


In class Practice 11/1/2019

Group three or four people team.

Part I

public protected

1. A base class's _____ and _____ members can be accessed in the base-class definition, in derived-class definitions and in friends of the base class and derived classes.
2. In a(n) _____ relationship, an object of a derived class also can be treated as an object of its base class. is-a or inheritance
3. In a(n) _____ relationship, a class object has one or more objects of other classes as members. has-a or aggregation or composition
4. A base class's protected access members have a level of protection between those of public and private access.
5. C++ provides for multiple inheritance, which allows a derived class to inherit from many base classes, even if the base classes are unrelated.
6. When an object of a derived class is instantiated, the base class's constructor is called implicitly or explicitly to do any necessary initialization of the base-class data members in the derived-class object.
7. When deriving a class with public inheritance, public members of the base class become public members of the derived class, and protected members of the base class become protected members of the derived class.
8. When deriving from a class with protected inheritance, public members of the base class become protected members of the derived class, and protected members of the base class become protected members of the derived class.
9. Which of the following is the correct function definition header for the getAge function which is a member of the Person class?
 - a. int getAge();
 - b. int getAge()
 - c. int Person:getAge()
 - d.  int Person::getAge()

10. Given the following class definition, how would you declare an object of the class, so that the object automatically called the default constructor?

```
class ItemClass
{
public:
    ItemClass();
    ItemClass(int newSize, float newCost);
    int getSize();
    float getCost();
    void setSize(int newSize);
    void setCost(float newCost);
private:
    int size;
    float cost;
};
```

- a. ItemClass() myItem;
- b. ItemClass myItem(1, 0.0);
- c. ItemClass myItem;
- d. ItemClass myItem();
- e. You can not do this

Part II

Answer the following question

1. (Student Inheritance Hierarchy) Draw an inheritance hierarchy for students at a university similar to the hierarchy shown in Fig. 11.2. Use Student as the base class of the hierarchy, then include classes UndergraduateStudent and GraduateStudent that derive from Student. Continue to extend the hierarchy as deep (i.e., as many levels) as possible. For example, Freshman, Sophomore, Junior and Senior might derive from UndergraduateStudent, and DoctoralStudent and MastersStudent might derive from GraduateStudent. After drawing the hierarchy, discuss the relationships that exist between the classes. [Note: You do not need to write any code for this exercise.] **in the attached picture**

Part III

1. Create two classes Account and CheckingAccount. Account is base class and CheckingAccount is a derived class from Account. An Account object contains an attribute called balance. **One setter and one getter** for this data member balance. Two other member functions. The function debit will subtract amount to balance and the other function credit will add an amount to the balance. Account has a one parameter constructor which will initialize the balance data.

The CheckingAccount which inherited from Account has one constructor with two arguments. The first argument is account **balance** and the second one is **transaction fee**. Transaction fee will be charged to credit call or debit call when amount was added or subtracted from balance. So you have to redefine the credit and debit member function of base class in the CheckingAccount class. You may define help function as needed.