

# Module 1 Review

TOTAL POINTS 12

1. When is the best time to use a design pattern? Choose **two** answers.

1 point

- ☐ For a problem that is unique to your program.
- ☒ For a commonly-encountered issue.
- ☒ When fixing spaghetti code
- ☐ When explaining a solution to your fellow developers

2. What is the purpose of the Singleton pattern? Select the **two correct** answers.

1 point

- ☐ to provide global access to an object
- ☐ to provide simple classes with only one method
- ☐ to enforce collaboration of a class with only one other class
- ☒ to enforce instantiation of only one object of a class

3. What does it mean to "let the subclass decide" in the Factory Method Pattern?

1 point

- ☐ the subclass defines the methods for concrete instantiation. As such, the type of object is determined by which subclass is instantiated.
- ☐ the subclass will pass a parameter into a factory that determines which object is instantiated.
- ☐ the subclass decides which object to create, but calls a method that is defined in the superclass to instantiate the class

4. What do we call the creation of an object, for example, with the 'new' operator in Java?

1 point

- ☒ concrete instantiation.
- ☐ class creation
- ☐ manifestation
- ☐ object realization

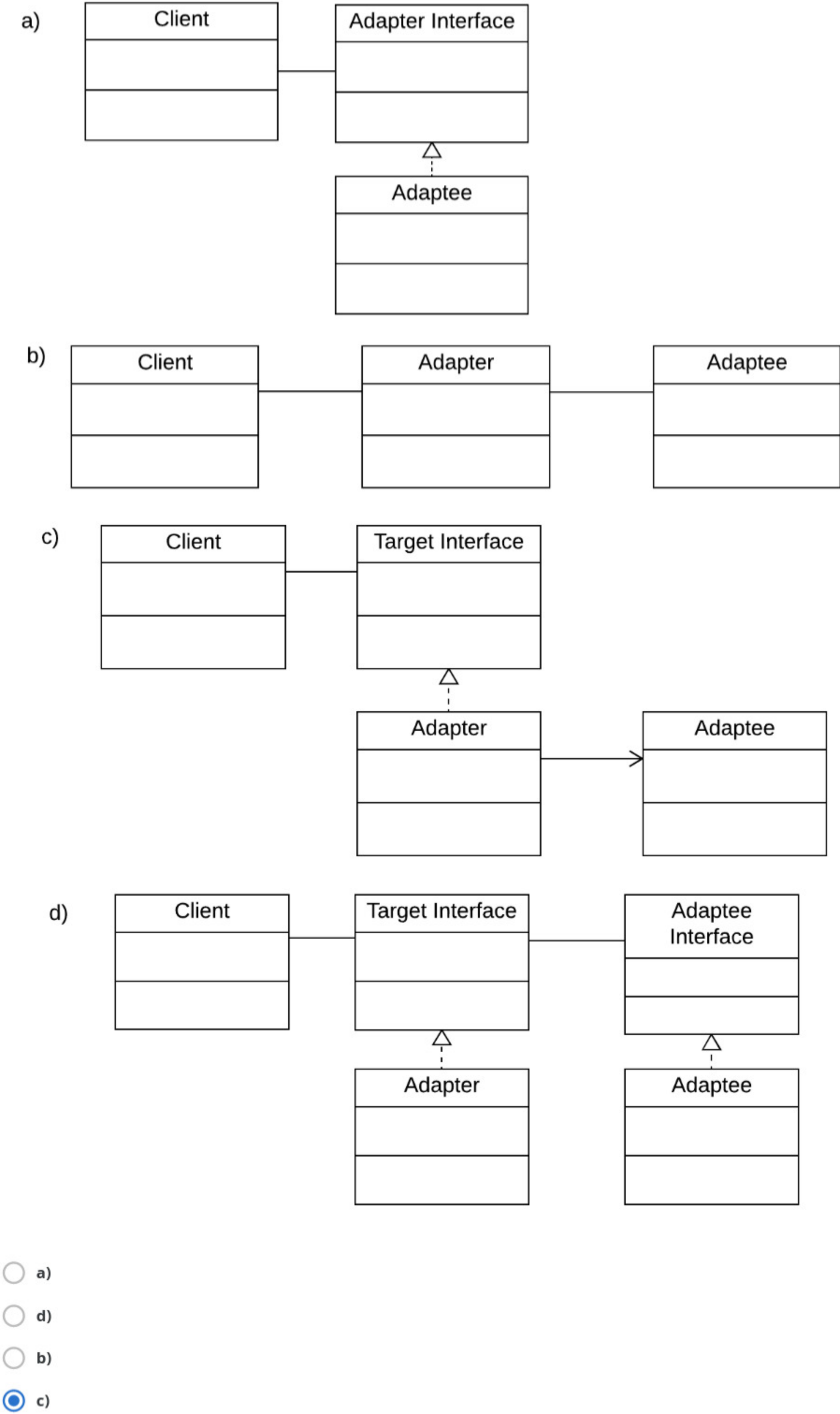
5. What are the advantages of the **Facade** pattern? Select the **three correct** answers.

1 point

- ☒ The Facade class redirects requests as needed
- ☒ The client and the subsystem are more loosely coupled
- ☐ The subsystem can handle more clients
- ☒ The complexity of the subsystem is hidden

6. Which of the following diagrams shows the **Adapter** pattern?

1 point



7. Which of these are the best applications for a **Composite** Pattern? Choose the **three correct** answers.

1 point

- ☐ Students in a class
- ☒ Elements in a user-interface dialog
- ☒ Music in a playlist
- ☒ Files and folders

8. Which of these is **NOT** a common application of the **Proxy** Pattern?

1 point

- ☐ remote proxy
- ☒ information proxy
- ☐ protection proxy
- ☐ virtual proxy

9. How does a **Decorator** Pattern work? Choose one.

1 point

- ☐ encapsulates a class to give it a different interface
- ☒ builds a behaviour by stacking objects
- ☐ expands the methods of a class with inheritance
- ☐ adding features to a class with a new class

10. What are the object types that are used in the **Composite** Pattern? Select the **two correct** answers.

1 point

- ☐ branch
- ☐ root
- ☒ leaf
- ☐ trunk
- ☒ composite

11. Many different clients need to create a similar object. You would like to outsource this concrete instantiation to a dedicated class. Which technique will you use, in one word?

1 point

Enter answer here

12. How do you enforce the creation of only one Singleton object? Select the **two correct** answers.

1 point

- ☐ Specify in the comments that only one Singleton object is to be instantiated.
- ☐ Throw an exception if a Singleton object is already instantiated
- ☒ Write a method that can create a new Singleton object or return the existing one.
- ☒ Give the Singleton class a private constructor

Upgrade to submit