**1.Create a base class called Shape with data members for height and width. Derive two classes Rectangle and Triangle from the base class. Write member functions to calculate the area and perimeter of each class**

#include<iostream>

using namespace std;

class shape {

public:

int length, breadth, height;

};

class rect : public shape {

public:

int area\_rect;

public:

void area() {

area\_rect = length \* breadth;

cout << "Area of rectangle: " << area\_rect << endl;

}

};

class triangle : public shape {

public:

int area\_tr;

public:

void area1() {

area\_tr = 0.5 \* length \* breadth \* height;

cout << "Area of triangle: " << area\_tr << endl;

}

};

int main() {

int l, b, h;

cout << "Length of the shape: ";

cin >> l;

cout << "Breadth of the shape: ";

cin >> b;

cout << "Height of the shape: ";

cin >> h;

triangle obj;

obj.length = l;

obj.breadth = b;

obj.height = h;

obj.area1();

rect obj1;

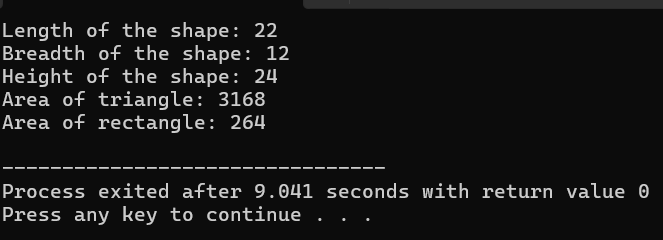
obj1.length = l;

obj1.breadth = b;

obj1.area();

return 0;

}



**2.Create a base class called vehicle with data members for make, model, and year. Derive two classes Car and Truck from the base class. The Car class should have additional data members for seating capacity and fuel type, while the Truck class should have additional data members for payload capacity and towing capacity. Write member functions to get and set the data members for each class**

#include<iostream>

using namespace std;

class vehicle{

public:

int year,make;

string model;

};

class car:public vehicle{

public:

string fuel;

int capcity;

public:

void veh1(string fuel,int capcity){

cout<<"model is :"<<model<<endl;

cout<<"make is:"<<make<<endl;

cout<<"year is :"<<year<<endl;

cout<<"fuel is:"<<fuel<<endl;

cout<<"capcity is :"<<capcity;

}

};

class truck:public vehicle{

public:

int payload,towing;

public:

void veh2(int payload,int towing){

cout<<"model is :"<<model<<endl;

cout<<"make is:"<<make<<endl;

cout<<"year is :"<<year<<endl;

cout<<"payload is :"<<payload;

cout<<"towing is :"<<towing;

}

};

int main(){

int year,take,capcity,payload,towing,make;

string fuel,model;

cout<<"car detalis:"<<endl;

cout<<"enter the model:";

cin>>model;

cout<<"enter the year:";

cin>>year;

cout<<"enter the make:";

cin>>make;

cout<<"fuel is :"<<fuel;

cin>>fuel;

cout<<"capcity is:";

cin>>capcity;

car obj;

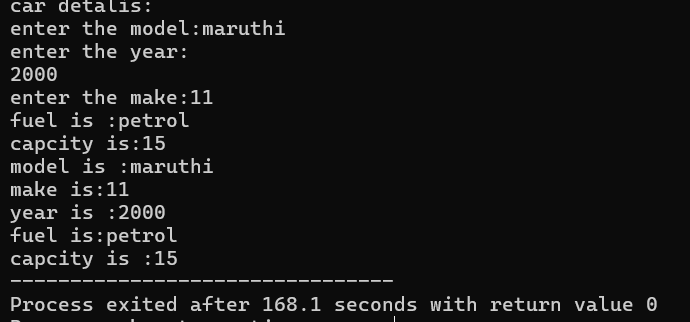
obj.model=model;

obj.year=year;

obj.make=make;

obj.veh1(fuel,capcity);

}



**3.Create a base class called Animal with data members for name, species, and age. Derive two classes Cat and Dog from the base class. The Cat class should have additional data members for color and breed, while the Dog class should have additional data members for weight and breed. Write member functions to get and set the data members for each class**

#include <iostream>

#include <string>

using namespace std;

class Animal {

protected:

string name;

string species;

int age;

public:

Animal(string \_name, string \_species, int \_age) : name(\_name), species(\_species), age(\_age) {}

string getName() const {

return name;

}

string getSpecies() const {

return species;

}

int getAge() const {

return age;

}

};

class Cat : public Animal {

private:

string color;

string breed;

public:

Cat(string \_name, string \_species, int \_age, string \_color, string \_breed)

: Animal(\_name, \_species, \_age), color(\_color), breed(\_breed) {}

string getColor() const {

return color;

}

string getBreed() const {

return breed;

}

};

class Dog : public Animal {

private:

double weight;

string breed;

public:

Dog(string \_name, string \_species, int \_age, double \_weight, string \_breed)

: Animal(\_name, \_species, \_age), weight(\_weight), breed(\_breed) {}

double getWeight() const {

return weight;

}

string getBreed() const {

return breed;

}

};

int main() {

Cat myCat("Fluffy", "Cat", 3, "White", "Persian");

Dog myDog("Buddy", "Dog", 5, 10.5, "Golden Retriever");

cout << "Cat - Name: " << myCat.getName() << ", Species: " << myCat.getSpecies() << ", Age: " << myCat.getAge()

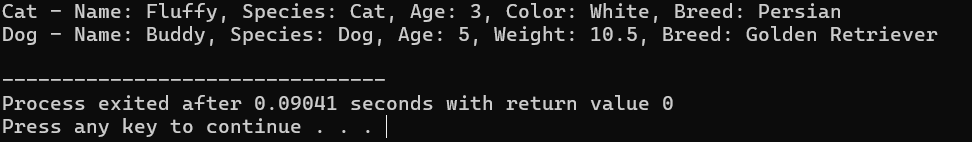
<< ", Color: " << myCat.getColor() << ", Breed: " << myCat.getBreed() << endl;

cout << "Dog - Name: " << myDog.getName() << ", Species: " << myDog.getSpecies() << ", Age: " << myDog.getAge()

<< ", Weight: " << myDog.getWeight() << ", Breed: " << myDog.getBreed() << endl;

return 0;

}



**4.Create a base class called Employee with data members for name, d, and salary Derive two classes Manager and Engineer from the base class. The Manager class should have additional data members for department and bonus, while the Engineer class should have additional data members for specialty and hours. Write member functions to get and set the data members for each class**

#include <iostream>

#include <string>

using namespace std;

class Employee {

protected:

string name;

int id;

double salary;

public:

Employee(string \_name, int \_id, double \_salary) : name(\_name), id(\_id), salary(\_salary) {}

string getName() const {

return name;

}

void setName(string \_name) {

name = \_name;

}

int getId() const {

return id;

}

void setId(int \_id) {

id = \_id;

}

double getSalary() const {

return salary;

}

void setSalary(double \_salary) {

salary = \_salary;

}

};

class Manager : public Employee {

private:

string department;

double bonus;

public:

Manager(string \_name, int \_id, double \_salary, string \_department, double \_bonus)

: Employee(\_name, \_id, \_salary), department(\_department), bonus(\_bonus) {}

string getDepartment() const {

return department;

}

void setDepartment(string \_department) {

department = \_department;

}

double getBonus() const {

return bonus;

}

void setBonus(double \_bonus) {

bonus = \_bonus;

}

};

class Engineer : public Employee {

private:

string specialty;

int hours;

public:

Engineer(string \_name, int \_id, double \_salary, string \_specialty, int \_hours)

: Employee(\_name, \_id, \_salary), specialty(\_specialty), hours(\_hours) {}

string getSpecialty() const {

return specialty;

}

void setSpecialty(string \_specialty) {

specialty = \_specialty;

}

int getHours() const {

return hours;

}

void setHours(int \_hours) {

hours = \_hours;

}

};

int main() {

Manager manager("John Doe", 101, 5000, "HR", 1000);

Engineer engineer("Alice Smith", 102, 4000, "Software Development", 40);

cout << "Manager - Name: " << manager.getName() << ", ID: " << manager.getId()

<< ", Salary: " << manager.getSalary() << ", Department: " << manager.getDepartment()

<< ", Bonus: " << manager.getBonus() << endl;

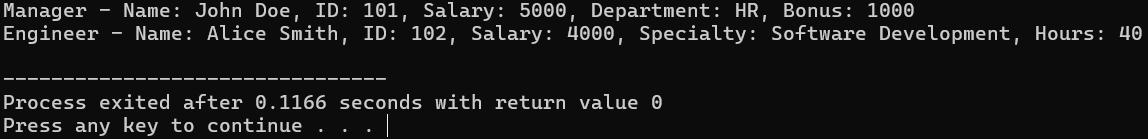
cout << "Engineer - Name: " << engineer.getName() << ", ID: " << engineer.getId()

<< ", Salary: " << engineer.getSalary() << ", Specialty: " << engineer.getSpecialty()

<< ", Hours: " << engineer.getHours() << endl;

return 0;

}



**5.Create a base class called Person with data members for name, age, and gender. Derive two classes Student and Teacher from the base class. The student class should have additional data members for roll number and class, while the Teacher class should have additional data members for subject and salary. Write member functions to get and set the data members for each class.**

#include <iostream>

#include <string>

using namespace std;

class Person {

protected:

string name;

int age;

char gender;

public:

Person(string \_name, int \_age, char \_gender) : name(\_name), age(\_age), gender(\_gender) {}

string getName() const {

return name;

}

void setName(string \_name) {

name = \_name;

}

int getAge() const {

return age;

}

void setAge(int \_age) {

age = \_age;

}

char getGender() const {

return gender;

}

void setGender(char \_gender) {

gender = \_gender;

}

};

class Student : public Person {

private:

int rollNumber;

string className;

public:

Student(string \_name, int \_age, char \_gender, int \_rollNumber, string \_className)

: Person(\_name, \_age, \_gender), rollNumber(\_rollNumber), className(\_className) {}

int getRollNumber() const {

return rollNumber;

}

void setRollNumber(int \_rollNumber) {

rollNumber = \_rollNumber;

}

string getClassName() const {

return className;

}

void setClassName(string \_className) {

className = \_className;

}

};

class Teacher : public Person {

private:

string subject;

double salary;

public:

Teacher(string \_name, int \_age, char \_gender, string \_subject, double \_salary)

: Person(\_name, \_age, \_gender), subject(\_subject), salary(\_salary) {}

string getSubject() const {

return subject;

}

void setSubject(string \_subject) {

subject = \_subject;

}

double getSalary() const {

return salary;

}

void setSalary(double \_salary) {

salary = \_salary;

}

};

int main() {

Student student("Alice", 20, 'F', 101, "12th Grade");

Teacher teacher("Mr. Smith", 35, 'M', "Mathematics", 50000.0);

cout << "Student - Name: " << student.getName() << ", Age: " << student.getAge() << ", Gender: " << student.getGender()

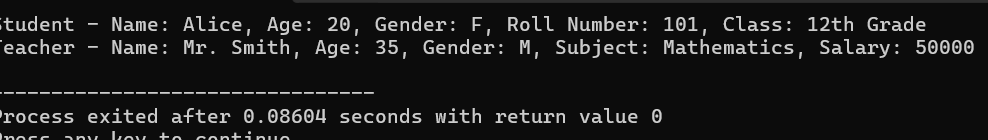
<< ", Roll Number: " << student.getRollNumber() << ", Class: " << student.getClassName() << endl;

cout << "Teacher - Name: " << teacher.getName() << ", Age: " << teacher.getAge() << ", Gender: " << teacher.getGender()

<< ", Subject: " << teacher.getSubject() << ", Salary: " << teacher.getSalary() << endl;

return 0;

}



**6.Write a C++ program to create a pointer to an integer and display its value.**

#include<iostream>

using namespace std;

int main(){

int num;

cout<<"enter the given number:";

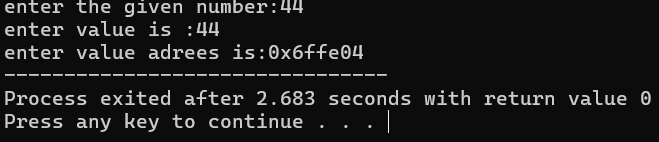
cin>>num;

int \*ptr=&num;

cout<<"enter value is :"<<num<<endl;

cout<<"enter value adrees is:"<<&num;

}



**7.Write a C++ program to create a pointer to a float and display its value.**

#include<iostream>

using namespace std;

int main(){

float num;

cout<<"enter the given number:";

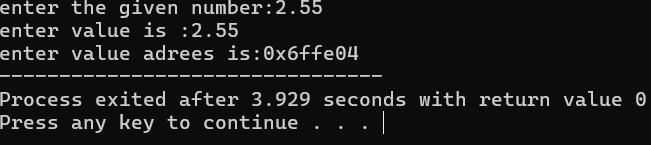
cin>>num;

float \*ptr= &num;

cout<<"enter value is :"<<num<<endl;

cout<<"enter value adrees is:"<<&num;

}



**8.Write a C++ program to create a pointer to a char and display its value**.

#include <iostream>

using namespace std;

int main() {

char \*ptr;

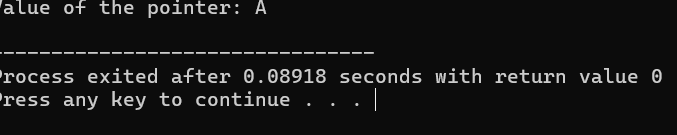
char ch = 'A';

ptr = &ch;

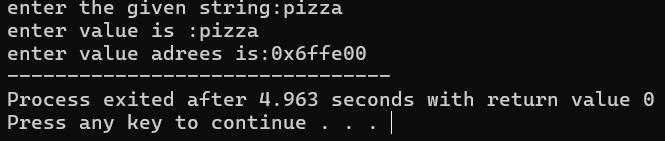
cout << "Value of the pointer: " << \*ptr << endl;

return 0;

}



**9.Write a C++ program to create a pointer to a string and display its value.**



**10.Write a C++ program to create a pointer to a double and display its value.**

#include <iostream>

using namespace std;

int main() {

double \*ptr;

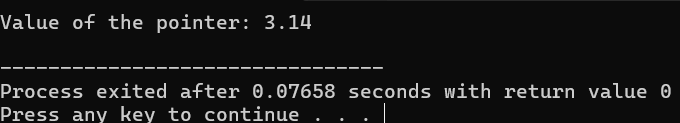
double value = 3.14;

ptr = &value;

cout << "Value of the pointer: " << \*ptr << endl;

return 0;

}

****

.

**11.Write a C++ program to create a pointer to an array of elements and display its value**.

#include <iostream>

using namespace std;

int main() {

int arr[] = {1, 2, 3, 4, 5};

int \*ptr = arr;

cout << "Values of the array using pointer:" << endl;

for (int i = 0; i < 5; ++i) {

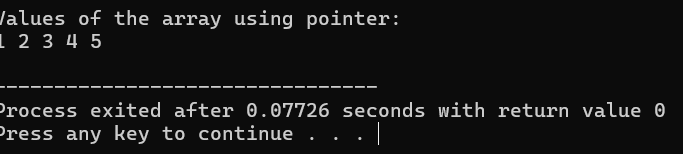
cout << \*(ptr + i) << " ";

}

cout << endl;

return 0;

}



**12.Write a C++ program to create a pointer to an array of character and display its value**.

#include <iostream>

using namespace std;

int main() {

char arr[] = {'H', 'e', 'l', 'l', 'o', '\0'};

char \*ptr = arr;

cout << "Value of the array using pointer: ";

while (\*ptr != '\0') {

cout << \*ptr;

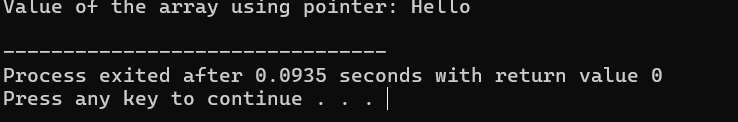
ptr++;

}

cout << endl;

return 0;

}



**13.Write a C++ program to create a pointer to an array of floats and display its value.**

#include <iostream>

using namespace std;

int main() {

float arr[] = {1.1f, 2.2f, 3.3f, 4.4f, 5.5f};

float \*ptr = arr;

cout << "Values of the array using pointer:" << endl;

for (int i = 0; i < 5; ++i) {

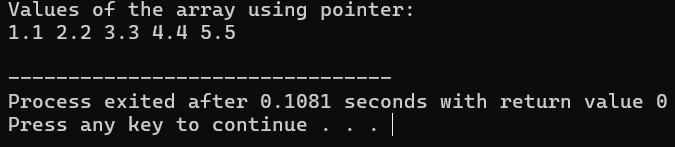
cout << \*(ptr + i) << " ";

}

cout << endl;

return 0;

}



**14.Write a C++ program to create a pointer to an object and display its attributes.**

#include <iostream>

using namespace std;

int add(int a, int b) {

return a + b;

}

int main() {

int (\*ptr)(int, int);

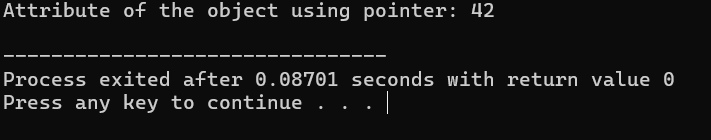
ptr = &add;

int result = ptr(5, 3);

cout << "Result of adding 5 and 3: " << result << endl;

return 0;

}



**15.Write a C++ program to create a pointer to a function and call the function using the pointer.**

#include <iostream>

using namespace std;

class MyClass {

public:

int attribute;

MyClass(int \_attribute) : attribute(\_attribute) {}

};

int main() {

MyClass obj(42);

MyClass \*ptr = &obj;

cout << "Attribute of the object using pointer: " << ptr->attribute << endl;

return 0;

}

