Lecture 22 OOP’s

#include <bits/stdc++.h>

using namespace std;

class Car{

private:

    float price;

public:

    // float price;

    int model;

    char name[20];

    Car()

    {

        cout<<"Inside the car"<<endl;

    }

    // Car(float p, int m, char \*n)

    // {

    //     price = p;

    //     model = m;

    //     strcpy(name, n);

    // }

    float get\_discount(float x)

    {

        return price\*(1-x); // 100(1-0.30) = 70

    }

    void display()

    {

        cout<<"Name "<<name<<endl;

        cout<<"Price "<<price<<endl;

        cout<<"Model\_no "<<model<<endl;

    }

    void set\_price(float p)

    {

        price = p;

    }

    float get\_price()

    {

        return price;

    }

};

int main() {

    Car c, e, f;

    // c.price = 100;

    c.set\_price(300);

    c.model = 66;

    c.name[0] = 'B';

    c.name[1] = 'M';

    c.name[2] = 'W';

    c.name[3] = '\0';

    //Car d(150, 90, 'A');

    cout<<c.name<<" "<<" Model "<<c.model<<endl;

    float discount;

    cin>>discount;

    cout<<c.get\_discount(discount)<<endl;

    c.display();

    return 0;

}

Function Overloading:

#include <iostream>

using namespace std;

void display(int a)

{

    cout<<"Integer "<<a<<endl;

}

void display(double b)

{

    cout<<"Double "<<b<<endl;

}

void display(int a, double b)

{

    cout<<"Integer "<<a<<" Double "<<b<<endl;

}

int main() {

    int a = 10;

    double b = 20.5;

    display(a);

    display(b);

    display(a, b);

    return 0;

}