**Name:** Muhammad Kaleem Ullah **Reg. Number:** FA19-BCE-007

**In Lab 08**

**Class Hierarchies (Multi-Level & Multiple Inheritance)**

* 1. Create a class Person having *name, age* and *gender* as its data members. Create another class Department which has *DepartmentName* and *ProgramName* as it data members. Derive a class Student from class Person and class Department which has student *ID*,*grade* and *number of courses* as its member variables.
     1. Write set and get functions to enter and display the data members.
     2. Write main function to implement these classes. Enter the student data to show multiple inheritance.

**Solution:**

* **Code:**

#include<iostream>

using namespace std;

class Person

{

private:

char name[100];

char gender;

int age;

public:

void getData()

{

cout<<"Enter the Name Of the Student : ";

cin>>name;

cout<<endl<<"Enter the Gender (Male(M) & Female(F)) : ";

cin>>gender;

cout<<endl<<"Enter the age : ";

cin>>age;

}

void putData()const

{

cout<<endl<<"Name : "<<name;

cout<<endl<<"Gender :"<<gender;

cout<<endl<<"Age : "<<age;

}

};

class Department:public Person

{

private:

char DepartmentName[100],ProgramName[100];

public:

void getData()

{

cout<<endl<<"Enter the Name of the Department : ";

cin>>DepartmentName;

cout<<endl<<"Enter the Program Name : ";

cin>>ProgramName;

}

void putData()const

{

cout<<endl<<"Department :"<<DepartmentName;

cout<<endl<<"Program Name : "<<ProgramName;

}

};

class Student:public Department

{

private:

int studentID,number;

char grade;

public:

void getData()

{

Person::getData();

Department::getData();

cout<<endl<<"Enter the Student Id : ";

cin>>studentID;

cout<<endl<<"Enter the Grade (A,B,C,D or F) : ";

cin>>grade;

cout<<endl<<"Enter the Number : ";

cin>>number;

}

void putData()const

{

Person::putData();

Department::putData();

cout<<endl<<"Student ID :"<<studentID;

cout<<endl<<"Grade : "<<grade;

cout<<endl<<"Number : "<<number;

}

};

int main()

{

Student s;

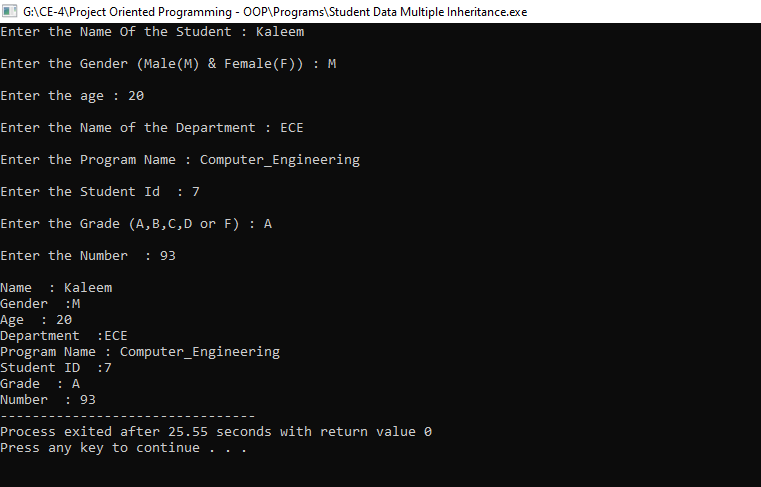
s.getData();

s.putData();

return 0;

}

* **Output:**



* 1. Design a class named Employee. The class should keep the following information in

•Employee name

•Employee number

•Hire date

Write one or more constructors and the appropriate accessor and mutator functions for the class. Next, write a class named ProductionWorker that is derived from the Employee class. The ProductionWorker class should have member variables to hold the following information:

•Shift (an integer)

•Hourly pay rate (a double )

The workday is divided into two shifts: day and night. The shift variable will hold an integer value representing the shift that the employee works. The day shift is shift 1, and the night shift is shift 2. For night shift hourly rate will be doubled, write down a function called salary to calculate the total salary of worker. Write one or more constructors and the appropriate accessor and mutator functions for the class. Demonstrate the classes by writing a program that uses a ProductionWorker object, ask the user for how many workers he wants to store the data and then display the recorded data.

**Solution:**

* **Code:**

#include<iostream>

#include<string>

using namespace std;

class Employee

{

protected:

string eName;

int eNumber,date,month,year;

public:

void getData()

{

cin.ignore();

cout<<"\nEnter the Employee Name : "; getline(cin,eName,'\n');

cout<<"\nEnter the Employee Number : ";cin>>eNumber;

}

void putData()const

{

cout<<"\nEmployee Name : "<<eName;

cout<<"\nEmployee Number : "<<eNumber;

}

};

class hireDate:public Employee

{

public:

void getData()

{

cout<<"Enter Your Hiring Date >> ";

cout<<"\nEnter the Data of Month : ";cin>>date;

cout<<"\nEnter the Month : ";cin>>month;

cout<<"\nEnter the Year : ";cin>>year;

}

void putData()const

{

cout<<"\nHiring Date is : "<<date<<"\\"<<month<<"\\"<<year;

}

};

class ProductionWorker:public Employee

{

private:

int shift,pay,hours;

double hourPerRate;

hireDate hd;

public:

ProductionWorker()

{

shift=0;

hourPerRate=0;

}

void getData()

{

Employee::getData();

hd.getData();

cout<<"\nFor day shift Enter '1'";

cout<<"\nFor night shift Enter '2' ";

cout<<"\nEnter the Shift : ";cin>>shift;

cout<<"\nEnter the Hour Per Rate : ";cin>>hourPerRate;

cout<<"\nEnter the hours : ";cin>>hours;

}

void convertInToText()

{

if (shift == 1)

{

cout<<"\nDay Shift";

}

else if(shift ==2)

{

cout<<"\nNight Shift";

}

else

{

cout<<"\nInvalid Choice For Shift !!";

}

}

void putData()

{

Employee::putData();

hd.putData();

convertInToText();

cout<<"\nHourly pay rate : "<<hourPerRate;

salary();

}

void salary()

{

if (shift == 1)

{

pay=hours\*hourPerRate;

cout<<"\nSalary : "<<pay;

}

else if(shift ==2)

{

pay=hours\*hourPerRate\*hourPerRate;

cout<<"\nSalary : "<<pay;

}

else

{

cout<<"\nInvalid Choice For Shift !!";

}

}

};

int main()

{

int n;

cout<<"Enter the Number Of Employee You Want To Enter : ";cin>>n;

ProductionWorker pw[n];

cout<<"Getting Data of Employee >> ";

for(int i=0;i<n;i++)

{

pw[i].getData();

}

cout<<"Stored Data of Employee >> ";

for(int i=0;i<n;i++)

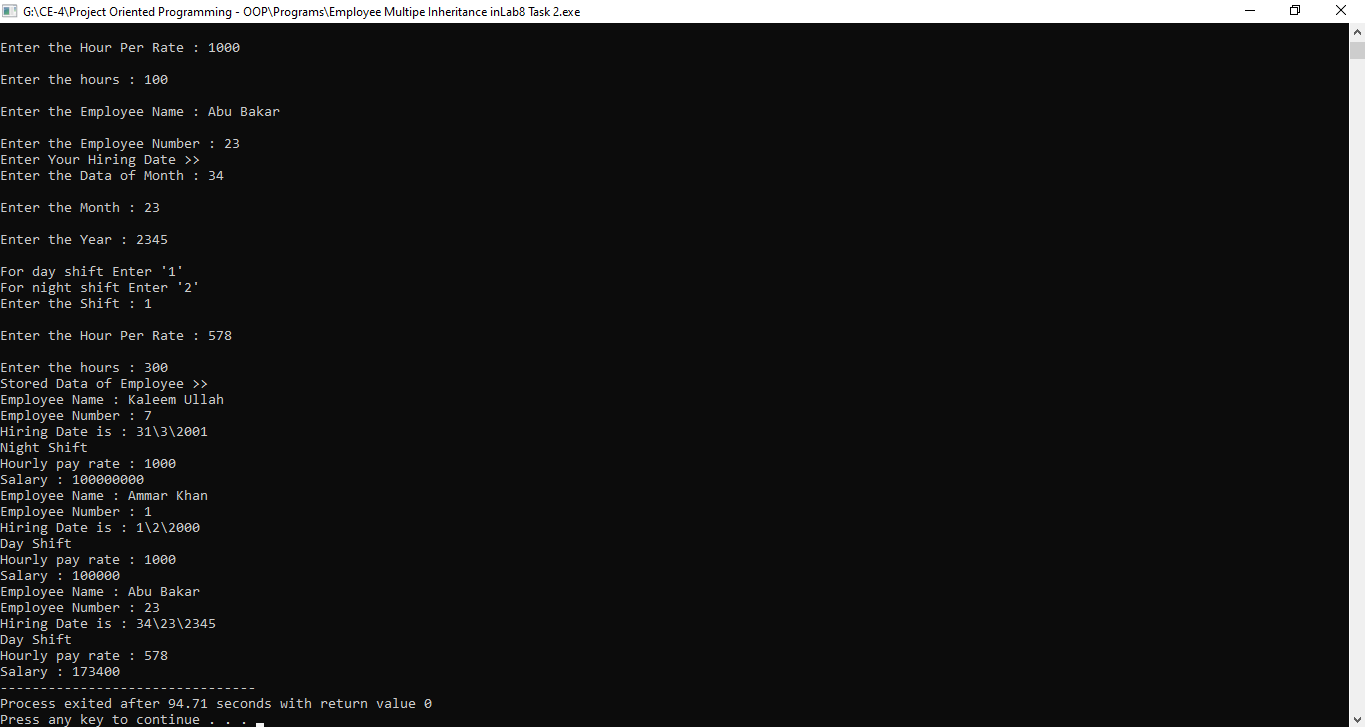
{

pw[i].putData();

}

}

* **Output:**



* 1. **Consider the following classes**

Create a class Date having day, month & year as its data members**.** Create another class called Time with its data members as hours, minutes & seconds. Write down the following functions for both classes:

1. void display(); // to displays the data
2. get() function // to accesses the data members
3. void set(); // to sets the values of data members

Define a class DateandTime from above two classes which displays both date and time.

1. Define an instance object of class DateTime called Watch.
2. Write a main () function that would initialize the values through the constructor functions, and then allows them to be reset through the set () functions. Be sure and display the results following the constructor before you use the set functions.
3. Through the use of the display () function, the time and date are to be displayed. Note that the display () functions in all three classes need to be defined, as well as the constructor and all the access functions.

**Solution:**

* **Code:**

#include<iostream>

using namespace std;

class Date

{

private:

int date,month,year;

public:

Date()

{

date=month=year=0;

}

void set()

{

cout<<"\nEnter Your Date >> ";

cout<<"\nEnter the Data of Month : ";cin>>date;

cout<<"\nEnter the Month : ";cin>>month;

cout<<"\nEnter the Year : ";cin>>year;

}

void Display()const

{

cout<<"\n Date : "<<date<<"\\"<<month<<"\\"<<year;

}

};

class Time

{

private:

int hours,minutes,seconds;

public:

Time()

{

hours=minutes=seconds=0;

}

void set()

{

cout<<"\nEnter the Hours : ";cin>>hours;

cout<<"\nEnter the Minutes : ";cin>>minutes;

cout<<"\nEnter the Seconds : ";cin>>seconds;

}

void Display()const

{

cout<<"\nTime : ";

cout<<hours<<":"<<minutes<<":"<<seconds;

}

};

class DateandTime:public Date,public Time

{

public:

void set()

{

Date::set();

Time::set();

}

void Display()const

{

Date::Display();

Time::Display();

}

};

int main()

{

DateandTime Watch;

Watch.Display();

DateandTime Watch1;

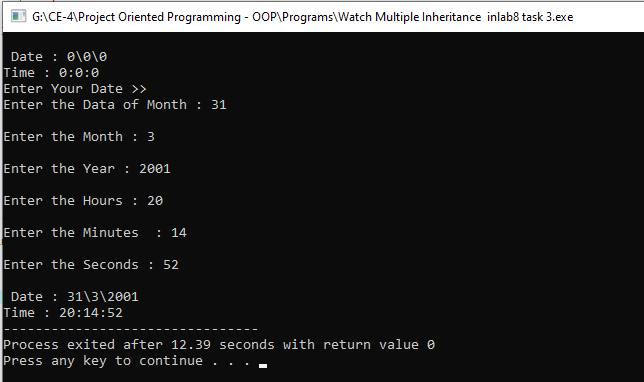
Watch1.set();

Watch1.Display();

return 0;

}

* **Output:**

****