

1

Which of the following statements about design principles and design patterns are true?
(Choose all that apply.)

- A. A design principle is focused on solving a specific commonly occurring problem.
- B. Design principles and design patterns are the same thing.
- C. Design principles are often applied throughout an application, whereas design patterns are applied to solve specific problems.
- D. Design patterns can only be applied to static classes.
- E. Design principles and design patterns tend to produce code that is easier to maintain and easier for other developers to read.

2

What is the result of the following code?

```
1: public interface CanClimb {  
2:     public abstract void climb();  
3: }  
4: public interface CanClimbTrees extends CanClimb {}  
5: public abstract class Chipmunk implements CanClimbTrees {  
6:     public abstract void chew();  
7: }  
8: public class EasternChipmunk extends Chipmunk {  
9:     public void chew() { System.out.println("Eastern Chipmunk is Chewing"); }  
10: }
```

- A. It compiles and runs without issue.
- B. The code will not compile because of line 2.
- C. The code will not compile because of line 4.
- D. The code will not compile because of line 5.
- E. The code will not compile because of line 8.
- F. It compiles but throws an exception at runtime.

3

Which of the following are valid functional interfaces? (Choose all that apply.)

```
public interface Climb {  
    public int climb();  
}  
  
public abstract class Swim {  
    public abstract Object swim(double speed, int duration);  
}  
  
public interface ArcticMountainClimb extends MountainClimb {  
    public default int getSpeed();  
}  
  
public interface MountainClimb extends Climb {}
```

- A. Climb
- B. Swim
- C. ArcticMountainClimb
- D. MountainClimb
- E. None of these are valid functional interfaces.

4

What are some of the properties of using the singleton pattern? (Choose all that apply.)

- A. Singleton object can be replaced with encapsulated setter method.
- B. Requires constructor of singleton class to be private.
- C. Singleton object must be named instance.
- D. Singleton object may be private or protected.
- E. Ensures that there is only one instance of an object in memory.
- F. Requires a public static method to retrieve the instance of the singleton.

5

What changes need to be made to make the following immutable object pattern correct?
(Choose all that apply.)

```
import java.util.List;
public class Seal {
    String name;
    private final List<Seal> friends;
    public Seal(String name, List<Seal> friends) {
        this.name = name;
        this.friends = friends;
    }
    public String getName() { return name; }
    public List<Seal> getFriends() { return friends; }
}
```

- A. None; the immutable object pattern is properly implemented.
- B. Have Seal implement the Immutable interface.
- C. Mark name final and private.
- D. Add setters for name and List<Seal> friends.
- E. Replace the getFriends() method with methods that do not give the caller direct access to the List<Seal> friends.
- F. Change the type of List<Seal> to List<Object>.
- G. Make a copy of the List<Seal> friends in the constructor.
- H. Mark the Seal class final.

6

Which of the following are true of interfaces? (Choose all that apply.)

- A. They can extend other classes.
- B. They cannot be extended.
- C. They enable classes to have multiple inheritance.
- D. They can only contain abstract methods.
- E. They can be declared final.
- F. All members of an interface are public.

7

What changes need to be made to make the following singleton pattern correct? (Choose all that apply.)

```
public class CheetahManager {  
    public static CheetahManager cheetahManager;  
    private CheetahManager() {}  
    public static CheetahManager getCheetahManager() {  
        if(cheetahManager == null) {  
            cheetahManager = new CheetahManager();  
        }  
        return cheetahManager;  
    }  
}
```

- A. None; the singleton pattern is properly implemented.
- B. Rename cheetahManager to instance.
- C. Rename getCheetahManager() to getInstance().
- D. Change the access modifier of cheetahManager from public to private.
- E. Mark cheetahManager final.
- F. Add synchronized to getCheetahManager().

8

What is the result of the following code?

```
1: public interface CanWalk {  
2:     default void walk() { System.out.println("Walking"); }  
3: }  
4: public interface CanRun {  
5:     public default void walk() { System.out.println("Walking"); }  
6:     public abstract void run();  
7: }  
8: public interface CanSprint extends CanWalk, CanRun {  
9:     void sprint();  
10: }
```

- A. The code compiles without issue.
- B. The code will not compile because of line 5.
- C. The code will not compile because of line 6.
- D. The code will not compile because of line 8.
- E. The code will not compile because of line 9.

9

Which of the following are properties of classes that define immutable objects? (Choose all that apply.)

- A. They don't define any getter methods.
- B. All of the instance variables marked `private` and `final`.
- C. They don't define any setter methods.
- D. They mark all instance variables `static`.
- E. They prevent methods from being overridden.
- F. All getter methods are marked `synchronized`.

10

Which of the following statements can be inserted in the blank line so that the code will compile successfully? (Choose all that apply.)

```
public interface CanHop {}  
public class Frog implements CanHop {  
    public static void main(String[] args) {  
        _____ frog = new TurtleFrog();  
    }  
}  
public class BrazilianHornedFrog extends Frog {}  
public class TurtleFrog extends Frog {}
```

- A. `Frog`
- B. `TurtleFrog`
- C. `BrazilianHornedFrog`
- D. `CanHop`
- E. `Object`
- F. `Long`

11

Which of the following statements about polymorphism are true? (Choose all that apply.)

- A. A reference to an object may be cast to a subclass of the object without an explicit cast.
- B. If a method takes a class that is the superclass of three different object references, then any of those objects may be passed as a parameter to the method.
- C. A reference to an object may be cast to a superclass of the object without an explicit cast.
- D. All cast exceptions can be detected at compile time.
- E. By defining a public instance method in the superclass, you guarantee that the specific method will be called in the parent class at runtime.

12

Choose the correct statement about the following code:

```
1: public interface Herbivore {  
2:     int amount = 10;  
3:     public static void eatGrass();  
4:     public int chew() {  
5:         return 13;  
6:     }  
7: }
```

- A. It compiles and runs without issue.
- B. The code will not compile because of line 2.
- C. The code will not compile because of line 3.
- D. The code will not compile because of line 4.
- E. The code will not compile because of lines 2 and 3.
- F. The code will not compile because of lines 3 and 4.

13

Which of the following are properties of classes that are properly encapsulated as a JavaBean? (Choose all that apply.)

- A. All instance variables are marked `final`.
- B. `boolean` instance variables are accessed with `is` or `get`.
- C. All instance variables are marked `private`.
- D. They implement the `JavaBean` interface.
- E. Variables are created using lazy instantiation.
- F. The first letter of the any getter/setter, after the `get`, `set`, or `is` prefix, must be uppercase.

14

Which of the following statements about inheritance and object composition are correct? (Choose all that apply.)

- A. Inheritance supports access to protected variables.
- B. Object composition tends to promote greater code reuse than inheritance.
- C. Inheritance relies on the has-a principle.
- D. Object composition supports method overriding at runtime.
- E. Object composition requires a class variable to be declared `public` or accessible from a `public` method to be used by a class in a different package.
- F. Object composition is always preferred to inheritance.

15

Which three scenarios would best benefit from using a singleton pattern? (Choose all three.)

- A. Create read-only objects that are thread-safe.
- B. Manage a reusable cache of objects.
- C. Ensure that all objects are lazily instantiated.
- D. Manage write access to a log file.
- E. Provide central access to application configuration data.
- F. Allow multiple instances of a static object to be managed in memory.

16

Choose the correct statement about the following code:

```
1: public interface CanFly {  
2:     void fly();  
3: }  
4: interface HasWings {  
5:     public abstract Object getWingSpan();  
6: }  
7: abstract class Falcon implements CanFly, HasWings {  
8: }
```

- A. It compiles without issue.
- B. The code will not compile because of line 2.
- C. The code will not compile because of line 4.
- D. The code will not compile because of line 5.
- E. The code will not compile because of lines 2 and 5.
- F. The code will not compile because the class Falcon doesn't implement the interface methods.