**Q1. Write a C++ Program illustrating the single inheritance feature and specific problem where we can use this concept for solve the specific problem using the single inheritance.**

#include <bits/stdc++.h>

using namespace std;

//Base class

class brand

{

public:

string brand;

};

// Sub class inheriting from Base Class

class model : public brand

{

public:

int model;

};

//main function

int main()

{

model latest;

latest.brand = "Iphone";

latest.model = 13;

cout << "Latest brand is " << latest.brand <<" Latest model is "<< latest.model<<endl;

return 0;

}



**Q2. Write a C++ Program illustrating the multiple inheritance feature and specific problem where we can use this concept for solve the specific problem using the multiple inheritance.**

#include<iostream>

using namespace std;

class A

{

public:

A() { cout << "A's constructor called" << endl; }

};

class B

{

public:

B() { cout << "B's constructor called" << endl; }

};

class C: public B, public A // Note the order

{

public:

C() { cout << "C's constructor called" << endl; }

};

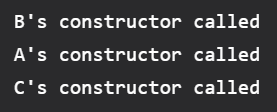
int main()

{

C c;

return 0;

}



**Q3. Write a C++ Program illustrating the multilevel inheritance feature and specific problem where we can use this concept for solve the specific problem using the multilevel inheritance.**

#include <iostream>

using namespace std;

class country

{

public:

country()

{

cout << "India\n";

}

};

class state : public country

{

public:

state()

{

cout << "Himachal Pradesh\n";

}

};

class city : public state

{

public:

city()

{

cout<<"Kullu\n";

}

};

class village : public city{

public:

village(){

cout<<"Malana";

}

};

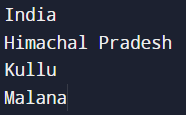
int main()

{

village a;

return 0;

}



**Q4. Write a C++ Program illustrating the hybrid inheritance feature and specific problem where we can use this concept for solve the specific problem using the hybrid inheritance.**

#include <iostream>

using namespace std;

class vehicle

{

public:

vehicle()

{

cout<< "This is a vehicle\n";

}

};

class Car: public vehicle

{

public:

Car()

{

cout<< "This is a car\n";

}

};

class Racing

{

public:

Racing()

{

cout<< "This is for Racing\n";

}

};

class Ferrari: public Car, public Racing

{

public:

Ferrari()

{

cout<< "Ferrari is a Racing Car\n";

}

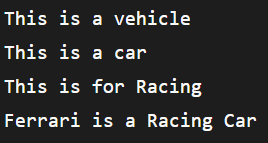
};

int main() {

Ferrari f;

return 0;

}



**Q5. Write a C++ Program illustrating the hierarchal inheritance feature and specific problem where we can use this concept for solve the specific problem using the hierarchal inheritance.**

#include <iostream>

using namespace std;

class A // Base class

{

public:

int x, y; // data members

voidgetdata() // to input x and y

{

cout<< "Enter value of x and y:\n";

cin>> x >> y;

}

};

class B : public A //B is derived from class base

{

public:

void product()

{

cout<< "\nProduct= " << x \* y <<endl; // Perform product

}

};

class C : public A //C is also derived from class base

{

public:

void sum()

{

cout<< "\nSum= " << x + y; // Perform sum

}

};

int main()

{

B obj1; //object of derived class B

C obj2; //object of derived class C

obj1.getdata(); // input x and y

obj1.product();

obj2.getdata();

obj2.sum();

return 0;

}

