**ADVANCED ASSIGNMENT - 3**

Name: **HITARTH GANATRA**

Sem: **II**

Div.: **1**

ID: **22BCA064**

Subject: **Fundamentals of Object Oriented Programming Language**

Q1

**CODE:**

*// Write a program to input Student Name, Student ID, Branch and Semester for Class Student and display the details of 3 students.*

#include<iostream>

using namespace std;

class Student{

    string name, ID, Branch;

    int sem;

    public:

        void getDetails(){

            cout << "Enter Name: ";

            cin >> name;

            cout << "Enter ID: ";

            fflush(stdin);

            cin >> ID;

            cout << "Enter Branch: ";

            cin >> Branch;

            cout << "Enter Semester: ";

            cin >> sem;

        }

        void displayDetails(){

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

            cout << "ID: " << ID << endl;

            cout << "Branch: " << Branch << endl;

            cout << "Semester: " << sem << endl<< endl;

        }

};

int main(){

    Student s[3];

    for(int i=0; i<3; i++){

        cout << "Enter details of Student " << i+1 << endl;

        s[i].getDetails();

    }

    for(int i=0; i<3; i++){

        s[i].displayDetails();

    }

    return 0;

}

Q2

**CODE:**

*// Write a program to input Item Name, Lot number, Quantity and Price for Class Item and display the details. (Use array of objects)*

#include<iostream>

using namespace std;

class Item{

    string name;

    int lotNo, qty, price;

    public:

        void getDetails(){

            cout << "Enter Item Name: ";

            cin >> name;

            cout << "Enter Lot Number: ";

            cin >> lotNo;

            cout << "Enter Quantity: ";

            cin >> qty;

            cout << "Enter Price: ";

            cin >> price;

        }

        void displayDetails(){

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

            cout << "Lot Number: " << lotNo << endl;

            cout << "Quantity: " << qty << endl;

            cout << "Price: " << price << endl<< endl;

        }

};

int main(){

    Item i[3];

    for(int j=0; j<3; j++){

        cout << "Enter details of Item " << j+1 << endl;

        i[j].getDetails();

    }

    for(int j=0; j<3; j++){

        i[j].displayDetails();

    }

    return 0;

}

Q3

**CODE:**

*// Write a program to take one number from user using a member function and display the number using a friend function.*

#include<iostream>

using namespace std;

class Sample{

    int num;

    public:

        void getNum(){

            cout << "Enter a number: ";

            cin >> num;

        }

        friend void displayNum(Sample);

};

void displayNum(Sample s){

    cout << "Number: " << s.num << endl;

}

int main(){

    Sample s1;

    s1.getNum();

    displayNum(s1);

    return 0;

}

Q4

**CODE:**

*// Write a program to input price of two products in a member function and display total bill amount in a friend function.*

#include<iostream>

using namespace std;

class Prices{

    int price1, price2;

    public:

        void getPrices(){

            cout << "Enter Prices of both the products : ";

            cin >> price1 >> price2;

        }

        friend void totalBill(Prices);

};

void totalBill(Prices p){

    cout << "Total Bill is : " << p.price1 + p.price2 << endl;

}

int main(){

    Prices wheat;

    wheat.getPrices();

    totalBill(wheat);

    return 0;

}

Q5

**CODE:**

*// Write a program to input three numbers in a member function and display maximum number using friend function.*

#include<iostream>

using namespace std;

class Numbers{

    int a, b, c;

    public:

        void get(int x, int y, int z){

            a = x;

            b = y;

            c = z;

        }

        friend void findMax(Numbers);

};

void findMax(Numbers n){

    if (n.a>n.b)

    {

        if (n.a>n.c)

        {

            cout << "The maximum number is : " << n.a << endl;

        }

        else

        {

            cout << "The maximum number is : " << n.c << endl;

        }

    }

    else

    {

        if (n.b>n.c)

        {

            cout << "The maximum number is : " << n.b << endl;

        }

        else

        {

            cout << "The maximum number is : " << n.c << endl;

        }

    }

};

int main(){

    Numbers n1;

    n1.get(71, 73, 70);

    findMax(n1);

    return 0;

}

Q6

**CODE:**

*// Write a program to calculate Simple Interest using friend function. The input must be taken in a member function designed outside the class.*

#include<iostream>

using namespace std;

class SimpleInterest{

    int p, r, t;

    public:

        void get();

        friend void calculate(SimpleInterest);

};

void SimpleInterest::get(){

    cout << "Enter Principal Amount : ";

    cin >> p;

    cout << "Enter Interest Rate : ";

    cin >> r;

    cout << "Enter Duration : ";

    cin >> t;

}

void calculate(SimpleInterest s1){

    cout << "Simple Interest is : " << (s1.p\*s1.r\*s1.t)/100 << endl;

}

int main(){

    SimpleInterest s;

    s.get();

    calculate(s);

    return 0;

}

Q7

**CODE:**

*// Write a program to find whether a number entered by user is odd or even using a friend function. The input must be taken using a member function defined outside the class.*

#include<iostream>

using namespace std;

class Number{

    int num = 0;

    public:

        void get();

        friend void check(Number);

};

void Number::get(){

    cout << "Enter a number : ";

    cin >> num;

}

void check(Number n){

    if (n.num % 2 == 0)

    {

        cout << "The entered number is even" << endl;

    }

    else

    {

        cout << "The entered number is odd" << endl;

    }

}

int main(){

    Number n1;

    n1.get();

    check(n1);

    return 0;

}

Q8

**CODE:**

*// Write a program to input one number in one class and another number in another class. Now make summation of these two numbers using a friend function.*

#include<iostream>

using namespace std;

class N1;

class N2;

class N1{

    int a;

    public:

        N1(int n) : a(n){};

        friend void sum(N1 a, N2 b);

};

class N2{

    int a;

    public:

        N2(int n) : a(n){};

        friend void sum(N1 a, N2 b);

};

void sum(N1 a, N2 b){

    cout << "The sum of both the numbers is : " << a.a + b.a << endl;

}

int main(){

    N1 a(10);

    N2 b(20);

    sum(a, b);

    return 0;

}

Q9

**CODE:**

*// Write a program to input one number in one class and another number in another class. Now largest of these two numbers using a friend function.*

#include<iostream>

using namespace std;

class N1;

class N2;

class N1{

    int a;

    public:

        N1(int n) : a(n){};

        friend void max(N1 a, N2 b);

};

class N2{

    int a;

    public:

        N2(int n) : a(n){};

        friend void max(N1 a, N2 b);

};

void max(N1 a, N2 b){

    if(a.a > b.a){

        cout << "The maximum of both is : " << a.a << endl;

    }

    else{

        cout << "The maximum of both is : " << b.a << endl;

    }

}

int main(){

    N1 a(10);

    N2 b(20);

    max(a, b);

    return 0;

}

Q10

**CODE:**

*// Write a program to input per day salary and number of days in a class “Salary”. Also input service tax and income tax in a class “Tax”. Design a friend function “Net\_Earning” where Total Salary-Total Tax will be calculated.*

#include<iostream>

using namespace std;

class Salary;

class Tax;

class Salary{

    int salary, days;

    public:

        Salary(int a, int b) : salary(a), days(b){};

        friend void net\_earning(Salary s, Tax t);

};

class Tax{

    int sTax, iTax;

    public:

        Tax(int a, int b) : sTax(a), iTax(b){};

        friend void net\_earning(Salary s, Tax t);

};

void net\_earning(Salary s, Tax t){

    cout << "Total Salary : " << s.salary \* s.days << endl;

    cout << "Total Tax : " << t.sTax + t.iTax << endl;

}

int main(){

    Salary s(100, 20);

    Tax t(300, 200);

    net\_earning(s, t);

    return 0;

}