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

Code:

#include <iostream>

#include <cmath>

class Shape {

protected:

double height;

double width;

public:

Shape(double h, double w) : height(h), width(w) {}

virtual double area() const = 0;

virtual double perimeter() const = 0;

};

class Rectangle : public Shape {

public:

Rectangle(double h, double w) : Shape(h, w) {}

double area() const override {

return height \* width;

}

double perimeter() const override {

return 2 \* (height + width);

}

};

class Triangle : public Shape {

private:

double side1, side2, side3;

public:

Triangle(double h, double w, double s1, double s2, double s3) : Shape(h, w), side1(s1), side2(s2), side3(s3) {}

double area() const override {

double s = (side1 + side2 + side3) / 2;

return std::sqrt(s \* (s - side1) \* (s - side2) \* (s - side3));

}

double perimeter() const override {

return side1 + side2 + side3;

}

};

int main() {

Rectangle rect(4, 5);

Triangle tri(3, 4, 5, 4, 3);

std::cout << "Rectangle Area: " << rect.area() << ", Perimeter: " << rect.perimeter() << std::endl;

std::cout << "Triangle Area: " << tri.area() << ", Perimeter: " << tri.perimeter() << std::endl;

return 0;

}

Output:

Rectangle Area: 20, Perimeter: 18

Triangle Area: 6, Perimeter: 12

1. 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

Code:

#include <iostream>

#include <string>

using namespace std;

class Vehicle {

protected:

string make;

string model;

int year;

public:

Vehicle(const string& make, const string& model, int year) : make(make), model(model), year(year) {}

string getMake() const { return make; }

void setMake(const string& newMake) { make = newMake; }

string getModel() const { return model; }

void setModel(const string& newModel) { model = newModel; }

int getYear() const { return year; }

void setYear(int newYear) { year = newYear; }

};

class Car : public Vehicle {

private:

int seatingCapacity;

string fuelType;

public:

Car(const string& make, const string& model, int year, int seatingCapacity, const string& fuelType)

: Vehicle(make, model, year), seatingCapacity(seatingCapacity), fuelType(fuelType) {}

int getSeatingCapacity() const { return seatingCapacity; }

void setSeatingCapacity(int capacity) { seatingCapacity = capacity; }

string getFuelType() const { return fuelType; }

void setFuelType(const string& type) { fuelType = type; }

};

class Truck : public Vehicle {

private:

int payloadCapacity;

int towingCapacity;

public:

Truck(const string& make, const string& model, int year, int payloadCapacity, int towingCapacity)

: Vehicle(make, model, year), payloadCapacity(payloadCapacity), towingCapacity(towingCapacity) {}

int getPayloadCapacity() const { return payloadCapacity; }

void setPayloadCapacity(int capacity) { payloadCapacity = capacity; }

int getTowingCapacity() const { return towingCapacity; }

void setTowingCapacity(int capacity) { towingCapacity = capacity; }

};

int main() {

Car car("Toyota", "Camry", 2022, 5, "Gasoline");

Truck truck("Ford", "F-150", 2021, 2000, 8000);

cout << "Car Make: " << car.getMake() << ", Model: " << car.getModel() << ", Year: " << car.getYear() << endl;

cout << "Car Seating Capacity: " << car.getSeatingCapacity() << ", Fuel Type: " << car.getFuelType() << endl;

cout << "Truck Make: " << truck.getMake() << ", Model: " << truck.getModel() << ", Year: " << truck.getYear() << endl;

cout << "Truck Payload Capacity: " << truck.getPayloadCapacity() << ", Towing Capacity: " << truck.getTowingCapacity() << endl;

return 0;

}

Output:

Car Make: Toyota, Model: Camry, Year: 2022

Car Seating Capacity: 5, Fuel Type: Gasoline

Truck Make: Ford, Model: F-150, Year: 2021

Truck Payload Capacity: 2000, Towing Capacity: 8000

1. 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

Code:

#include <iostream>

#include <string>

using namespace std;

class Animal {

protected:

string name;

string species;

int age;

public:

Animal(const string& name, const string& species, int age)

: name(name), species(species), age(age) {}

string getName() const { return name; }

void setName(const string& newName) { name = newName; }

string getSpecies() const { return species; }

void setSpecies(const string& newSpecies) { species = newSpecies; }

int getAge() const { return age; }

void setAge(int newAge) { age = newAge; }

};

class Cat : public Animal {

private:

string color;

string breed;

public:

Cat(const string& name, const string& species, int age, const string& color, const string& breed)

: Animal(name, species, age), color(color), breed(breed) {}

string getColor() const { return color; }

void setColor(const string& newColor) { color = newColor; }

string getBreed() const { return breed; }

void setBreed(const string& newBreed) { breed = newBreed; }

};

class Dog : public Animal {

private:

double weight;

string breed;

public:

Dog(const string& name, const string& species, int age, double weight, const string& breed)

: Animal(name, species, age), weight(weight), breed(breed) {}

double getWeight() const { return weight; }

void setWeight(double newWeight) { weight = newWeight; }

string getBreed() const { return breed; }

void setBreed(const string& newBreed) { breed = newBreed; }

};

int main() {

Cat cat("Whiskers", "Cat", 3, "Tabby", "Maine Coon");

Dog dog("Buddy", "Dog", 5, 25.5, "Golden Retriever");

cout << "Cat Name: " << cat.getName() << ", Species: " << cat.getSpecies() << ", Age: " << cat.getAge() << endl;

cout << "Cat Color: " << cat.getColor() << ", Breed: " << cat.getBreed() << endl;

cout << "Dog Name: " << dog.getName() << ", Species: " << dog.getSpecies() << ", Age: " << dog.getAge() << endl;

cout << "Dog Weight: " << dog.getWeight() << " lbs, Breed: " << dog.getBreed() << endl;

return 0;

}

Output:

Cat Name: Whiskers, Species: Cat, Age: 3

Cat Color: Tabby, Breed: Maine Coon

Dog Name: Buddy, Species: Dog, Age: 5

Dog Weight: 25.5 lbs, Breed: Golden Retriever

1. 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

Code:

#include <iostream>

#include <string>

using namespace std;

class Employee {

protected:

string name;

int id;

double salary;

public:

Employee(const string& name, int id, double salary)

: name(name), id(id), salary(salary) {}

string getName() const { return name; }

void setName(const string& newName) { name = newName; }

int getId() const { return id; }

void setId(int newId) { id = newId; }

double getSalary() const { return salary; }

void setSalary(double newSalary) { salary = newSalary; }

};

class Manager : public Employee {

private:

string department;

double bonus;

public:

Manager(const string& name, int id, double salary, const string& department, double bonus)

: Employee(name, id, salary), department(department), bonus(bonus) {}

string getDepartment() const { return department; }

void setDepartment(const string& newDepartment) { department = newDepartment; }

double getBonus() const { return bonus; }

void setBonus(double newBonus) { bonus = newBonus; }

};

class Engineer : public Employee {

private:

string specialty;

int hours;

public:

Engineer(const string& name, int id, double salary, const string& specialty, int hours)

: Employee(name, id, salary), specialty(specialty), hours(hours) {}

string getSpecialty() const { return specialty; }

void setSpecialty(const string& newSpecialty) { specialty = newSpecialty; }

int getHours() const { return hours; }

void setHours(int newHours) { hours = newHours; }

};

int main() {

Manager manager("John Doe", 1001, 75000, "Marketing", 10000);

Engineer engineer("Jane Smith", 2001, 80000, "Software Development", 40);

cout << "Manager Name: " << manager.getName() << ", ID: " << manager.getId() << ", Salary: $" << manager.getSalary() << endl;

cout << "Manager Department: " << manager.getDepartment() << ", Bonus: $" << manager.getBonus() << endl;

cout << "Engineer Name: " << engineer.getName() << ", ID: " << engineer.getId() << ", Salary: $" << engineer.getSalary() << endl;

cout << "Engineer Specialty: " << engineer.getSpecialty() << ", Hours: " << engineer.getHours() << endl;

return 0;

}

Output:

Manager Name: John Doe, ID: 1001, Salary: $75000

Manager Department: Marketing, Bonus: $10000

Engineer Name: Jane Smith, ID: 2001, Salary: $80000

Engineer Specialty: Software Development, Hours: 40

1. 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.

Code:

#include <iostream>

#include <string>

using namespace std;

class Person {

protected:

string name;

int age;

char gender;

public:

Person(const string& name, int age, char gender)

: name(name), age(age), gender(gender) {}

string getName() const { return name; }

void setName(const string& newName) { name = newName; }

int getAge() const { return age; }

void setAge(int newAge) { age = newAge; }

char getGender() const { return gender; }

void setGender(char newGender) { gender = newGender; }

};

class Student : public Person {

private:

int rollNumber;

string className;

public:

Student(const string& name, int age, char gender, int rollNumber, const string& className)

: Person(name, age, gender), rollNumber(rollNumber), className(className) {}

int getRollNumber() const { return rollNumber; }

void setRollNumber(int newRollNumber) { rollNumber = newRollNumber; }

string getClassName() const { return className; }

void setClassName(const string& newClassName) { className = newClassName; }

};

class Teacher : public Person {

private:

string subject;

double salary;

public:

Teacher(const string& name, int age, char gender, const string& subject, double salary)

: Person(name, age, gender), subject(subject), salary(salary) {}

string getSubject() const { return subject; }

void setSubject(const string& newSubject) { subject = newSubject; }

double getSalary() const { return salary; }

void setSalary(double newSalary) { salary = newSalary; }

};

int main() {

Student student("John Doe", 18, 'M', 101, "12th Grade");

Teacher teacher("Jane Smith", 35, 'F', "Mathematics", 50000);

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

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

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

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

return 0;

}

Output:

Student Name: John Doe, Age: 18, Gender: M

Student Roll Number: 101, Class: 12th Grade

Teacher Name: Jane Smith, Age: 35, Gender: F

Teacher Subject: Mathematics, Salary: $50000