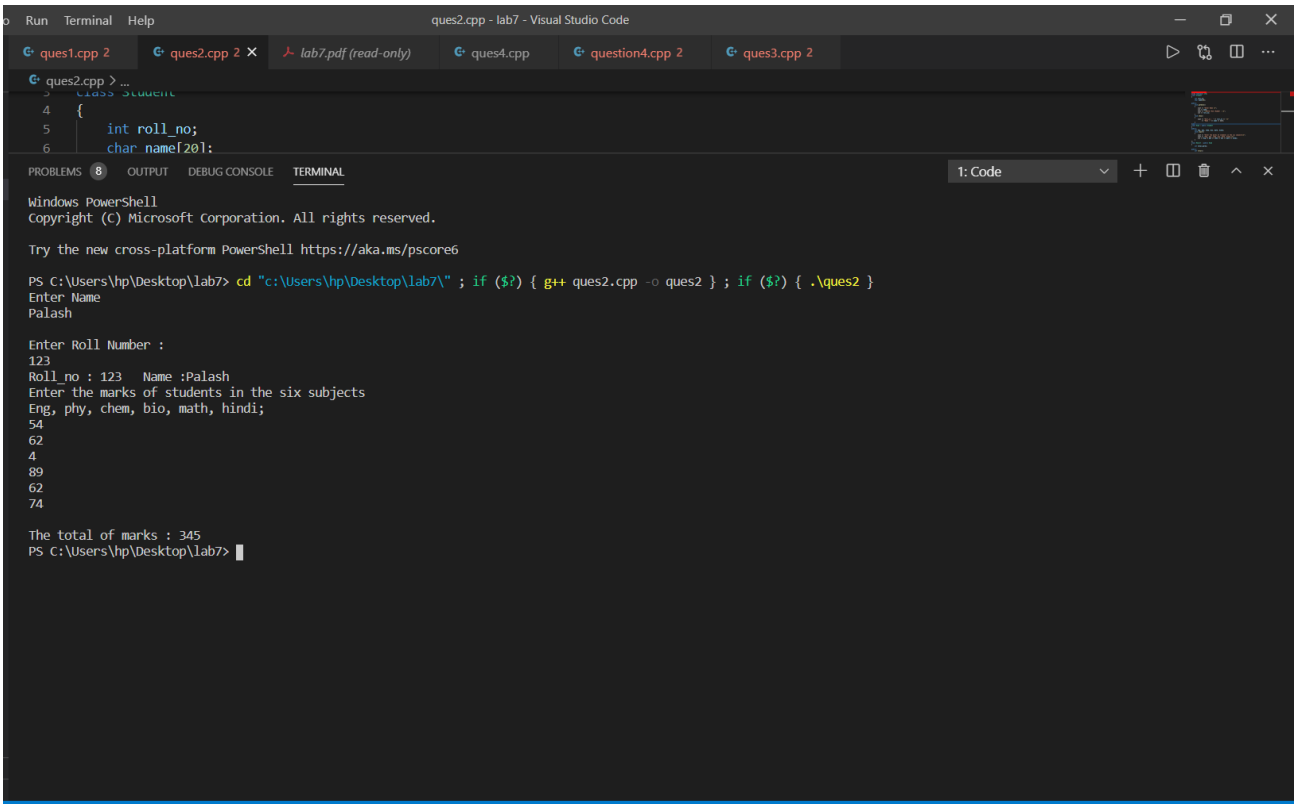
Consider an example of declaring the examination result. Design three classes: Student, Exam, and Result. The Student class has data members representing roll number, name.Create the class Exam by inheriting Student class. The Exam class adds fields(data members)representing the marks scored in six subjects. Derive the Result from the Exam class, and it has its own fields such as total_marks. Write an interactive program to model this relationship.

Answer :

```
#include <iostream>
using namespace std;
class Student
{
    int roll_no;
    char name[20];

public:
    void getdata()
    {
        cout << "Enter Name \n";
        cin >> name;
        cout << "\nEnter Roll Number : \n";
        cin >> roll_no;
    }
    void show()
    {
        cout << "roll_no : " << roll_no << "\t"
             << "Name :" << name << endl;
    }
};
class Exam : public Student
{
public:
    int Eng, phy, chem, bio, math, hindi;
    void input()
    {
        cout << "Enter the marks of students in the six subjects\n";
        cout << "Eng, phy, chem, bio, math, hindi;\n";
        cin >> Eng >> phy >> chem >> bio >> math >> hindi;
    }
};
class Result : public Exam
{
```

```
    int total_marks;

public:
    int total()
    {

        total_marks = Eng + phy + chem + bio + math + hindi;
        return total_marks;
    }
    void display()
    {
        cout << "\nThe total of marks : " << total();
    }
};
int main()
{
    Result R;
    R.getdata();
    R.show();
    R.input();
    R.display();
}
```

Output:



Question 3.

There is a class student, that stores name of school or university from which he is enrolled and name of highest degree he has obtained so far. It has the function to get and display the members. Design a class Employee with name and employee number. Derive Manager, Scientist and Laborer classesfrom Employee class. The manager class has extra attribute title(string type)and dues(float type). The scientist class has extra attributes number of publications. The Laborer class has nothing extra. The classes have necessary functions for set and display the information.The manager and scientist are studentsof a university also. Use inheritance.Test your program by creating objects of type manager, scientists and laborer.

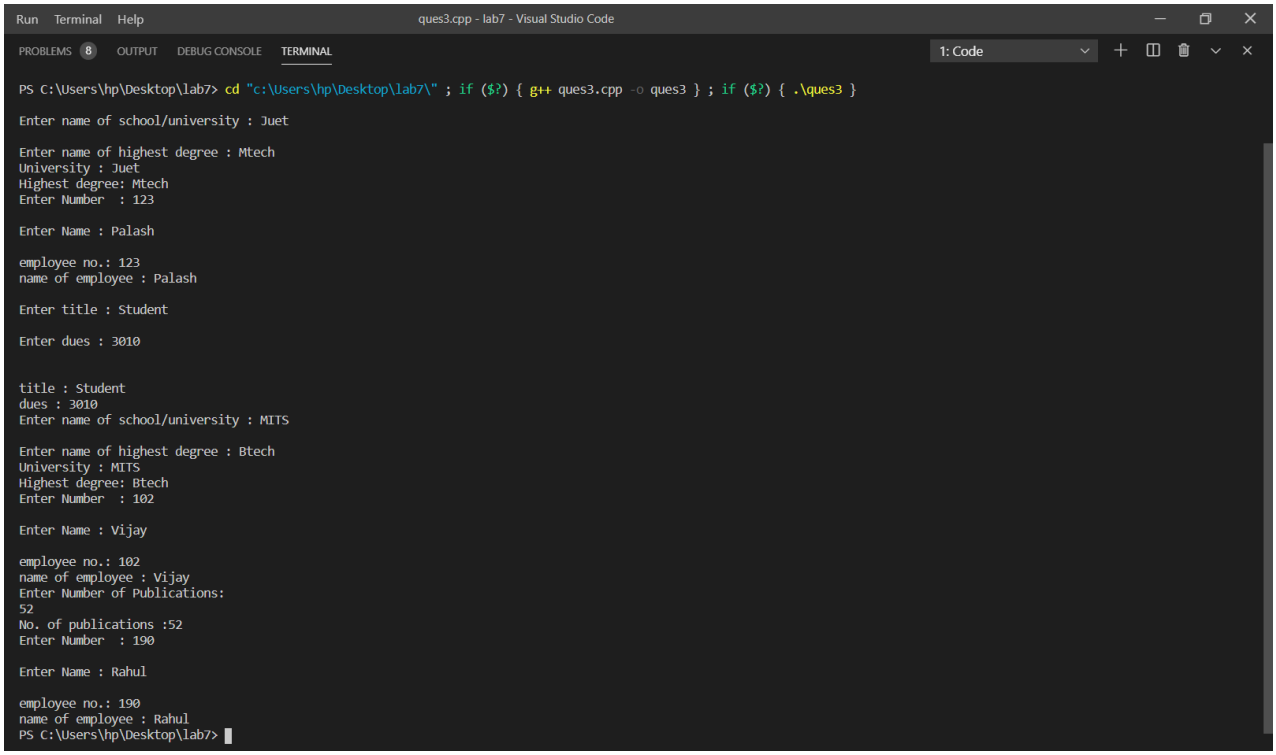
Answer :

```
#include <iostream>
using namespace std;
class student
{
    char n_of_s[40];
    char degree[30];

public:
    void gdata()
    {
        cout << "\nEnter name of school/university : ";
        cin >> n_of_s;
        cout << "\nEnter name of highest degree : ";
        cin >> degree;
    }
    void sdata()
    {
        cout << "University : " << n_of_s << endl;
        cout << "Highest degree: " << degree << endl;
    }
};

class employee
{
    int no;
    char name[20];

public:
    void getdata()
```

```
    {
        cout << "Enter Number  : ";
        cin >> no;
        cout << "\nEnter Name : ";
        cin >> name;
        cout << endl;
    }
    void showData()
    {
        cout << "employee no.: " << no << endl;
        cout << "name of employee : " << name << endl;
    }
};
class manager : public employee,public student
{
    char title[40];
    float dues;

public:
    void get()
    {
        cout << "\nEnter title : ";
        cin >> title;
        cout << "\nEnter dues : ";
        cin >> dues;
        cout << endl;
    }
    void show()
    {
        cout << "\ntitle : " << title;
        cout << "\ndues : " << dues;

    }
};
class scientist : public employee,public student
{
    int n_of_publ;

public:
    void enter()
    {
        cout << "\nEnter Number of Publications: ";
        cin >> n_of_publ;
    }
    void display()
    {
        cout << "\nNo. of publications :" << n_of_publ;
    }
```

```cpp
};
class laborer : public employee
{
};
int main()
{
    manager mn;
    mn.gdata();
    mn.sdata();
    mn.getdata();
    mn.showData();
    mn.get();
    mn.show();
    scientist st;
    st.gdata();
    st.sdata();
    st.getdata();
    st.showData();
    st.enter();
    st.display();
    laborer lb;
    lb.getdata();
    lb.showData();
    return 0;
}
```

Output:



Question 4.

An educational institution wishes to maintain a database of its employees. The database is divided into a number of classes whose hierarchical relationships are shown in Fig.1. The figure also shows the minimum information required for each class. Specify all the classes and define methods to create the database and retrieve individual information as and when required.

Answer :

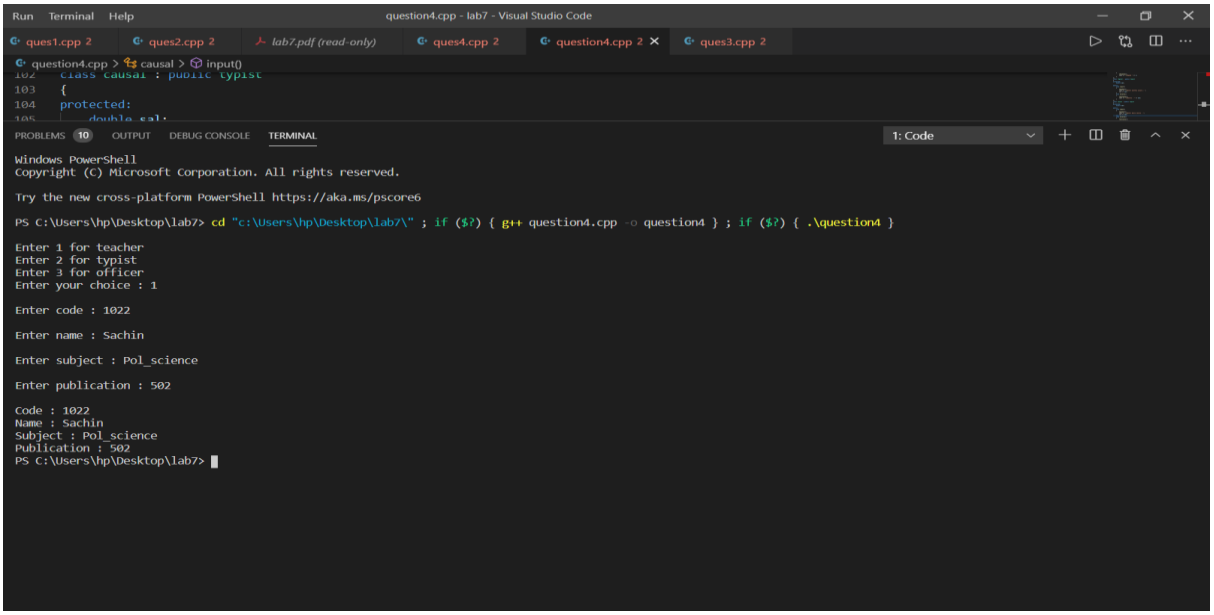
```
#include <iostream>
using namespace std;
class staff
{
protected:
    int code;
    string name;

public:
    void sinput()
    {
        cout << "\nEnter code : ";
        cin >> code;
        cout << "\nEnter name : ";
        cin >> name;
    }
    void idisplay()
    {

        cout
            << "\nCode : " << code;
        cout << "\nName : " << name;
    }
};
class teacher : public staff
{
protected:
    string sub;
    string pub;

public:
    void tinput()
    {
        sinput();
        cout << "\nEnter subject : ";
        cin >> sub;
```

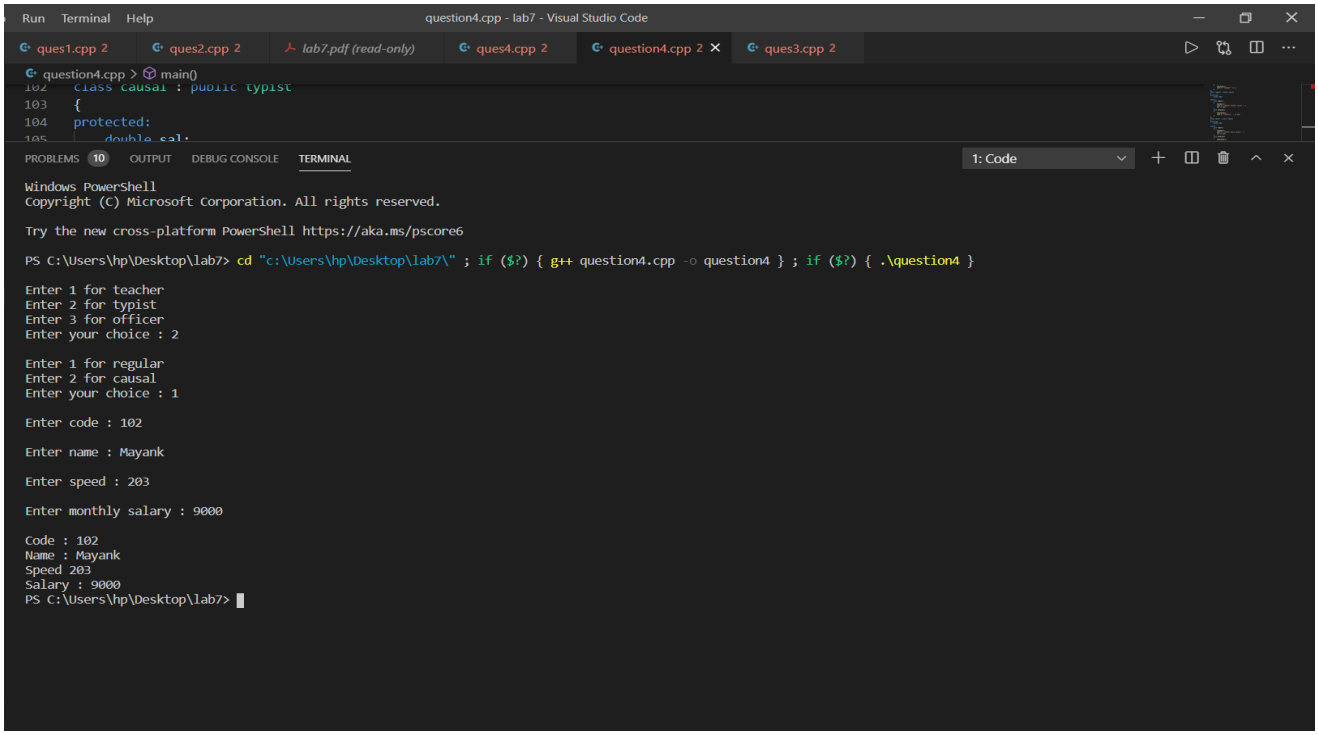
```
        cout << "\nEnter publication : ";
        cin >> pub;
    }
    void tdisplay()
    {
        idisplay();
        cout << "\nSubject : " << sub;
        cout << "\nPublication : " << pub;
    }
};
class officer : public staff
{
protected:
    string g;

public:
    void oinput()
    {
        sinput();
        cout << "\nEnter grade : ";
        cin >> g;
    }
    void odisplay()
    {

        idisplay();
        cout << "\nGrade : " << g;
    }
};
class typist : public staff
{
protected:
    double s;

public:
    void tpinput()
    {
        sinput();
        cout << "\nEnter speed : ";
        cin >> s;
    }
    void tydisplay()
    {
        idisplay();
        cout << "\nSpeed " << s;
    }
};
class regular : public typist
```

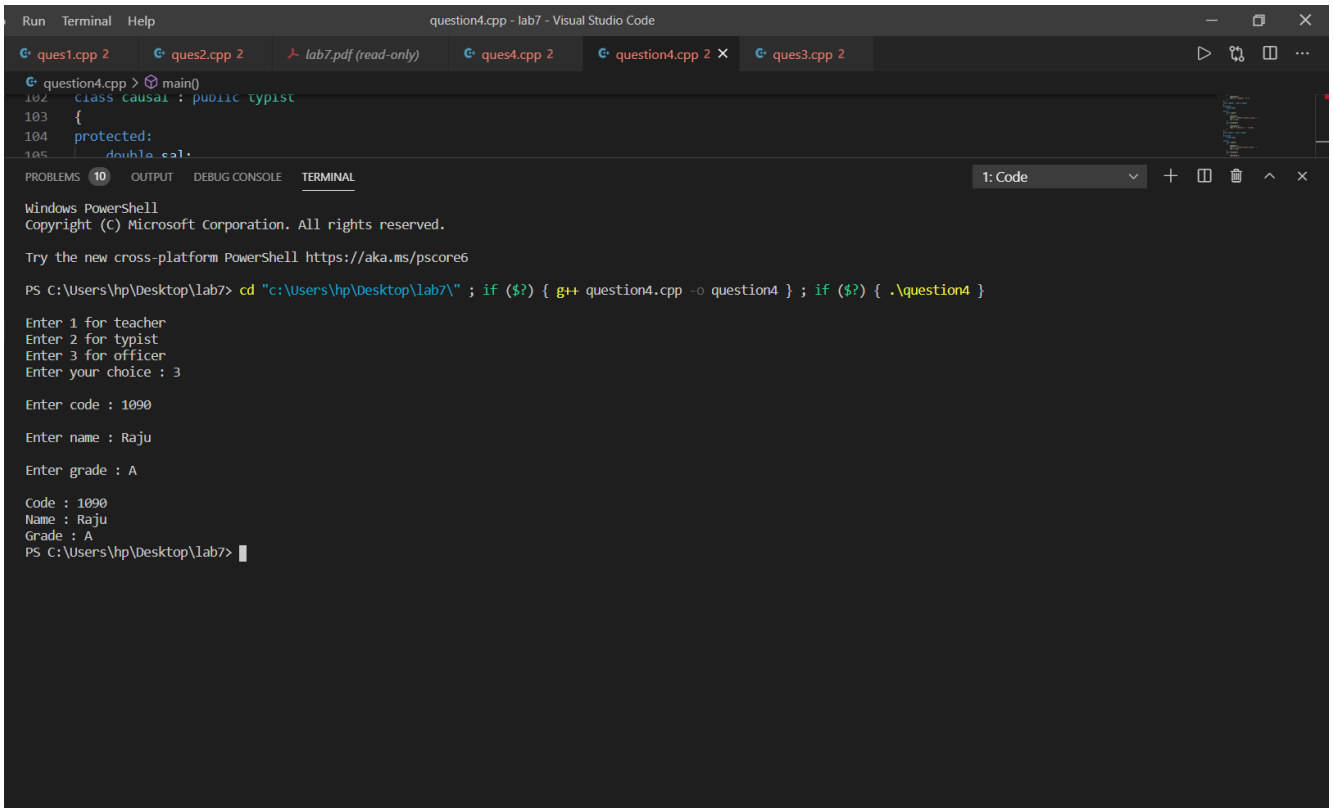
```
{
protected:
    double sal;

public:
    void input()
    {
        tpinput();
        cout << "\nEnter monthly salary : ";
        cin >> sal;
    }
    void display()
    {
        tydisplay();
        cout << "\nSalary : " << sal;
    }
};
class causal : public typist
{
protected:
    double sal;

public:
    void input()
    {
        tpinput();
        cout << "\nEnter daily salary : ";
        cin >> sal;
    }
    void display()
    {
        tydisplay();
        cout << "\nSalary : " << sal;
    }
};
int main()
{
    int ch, ch2;
    cout << "\nEnter 1 for teacher";
    cout << "\nEnter 2 for typist";
    cout << "\nEnter 3 for officer";
    cout << "\nEnter your choice : ";
    cin >> ch;
    if (ch == 1)
    {
        teacher t;
        t.tinput();
        t.tdisplay();
```

```
    }
    else if (ch == 3)
    {
        officer o;
        o.oinput();
        o.odisplay();
    }
    else if (ch == 2)

    {
        cout << "\nEnter 1 for regular";
        cout << "\nEnter 2 for causal";
        cout << "\nEnter your choice : ";
        cin >> ch2;
        if (ch2 == 1)
        {
            regular r;
            r.input();
            r.display();
        }
        else if (ch2 == 2)
        {
            causal c;
            c.input();
            c.display();
        }
    }
    return 0;
}
```

Output:




Run Terminal Help
question4.cpp - lab7 - Visual Studio Code
ques1.cpp 2
ques2.cpp 2
lab7.pdf (read-only)
ques4.cpp 2
question4.cpp 2
ques3.cpp 2
question4.cpp > main()
class causal : public typist
{
protected:
double sal;
PROBLEMS 10
OUTPUT
DEBUG CONSOLE
TERMINAL
1: Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\hp\Desktop\lab7> cd "c:\Users\hp\Desktop\lab7\" ; if ($?) { g++ question4.cpp -o question4 } ; if ($?) { .\question4 }
Enter 1 for teacher
Enter 2 for typist
Enter 3 for officer
Enter your choice : 2
Enter 1 for regular
Enter 2 for causal
Enter your choice : 1
Enter code : 102
Enter name : Mayank
Enter speed : 203
Enter monthly salary : 9000
Code : 102
Name : Mayank
Speed 203
Salary : 9000
PS C:\Users\hp\Desktop\lab7>


Run Terminal Help
question4.cpp - lab7 - Visual Studio Code
ques1.cpp 2
ques2.cpp 2
lab7.pdf (read-only)
ques4.cpp 2
question4.cpp 2
ques3.cpp 2
question4.cpp > main()
class causal : public typist
{
protected:
double sal;
PROBLEMS 10
OUTPUT
DEBUG CONSOLE
TERMINAL
1: Code
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.
Try the new cross-platform PowerShell https://aka.ms/pscore6
PS C:\Users\hp\Desktop\lab7> cd "c:\Users\hp\Desktop\lab7\" ; if ($?) { g++ question4.cpp -o question4 } ; if ($?) { .\question4 }
Enter 1 for teacher
Enter 2 for typist
Enter 3 for officer
Enter your choice : 3
Enter code : 1090
Enter name : Raju
Enter grade : A
Code : 1090
Name : Raju
Grade : A
PS C:\Users\hp\Desktop\lab7>