1. Which of the following statements about abstract methods/classes in Java are FALSE?
2. Constructors cannot be abstract
3. An abstract class cannot be instantiated
4. Static methods cannot be declared abstract
5. A subclass of an abstract class must define the abstract methods
6. Which of these interfaces are implicitly implemented by arrays?
7. Collection
8. Cloneable
9. Iterable
10. Serializable
11. Which of the following modificators are implicitly set to any interface method?
12. abstract
13. final
14. static
15. public
16. Which of the following modificators are implicitly set to any interface field?
17. abstract
18. final
19. static
20. public
21. What is the output of the following code?

**class** SuperClass {

**public void** method() {

System.out.print("SuperClass");

}

}

**interface** Interface {

**public void** method();

}

**class** ChildClass **extends** SuperClass **implements** Interface {

**public void** method() {

System.out.print("ChildClass");

}

}

**class** Test {

**public static void** main(String[] args) {

**new** ChildClass().method();

((SuperClass)**new** ChildClass()).method();

((Interface)**new** ChildClass()).method();

}

}

1. ChildClassChildClassChildClass
2. ChildClasSuperClassChildClass
3. SuperClassSuperClassChildClass
4. SuperClassChildClassSuperClass
5. Which of the following is/are legal method declaration(s) in abstract class?
6. public abstract void print();
7. public void print();
8. final abstract print();
9. protected abstract void print();
10. private abstract void print();
11. abstract void print();
12. What is the output of the following code?
13. **abstract class** ExmplClass {
14. **public void** doTheThing() {
15. System.out.println("Cool!");
16. }
17. }
18. **class** Test {
19. **public static void** main(String[] args) {
20. (**new** ExmplClass()).doTheThing();
21. }
22. }
23. Compilation error on line 1
24. Compilation error on line 8
25. Outputs string “Cool!”
26. InstantiationException on line 8
27. Which is true?
28. “X extends Y” is correct if and only if X is a class and Y is an interface
29. “X extends Y” is correct if and only if X is an interface and Y is a class
30. “X extends Y” is correct if X and Y are either both classes or both interfaces
31. “X extends Y” is correct for all combinations of X and Y being classes and/or interfaces
32. Pick the INCORRECT statement
33. A class can implement more than one interface
34. A class can extend more than one class
35. An interface can extend more than one interface
36. An interface cannot extend a class
37. A class implements an interface but it does not override all the methods of that interface, then \_\_\_\_\_\_? (in JDK 7)
38. It should be declared as final class
39. It should be declared as abstract class
40. It will successfully compile
41. None of above

(What should be the answer for JDK 8(+)?)

1. Which of the following variables is incorrectly declared?

**interface** Interface {

**int** a = 0;

**public int** b = 1;

**public static int** c = 2;

**public final int** d = 3;

**public static final int** e = 4;

}

1. a
2. b
3. c
4. d
5. e
6. all correct
7. Is the following interface valid?

public interface MyWonderfulInterface {}

1. Yes
2. No

13) Which of the following statements is true?

a). A class can extend multiple base classes.

b). You can implement only one interface since java does not support multiple inheritance.

c). You can implement multiple interfaces.

d). You can either extend a class or implement an interface (but not both) at a time.

14) Consider the following three classes: University, Department, and CS\_Department. The University and Department classes are related with relation r1, and the Department and CS\_Department classes are related with relation r2. Which combination of these relations is appropriate?

a). r1: inheritance, r2: inheritance

b). r1: composition, r2: inheritance

c). r1: inheritance, r2: composition

d). r1:composition,r2:composition

Answer: b. (a university has many departments, so they share a has-a relationship between them, a composition relationship. CS\_Department is a department, so these two share a is-a relationship between them, an inheritance relationship.)

15). You need to model a file system where there could be subfolders and files in a folder. What is the most appropriate design choice in this case to represent the relationship between Folder and File classes?

a). Use composition to model the relationship of “a Folder object consists of File objects.”

b). Use composition to model the relationship of “a Folder object consists of File objects or Folder objects.”

c). Use inheritance to define a superclass (say FolderItem) and make Folder and File classes subclasses to this class. Use composition to model the relationship “a Folder object consists of FolderItem objects.”

d). Use inheritance between Folder and File classes to model the relationship “a Folder is of type File.”

16) Pick the correct statement(s)

a). abstract class can declare and define constructor

b). abstract class can be final

c). abstract class can declare and define static methods

d). it is necessary for abstract class to have abstract method

e). abstract class can contain main method

17) Given:

**interface** Machine { }

**interface** Engine { }

**abstract interface** Tractor **extends** Machine, Engine {

void pullStuff();

}

**class** Deere **implements** Tractor {

**public void** pullStuff() { System.out.print("pulling ");}

}

**class** LT255 **implements** Tractor **extends** Deere {

**public void** pullStuff() {

System.out.print("pulling harder ");

}

}

**public class** LT155 **extends** Deere **implements** Tractor, Engine { }

What is the result? (Choose all that apply.)

A. Compilation succeeds.

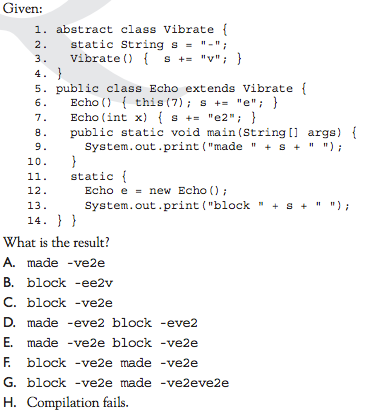
B. Compilation fails because of error(s) in Tractor.

C. Compilation fails because of error(s) in Deere.

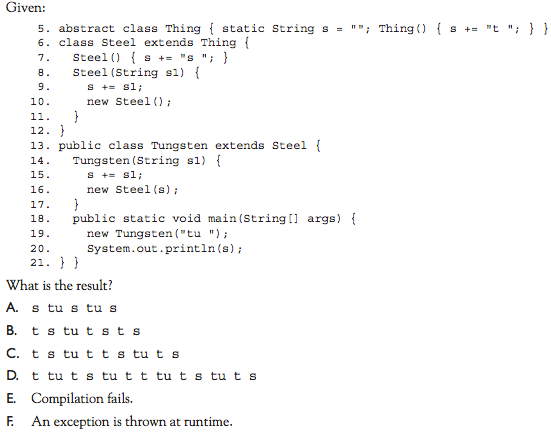
D. Compilation fails because of error(s) in LT255.

E. Compilation fails because of error(s) in LT155.

18)



19)



20)

