![socrative]

OCA Practice Questions Ch. 1 ('Copy')

1. Which of the following method signatures is a valid declaration of an entry point in a Java application?

A public void main(String[] args)

B public static void main()

C private static void start(String[] mydata)

✓ **D** public static final void main(String[] mydata)

> **i** An entry point in a Java application consists of a main() method with a single String[] argument, return type of void, and modifiers public and static. The name of the variable in the input argument does not matter. Option A is missing the static modifier, Option B is missing the String[] argument, and Option C has the wrong access modifier and method name. Only Option D fulfills these requirements. Note that the modifier final is optional and may be added to an entry point method.

2. The following class diagram demonstrates the relationship between Gold and Silver, which extend the Metal class. Assume the attributes are all declared public. Which statement about the diagram is not true?

✓ **A** The diagram demonstrates platform independence in Java.

B The diagram demonstrates object-oriented design in Java.

C The Gold and Silver classes inherit weight and color attributes from the Metal class.

D Gold does not inherit the luster attribute.
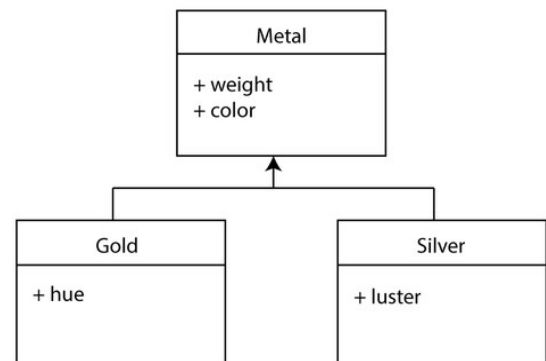
> **i** The diagram is an example of object-oriented design in Java, making Option B a true statement. Options C and D are also true, as they follow from the inheritance model in the diagram. Option A is the correct answer, since platform independence has nothing to do with the diagram.

3. What is the proper filename extension for a Java bytecode compiled file?

A .java

B .bytecode

✓ **C** .class

D .dll

> **i** The proper extension for a Java compiled bytecode file is .class, making Option C the correct answer.

4.
```
1: import java.util.*;
2: import java.sql.*;
3: public class BirthdayManager {
4:     private Date rob = new Date();
5:     private java.util.Date sharon = new java.util.Date();
6: }
```

Given that a class exists in both the and packages, what is the result of compiling the following class?

A   The code does not compile because of lines 1 and 2.

✓ B   The code does not compile because of line 4.

C   The code does not compile because of line 5.

D   The code compiles without issue.

ℹ The fact that the Date class exists in both packages does not impact the ability to import both packages, so lines 1 and 2 compile, without issue and Option A is incorrect. Line 4 will not compile because the Date class used is ambiguous, making Option B correct and Option D incorrect. Finally, Option C is incorrect because line 5 does compile, as the fully qualified name of the class is used.

5. Which of the following is not a facet of traditional object-oriented programming languages?

✓ A   Objects are grouped as procedures, separate from the data they act on.

B   An object can take many forms via casting.

C   An object can hold data, referred to as attributes.

D   An object can perform actions, via methods.

ℹ Options B, C, and D are each attributes of traditional object-oriented programming. Option A is incorrect as an object-oriented project tends to group data and the actions related to that data into a single object.

6. Which variables have a scope limited to a method?

A   Interface variables

B   Class variables

C   Instance variables

✓ D   Local variables

ℹ Only local variables have such a small scope, making Option D the correct answer.

7. Which package is imported into every Java class by default?

A   java.util

✓ B   java.lang

C   system.lang

D   java.system

ℹ The package java.lang is imported into every Java class, so Option B is correct. The other options must be explicitly imported. Option A exists but must be explicitly imported. Options C and D do not exist in the

8. Which of the following is not a valid code comment in Java?

A // Add 5 to the result

B /*** TODO: Fix bug 12312 ***/

✓ **C** # Add configuration value

D /* Read file from system ****/

ⓘ Java accepts Options A, B, and D as valid comments. Note that the /* */ syntax can have additional (and uneven) star (*) characters as shown in B and D. Option C is incorrect as hashtag (#) is not a valid comment character in Java

9. Which statement about a valid .java file is true?

A It can only contain one class declaration.

B It can contain one public class declaration and one public interface definition.

C It must define at least one public class.

✓ **D** It may define at most one public class.

ⓘ A valid .java file may define any number of classes or interfaces but have at most one public class. It can also not define any public classes. For these reasons, Option A, B, and C are incorrect, leaving Option D as the only correct answer.

10.
```
package competition;
public class Robot {
    static String weight = "A lot";
    /* default */ double ageMonths = 5, ageDays = 2;
    private static boolean success = true;

    public void main(String[] args) {
        final String retries = "1";
        // P1
    }
}
```

Given the following application, fill in the missing values in the table starting from the top and going downward.

Variable Type	Number of Variables Accessible at P1
Class	____
Instance	____
Local	____

A 2, 0, 1

✓ **B** 2, 2, 1

C 1, 0, 1

D 0, 2, 1

ⓘ Notice in this question that main() is not a static method, therefore it can access both class and instance

variables. Since there are two class variables and two instance variables defined, Option B is the correct answer.

11. Which statement about import statements is true?

A The class will not compile if it contains unused import statements.

✓ **B** Unused import statements can be removed from the class without causing a class to become unable to be compiled.

C The class will not compile if a duplicate import statement is present.

D If a class contains an import statement for a class used in the program that cannot be found, it can still compile.

ⓘ A class will compile if it has unused or redundant import statements, making Option A and C incorrect. Option D is also incorrect as the compiler must be able to locate the class of the import statement. The correct answer is Option B. Removing unused import statements does not cause a class to become unable to be compiled.

12.
```
1: public class ParkRanger {
2:     int birds = 10;
3:     public static void main(String[] data) {
4:         int trees = 5;
5:         System.out.print(trees+birds);
6:     }
7: }
```

What is the result of compiling and executing the following class?

✓ **A** It does not compile.

B It compiles but throws an exception at runtime.

C It compiles and outputs 5.

D It compiles and outputs 15.

ⓘ The code does not compile because of line 5, making Option A the correct answer. For this question, it helps to understand variable scope. The main() method is static and does not have access to any class instance variables. The birds variable is not static and requires a class instance variable to access. Therefore, the code does not compile when the static method attempts to access a non-static variable without an instance of the class.

13. Which statements about Java are true?
I. The java command can execute .java and .class files.
II. Java is not object oriented.
III. The javac command compiles directly into native machine code.

A I only

B III only

C II and III

✓ **D** None are true.

ⓘ The java command can only execute compiled .class files, so I is false. Java is most certainly object

oriented, one of the key design principles, so II is also false. The javac command compiles into bytecode, which must be run in a Java virtual machine (JVM), and is not native machine code, so III is false as well. Since none of the statements are true, Option D is the correct answer.

14. Which of the following lines of code is not allowed as the first line of a Java class file?

A   import widget.*;

B   // Widget Manager

C   package sprockets;

✓ D   int facilityNumber;

ⓘ   A class can start with a comment, an optional package statement, or an import statement if there is no package statement. It cannot start with a variable definition, making Option D the correct answer.

15. Which one of the following statements is true about using packages to organize your code in Java?

A   Every class is required to include a package declaration.

B   To create a new package, you need to add a package.init file to the directory.

✓ C   Packages allow you to limit access to classes, methods, or data from classes outside the package.

D   It is not possible to restrict access to objects and methods within a package.

ⓘ   Classes may be defined without a package declaration and are placed in the default package, so Option A is incorrect. Option B is a completely false statement as no such file is required in Java. Option C is correct as it is one of the primary reasons for organizing your application into packages. Option D is incorrect as package-private allows access to methods and variables to be limited to those classes within the same package.

16. Given that the current directory is /user/home, with an application Java file in /user/home/Manager.java that uses the default package, which are the correct commands to compile and run the application in Java?

A   javac Manager
     java Manager

✓ B   javac Manager.java
     java Manager

C   javac Manager
     java Manager.class

D   javac Manager.java
     java Manager.class

ⓘ   The compilation command requires the full or relative name of the file, including the .java extension, making Options A and C incorrect. The execution command requires the class name without a filename extension, making Option D incorrect. Option B is the only correct set of compilation and execution commands.

17. Structuring a Java class such that only methods within the class can access its instance variables is referred to as _____.

A platform independence

B object orientation

C inheritance

✓ D encapsulation

> ℹ Encapsulation is the technique of removing access to a class's instance variables from processes outside the class, making Option D the correct answer.

18.
```
String tree = "pine";
int count = 0;
if (tree.equals("pine")) {
    int height = 55;
    count = count + 1;
}
System.out.print(height + count);
```

What is the output of the following code snippet?

A 1

B 55

C 56

✓ D It does not compile.

> ℹ The height variable is declared within the if-then statement block. Therefore, it cannot be referenced outside the if-then statement and the code does not compile.

19. Which of the following is true of a Java bytecode file?

✓ A It can be run on any computer with a compatible JVM.

B It can only be executed on the same type of computer that it was created on.

C It can be easily read and modified in a standard text editor.

D It requires the corresponding .java that created it to execute.

> ℹ A Java bytecode file is a binary encoded set of instructions designed to be run on any computer with a compatible JVM, making Option A the only correct answer. By compatible JVM, we mean one capable of running the class file. For example, a Java 6 JVM may have trouble executing a Java 8 compiled file. Option B is incorrect, and is more a facet of machine language compiled programming classes. Option C is also incorrect as binary data is not particularly human readable. Finally, Option D is incorrect as the compiled file can be distributed without its .java source file and execute without issue.

20. What is the correct character for terminating a statement in Java?

A A colon (:)

B An end-of-line character

C A tab character

✓ **D** A semicolon (;)

> ℹ Unlike with some other programming languages, the proper way to terminate a line of code is with a semicolon (;), making D the only correct answer.

21.
```
1: public class Tolls {
2:     private static int yesterday = 1;
3:     int tomorrow = 10;
4:     public static void main(String[] args) {
5:         Tolls tolls = new Tolls();
6:         int today=20, tomorrow = 40;
7:         System.out.print(today + tolls.tomorrow + tolls.yesterday);
8:     }
9: }
```

What is the result of compiling and executing the following class?

A The code does not compile due to line 6.

B The code does not compile due to line 7.

✓ **C** 31

D 61

> ℹ The code compiles and runs without issue, so Options A and B are incorrect. The question relies on your ability to understand variable scope. The variable today has local scope to the method in which it is executed. The variable tomorrow is re-declared in the method, but the reference used on line 7 is to the instance variable with a value of 10. Finally, the variable tomorrow is static. While using an instance reference to access a static variable is not recommended, it does not prevent the variable from being read. The result is line 7 evaluates and prints (20 + 10 + 1) = 31, making C the correct answer.

22.
```
1: public ThisClassDoesNotCompile {
2:     double int count;
3:     void errors() {}
4:     static void private limit; }
```

Given the following class definition, which is the only line that does not contain a compilation error?

A Line 1

B Line 2

✓ **C** Line 3

D Line 4

> ℹ Line 1 is missing the class keyword. Line 2 contains two types for the same variable. Line 3 is a valid definition for a method, making C the correct answer. Finally, line 4 contains an access modifier, private, after the return type, which is not allowed. In addition, void is an invalid type for variables.

23. Which of the following features allows a Java class to be run on a wide variety of computers and devices?

A Encapsulation

B Object oriented

C Inheritance

✓ D Platform independence

> ℹ Platform independence is the property of Java that allows it to be run on a variety of different devices.

24. Which of the following is not a property of a JVM?

✓ A It prevents Java bytecode from being easily decoded/decompiled.

B It supports platform independence.

C It manages memory for the application.

D It translates Java instructions to machine instructions.

> ℹ Options B, C, and D are each correct statements about JVMs. Option A is incorrect. Not only is it not a statement about JVMs, it is actually false as Java bytecode can often be easily decoded/decompiled.

25. Which of the following variables are always in scope for the entire program?

A Package variables

✓ B Class variables

C Instance variables

D Local variables

> ℹ There is no such thing as package variables, so Option A is incorrect. Option C is incorrect as the variable is only in scope within a specific instance of the class. Option D is also incorrect as the variable is only in scope for a single method that it is defined in. Option B is the only correct answer as class variables are in scope within the program.

26.

```
import television.actor.*;
import movie.director.*;
```

Given the following wildcard import statements, which class would be included in the import?

A television.actor.recurring.Marie

B movie.directors.John

✓ C television.actor.Package

D movie.NewRelease

> ℹ Option A is incorrect as the sub-package recurring is not included by the import statements. Option B is also incorrect as it uses the plural directors instead of the singular director used in the import statements.

Option D is incorrect as the wildcard is applied to the sub-package movie.director, not the package movie. Finally, Option C is correct as it is a valid class accessible from the wildcard import.

27. Which is the correct order of statements for a Java class file?

A import statements, package statement, class declaration

B package statement, class declaration, import statement

C class declaration, import statements, package declaration

✓ D package statement, import statements, class declaration

> i Java classes are defined in this order: package statement, import statements, class declaration, making Option D the only correct answer. Note that not all of these statements are required. For example, a class may not have a package statement, but if it does, it must come first in the file.

28.
```
package planetarium;
import java.lang.*;
import stars.*;
import java.lang.Object;
import stars.Blackhole;
public class Observer {
    public void find(Blackhole blackhole)
}
```

Given the following class definition, what is the maximum number of import statements that can be discarded and still have the code compile? For this question, assume that the Blackhole class is defined only in the stars package.

A Zero

B One

C Two

✓ D Three

> i The import statements for stars.* and stars.Blackhole are redundant import statements, since only the class Blackhole is used, and therefore one of them can be safely removed. The import statements java.lang.* and java.lang.Object are both not required as java.lang is automatically imported in every Java class. Therefore, three of the four import statements can be safely removed, making the correct answer Option D.

29.

```java
package forest;
public class Deer {
    public static void main(String... deerParams) {
        System.out.print(deerParams[2]);
    }
}
```

Given the following class definition, which command will cause the application to output the message White-tailed?

A   java forest.Deer deer 5 "White-tailed deer"

B   java forest.Deer "White-tailed deer" deer 3

✓ C   java forest.Deer Red deer White-tailed deer

D   java forest.Deer My "deer White-tailed"

> **i**   The application prints the third argument of the input methods. Note that double quotes "" group input arguments. Therefore, the third argument of Option A is White-tailed deer. The third argument of Option B is 3. The third argument of Option C is White-tailed, making it the correct answer. Finally, Option D only has two input arguments, leading to an ArrayIndexOutOfBoundsException trying to read the third argument at runtime.

30. Which of the following is a true statement?

A   The java command compiles a .java file into a .class file.

✓ B   The javac command compiles a .java file into a .class file.

C   The java command compiles a .class file into a .java file.

D   The javac command compiles a .class file into a .java file.

> **i**   The javac command compiles a .java file into a .class bytecode file, making Option B the correct answer.

31. Which of the following statements about Java is true?

A   Java is a procedural programming language.

✓ B   Java allows method overloading.

C   Java allows operator overloading.

D   Java allows direct access to objects in memory.

> **i**   Java is object oriented, not procedural, so Option A is a false statement. Java allows method overloading in subclasses, so Option B is correct. Operator overloading is permitted in languages like C++, not Java, so Option C is also untrue. Finally, Option D is not a true statement as the JVM manages the location of objects in memory that can change and is transparent to the Java application.

32.

```
_____ agent;
public _____ Banker {
    private static _____ getMaxWithdrawal() {
        return 10;
    }
}
```

Given the following code, what values inserted in order into the blank lines, allow the code to compile?

A import, class, null

B import, interface, void

C package, int, int

✓ D package, class, long

> ℹ Option A is incorrect as the return type of the method cannot be null. Option B is also incorrect as the return type cannot be void if the method uses a return statement. Option C is incorrect too as the class keyword is replaced with int. Option D is correct because it's the only answer that allows the code to compile without issue. Note that other values are possible for this question. For example, either int or long can be entered in the last blank. The key here is that only one of the available answer choices allows the code to compile.

33. What is the output of the following application?

✓ A 2 5

B 8 5

C 6 5

D The code does not compile.

```
public class Airplane {
    static int start = 2;
    final int end;
    public Airplane(int x) {
        x = 4;
        end = x;
    }
    public void fly(int distance) {
        System.out.print(end-start+" ");
        System.out.print(distance);
    }
    public static void main(String... start) {
        new Airplane(10).fly(5);
    }
}
```

> ℹ The code compiles so Option D is incorrect. The input to the constructor is ignored, making the assignment of end to be 4. Since start is 2, the subtraction of 4 by 2 results in the application printing 2, followed by 5, making Option A the correct answer.

34. What is one of the most important reasons that Java supports extending classes via inheritance?

A Inheritance requires that a class that extends another class be in the same package.

B The program must spend extra time/resources at runtime jumping through multiple layers of inheritance to determine precise methods and variables.

C Method signature changes in parent classes may break subclasses that use overloaded methods.

✓ D Developers minimize duplicate code in new classes by sharing code in a common parent class.

> ℹ Option A is a false statement, while Options B and C are actually arguments against using inheritance. Option D is one of the most important reasons Java supports inheritance, to allow increased code reuse among classes.

35. Which of the following is a valid code comment in Java?

✓ **A** //////// Walk my dog

  B #! Go team!

  C / Process fails at runtime /

  D None of the above

> **i** The double slash (//) syntax can have any number of slashes as a comment, so long as it starts with two of them, making Option A the correct answer. The (#) is not a comment character in Java, regardless of whether it is followed by a (!), so Option B is incorrect. Option C is incorrect as a single slash (/) is not a valid comment in Java. Finally, Option D is incorrect as Option A is a valid comment.

36. Which of the following method signatures is not a valid declaration of an entry point in a Java application?

  A public static void main(String... arguments)

✓ **B** public static void main(String arguments)

  C public static final void main(String[] arguments)

  D public static void main(String[] arguments)

> **i** An entry point in a Java application consists of a main() method with a single String[] argument, return type of void, and modifiers public and static. Option D is the typical syntax for this method, although Options A and C are also valid forms of this method. Note that the modifier final is optional and may be added to the method signature. Furthermore, the main() method may take a vararg or array. Option B is the only invalid declaration as it does not take an array as an argument.

37. Given the file Magnet.java below, which of the marked lines can you independently insert the line public String color; into and still have the code compile?

  A a1 and a3

✓ **B** a2 and a4

  C a2, a3, and a4

  D a1, a2, a3, and a4

```
// line a1
public class Magnet {
    // line a2
    public void attach() {
        // line a3
    }
    // line a4
}
```

> **i** The line of code cannot be inserted at a1 because no variables are allowed outside of the class declaration in this file, making Options A and D incorrect. The line of code can also not be inserted at a3 as local variables defined within methods cannot have access modifiers such as public, making Option C incorrect. The code can be inserted independently at a2 and a4 as instance variables can be defined anywhere in the class outside a method. Therefore, Option B is the correct choice.

38. What is required to define a valid Java class file?

✓ **A** A class declaration

  B A package statement

  C At least one import statement

  D The public modifier

> ℹ Option A is the only correct answer as a class definition is the only required component in a Java class file. Note that we said a Java class file here; Java also allows interfaces and enums to be defined in a file. A package statement and import statements are optional for declaring a class, making Options B and C incorrect. A class may also be defined with package-level access in a file, making Option D an incorrect answer.

39. What is the proper filename extension for a Java source file?

  A .jav

  B .class

  C .source

✓ **D** .java

> ℹ The proper extension for a Java compiled bytecode file is .java, making Option D the correct answer.

40.
```
1: package pocket;
2: import pocket.complex.*;
3: import java.util.*;
4: public class Calculator {
5:    public static void main(String[] args) {
6:       System.out.print(Math.floor(5));
7:    }
8: }
```

Given that a Math class exists in both the java.lang and pocket.complex packages, what is the result of compiling the following class?

  A The code does not compile because of line 2.

  B The code does not compile because of line 3.

✓ **C** The code does not compile because of line 6.

  D The code compiles without issue.

> ℹ Remember that java.lang is automatically imported in all Java classes, therefore both java.lang.Math and pocket.complex.Math are both imported into this class. Importing both sets of packages does not cause any compilation issues, making Option A incorrect. Line 3 is unnecessary import but including it does not prevent the class from compiling, making Option B incorrect. While both versions of Math may be imported into the class, the usage of the Math class requires a package name. Because of this, line 6 does not compile as the class reference is ambiguous, making Option C the correct answer and Option D incorrect.

41.

```
import dog.*;
import dog.puppy.*;
```

Given a class that uses the following import statements, which class would not be automatically accessible within the class without using its full package name?

✓ **A** dog.puppy.female.KC

B dog.puppy.Georgette

C dog.Webby

D java.lang.Object

ⓘ Options B and C are accessible within the class as they are covered by the import statements. Option D is also fine as java.lang.Object is available without an explicit import. The only class not automatically accessible within the class without the full package name is dog.puppy.female.KC as the import statements do not include sub-packages; therefore, Option A is the correct answer.

42. _____ is the technique of structuring programming data as a unit consisting of attributes, with actions defined on the unit.

A Encapsulation

✓ **B** Object orientation

C Platform independence

D Polymorphism

ⓘ Object-oriented programming is the technique of structuring data into objects, which may contain data and a set of actions that operate on the data, making Option B the correct answer.

43.
```
package food;
import food.vegetables.*;
import food.fruit.*;
import java.util.Date;

public class Grocery {
    Apple a; Broccoli b;  Date c;
}
```

Given the following class definition, what is the maximum number of import statements that can be discarded and still have the code compile? For this question, assume that the Broccoli class is in the food.vegetables package, and the Apple class is the food.fruit package.

✓ **A** 0

B 1

C 2

D 3

ⓘ All of the import statements in this class are required. Removing any of them would cause the class to not

44.
```
public class Keyboard {
    private boolean numLock = true;
    static boolean capLock = false;
    public static void main(String... shortcuts) {
        System.out.print(numLock+" "+capLock);
    }
}
```

Given the following application, what is the expected output?

A true false

B false false

✓ **C** It does not compile.

D It compiles but throws an exception at runtime.

ⓘ The numLock variable is not accessible in the static main() method without an instance of the Keyboard class; therefore, the code does not compile, and Option C is the correct answer.

45.
```
public class RollerSkates {
    static int wheels = 1;
    int tracks = 5;
    public static void main(String[] arguments) {
        RollerSkates s = new RollerSkates();
        int feet=4, tracks = 15;
        System.out.print(feet + tracks + s.wheels);
    }
}
```

What is the result of compiling and executing the following class?

A The code does not compile.

B 5

C 10

✓ **D** 20

ⓘ The code compiles and runs without issue, so Option A is incorrect. The question involves understanding the value and scope of each variable at the print() statement. The variables feet and tracks are locally scoped and set to 4 and 15, respectively, ignoring the value of tracks of 5 in the instance of the class. Finally, the static variable s.wheels has a value of 1. The result is the combined value is 20, making Option D the correct answer.

46.
```
package sports;
public class Bicycle {
    String color = "red";
    private void printColor(String color) {
        color = "purple";
        System.out.print(color);
    }
    public static void main(String[] rider) {
        new Bicycle().printColor("blue");
    }
}
```

What is the result of compiling and executing the following class?

A red

✓ **B** purple

C blue

D It does not compile.

ⓘ First off, the color variable defined in the instance and set to red is ignored in the method printColor() as local scope overrides instance scope, so Option A is incorrect. The value of color passed to the printColor() method is blue, but that is lost by the assignment to purple, making Option B the correct answer and Option C incorrect. Option D is incorrect as the code compiles and runs without issue.

47. Which statements about calling the compilation command javac and the execution command java are true?
I. java may use a period . to separate packages.
II. javac takes a .java file and returns a .class file.
III. java may use a slash (/) to separate packages.

A I only

B II only

✓ **C** I and II

D I, II, and III

ⓘ The javac command takes a text-based .java file and returns a binary bytecode .class file, making II a true statement. The java command uses a period (.) to separate packages, not a slash (/), making I a true statement and III a false statement. For these reasons, Option C is the correct answer.

48.
```
package forecast;
public class Weather {
    private static boolean heatWave = true;
    public static void main() {
        boolean heatWave = false;
        System.out.print(heatWave);
    }
}
```

What is the result of compiling and executing the following application?

A true

B false

C It does not compile.

✓ **D** It compiles but throws an error at runtime.

> ⓘ The application compiles without issue, so Option C is incorrect. The application does not execute though, as the main() method does not have the correct method signature. It is missing the required input argument, an array of String. Trying to execute the application without a proper entry point produces an error, making Option D the correct answer.

49. Given the following class diagram, which Java implementation most closely matches this structure?

Book
+ numberOfPages
+ getRating()

A public class Book { public int numOfPages;

B public class Book { public String getRating() {return null;} }

✓ **C** public class Book { public int numberOfPages; public String getRating() {return null;} }

D public class Book { void numberOfPages; }
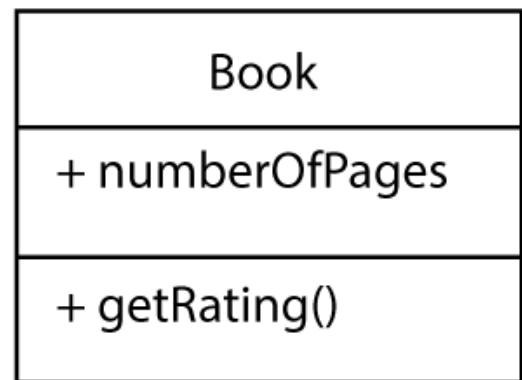
> ⓘ Option A does not compile because it is missing the closing bracket for the class. Option D does also not compile as void is not a valid type for a variable. Regardless, Options A and D are incorrect as they are missing the getRating() method. Note that Option A also uses an abbreviation for numberOfPages. Option B is incorrect as it is missing the numberOfPages attribute. Option C is the correct answer as it properly defines the attribute numberOfPages and method getRating().

50. Which statement about the JVM is true?

A The JVM schedules garbage collection on a predictable schedule.

B The JVM ensures that the application will always terminate.

✓ **C** The JVM requires a properly defined entry point method to execute the application.

D A Java compiled code can be run on any computer.

> ⓘ Garbage collection can happen at any time while an application is running, especially if the available memory suddenly becomes low, making Option A incorrect. Option B is also incorrect, since it is trivial to create a Java application with an infinite loop that never terminates. Option D is incorrect because the computer must be able to run the JVM in order to execute a Java class. Option C is the only correct answer, as the JVM does require an entry point method to begin executing the application.