Quiz 12 Answers

1. In order to use a `dynamic_cast<>` in a C++ program, runtime type information must be enabled.

   true

2. In C++, a call to a `virtual` method though a base class pointer to a derived class object will use

   dynamic binding / dynamic linkage.

3. With `public` inheritance, the `protected` members of a base class can be accessed by

   All of the above.

4. What is the C++ syntax for declaring the class `Dog` to be a derived class of the base class `Pet`?

   ```
   class Dog : public Pet
   ```

5. When a derived class object is destroyed, in what order are the destructors executed?

   First the derived class destructor is executed, then the base class destructor is executed.

6. In C++, what is a *pure virtual method*?

   A pure virtual method has no definition, just a prototype.

7. In C++, what is an *abstract* class?

   An abstract class is a class that contains at least one pure virtual method.

8. Assume that `Pet` is a base class, and `Cat` is a class derived from `Pet` using `public` inheritance. What is the correct way to call the `print()` method of the `Pet` class from the `print()` method of the `Cat` class?

   ```
   Pet::print();
   ```

9. When a derived class object is created, in what order are the constructor bodies executed?

   First the base class constructor body is executed, then the derived class constructor body is executed.

10. A base class pointer that points to a derived class object can be used to call

    only methods declared in the base class.