



OCA / OCP Java SE 8 Programmer Practice Tests

PREV
Chapter 4 Creating and Using Arrays

↩

AA

⌵

🔍

Chapter 6 Working with Methods and Encapsulation

NEXT
▶

Chapter 5 Using Loop Constructs

THE OCA EXAM TOPICS COVERED IN THIS PRACTICE
TEST INCLUDE THE FOLLOWING:

- ✓ Using Loop Constructs
 - Create and use while loops
 - Create and use for loops including the enhanced for loop
 - Create and use do/while loops
 - Compare loop constructs
 - Use break and continue
1. Which type of loop is best known for its boolean condition that controls entry to the loop?
 1. do-while loop
 2. for (traditional)
 3. for-each
 4. while
 2. Which type of loop is best known for using an index or counter?
 1. do-while loop
 2. for (traditional)
 3. for-each
 4. while
 3. Which type of loop is guaranteed to have the body execute at least once?
 1. do-while loop
 2. for (traditional)
 3. for-each
 4. while
 4. Which of the following can loop through an array without referring to the elements by index?
 1. do-while loop
 2. for (traditional)
 3. for-each
 4. while

You have 2 days left in your trial, Gtucker716. Subscribe today. [See pricing options.](#)

1. break
 2. continue
 3. end
 4. skip
6. What keyword is used to proceed with execution immediately after a loop?
1. break
 2. continue
 3. end
 4. skip
7. Which type of loop has three segments within parentheses?
1. do-while loop
 2. for (traditional)
 3. for-each
 4. while
8. Which of the following statements is/are true?
1. A traditional for loop can iterate through an array starting from index 0.
 2. A traditional for loop can iterate through an array starting from the end.
1. Only I
 2. Only II
 3. Both statements
 4. Neither statement
9. Which of the following statements is/are true?
1. A for-each loop can iterate through an array starting from index 0.
 2. A for-each loop can iterate through an array starting from the end.
1. Only I
 2. Only II
 3. Both statements
 4. Neither statement
10. Which type of loop has a boolean condition that is first checked after a single iteration through the loop?
1. do-while loop
 2. for (traditional)
 3. for-each
 4. while
11. What does the following code output?
-
- ```
int singer = 0;
while (singer)
 System.out.println(singer++);
```
- 
1. 0
  2. The code does not compile.
  3. The loops complete with no output.
  4. This is an infinite loop.

12. What does the following code output?

---

```
List<String> drinks = Arrays.asList("can", "cup");
for (int container = drinks.size() - 1; container >= 0; container--)
 System.out.print(drinks.get(container) + ",");
```

---

- 
1. can, cup,
  2. cup, can,
  3. The code does not compile.
  4. None of the above

13. What does the following code output?

---

```
public static void main(String[] args) {
 List<String> bottles = Arrays.asList("glass", "plastic");
 for (int type = 0; type < bottles.size();) {
 System.out.print(bottles.get(type) + ",");
 break;
 }
 System.out.print("end");
}
```

---

1. glass,end
2. glass,plastic,end
3. The code does not compile.
4. None of the above

14. What does the following code output?

---

```
String letters = "";
while (letters.length() != 2)
 letters+="a";
System.out.println(letters);
```

---

1. aa
2. aaa
3. The loops complete with no output.
4. This is an infinite loop.

15. What is the result of the following when run with java  
peregrine.TimeLoop September 3 1940?

---

```
package peregrine;
public class TimeLoop {
 public static void main(String[] args) {
 for (int i = args.length; i>=0; i++)
 System.out.println("args");
 }
}
```

---

1. args
2. argsargs
3. The code does not compile.
4. None of the above

16. What is the output of the following code?

---

```
package chicago;
public class Loop {
 private static int count;
 private static String[] stops = new String[] { "Washington",
 "Monroe", "Jackson", "LaSalle" };
 public static void main(String[] args) {
 while (count < stops.length) {
 if (stops[count++].length() < 8) {
 break;
 }
 }
 System.out.println(count);
 }
}
```

---

1. 1
2. 2
3. 4
4. The code does not compile.

17. What is the result of the following code?

```
do {
 int count = 0;
 do {
 count++;
 } while (count < 2);
 break;
} while (true);
System.out.println(count);
```

1. 2
2. 3
3. The code does not compile.
4. This is an infinite loop.

18. Which of the following segments of a for loop can be left blank?

```
for (segmentA; segmentB; segmentC) {
}
```

1. segmentA
2. segmentB
3. segmentC
4. All of the above

19. How many of the loop types (while, do while, traditional for, and enhanced for) allow you to write code that creates an infinite loop?

1. One
2. Two
3. Three
4. Four

20. What is the output of the following?

```
List<String> drinks = Arrays.asList("can", "cup");
for (int container = 0; container < drinks.size(); container++)
 System.out.print(drinks.get(container) + ",");
```

1. can, cup,
2. cup, can,
3. The code does not compile.
4. None of the above

21. What happens when running the following code?

```
do {
 System.out.println("helium");
} while (false);
```

1. It completes successfully without output.
2. It outputs helium once.
3. It keeps outputting helium.
4. The code does not compile.

22. Which of the following is equivalent to this code snippet given an array of String objects?

```
for (int i=0; i<fun.length; i++)
 System.out.println(fun[i]);
```

1. for (String f = fun) System.out.println(f);
2. for (String f : fun) System.out.println(f);
3. for (String = fun) System.out.println(it);
4. None of the above

23. How many of these statements can be inserted after the `println` to have the code flow follow the arrow in this diagram?

---

```
break;
break letters;
break numbers;
```

---

```
 }
}
```

1. None
2. One
3. Two
4. Three

24. Using the diagram in the previous question, how many of these statements can be inserted after the `println` to have the code flow follow the arrow in the diagram?

---

```
continue;
continue letters;
continue numbers;
```

---

1. None
2. One
3. Two
4. Three

25. What does the following code output?

---

```
int singer = 0;
while (singer > 0)
 System.out.println(singer++);
```

---

1. 0
2. The code does not compile.
3. The loops completes with no output.
4. This is an infinite loop.

26. Which of the following types is `taxis` not allowed to be in order for this code to compile?

---

```
for (Object obj : taxis) {
}
```

---

1. `ArrayList<Integer>`
2. `int[]`
3. `StringBuilder`
4. All of these are allowed.

27. What is the output of the following?

---

```
boolean balloonInflated = false;
do {
 if (!balloonInflated) {
 balloonInflated = true;
 System.out.print("inflate-");
 }
} while (!balloonInflated);
System.out.println("done");
```

---

1. done
2. inflate-done
3. The code does not compile.
4. This is an infinite loop.

28. What does the following code output?

---

```
String letters = "";
while (letters.length() != 3)
 letters+="ab";
System.out.println(letters);
```

---

1. ab
  2. abab
  3. The loop completes with no output.
  4. This is an infinite loop.
29. What describes the order in which the three expressions appear in a for loop?
1. boolean conditional, initialization expression, update statement
  2. initialization expression, boolean conditional, update statement
  3. initialization expression, update statement, boolean conditional
  4. None of the above
30. What is the result of the following?

---

```
int count = 10;
List<Character> chars = new ArrayList<>();
do {
 chars.add('a');
 for (Character x : chars) count -=1;
} while (count > 0);
System.out.println(chars.size());
```

---

1. 3
2. 4
3. The code does not compile.
4. None of the above

31. What is the result of the following?

---

```
int k = 0;
for (int i = 10; i > 0; i--) {
 while (i > 3) i -= 3;
 k += 1;
}
System.out.println(k);
```

---

1. 1
2. 2
3. 3
4. 4

32. Which of the following is equivalent to this code snippet given an array of String objects?

---

```
for (int i=fun.length-1; i>=0; i--)
 System.out.println(fun[i]);
```

---

1. for (String f = fun) System.out.println(f);
2. for (String f : fun) System.out.println(f);
3. for (String f fun) System.out.println(it);
4. None of the above

33. What does the following code output?

---

```
public static void main(String[] args) {
 List<String> bottles = Arrays.asList("glass", "plastic");
 for (int type = 0; type < bottles.size();)
 System.out.print(bottles.get(type) + ",");
 break;
 System.out.print("end");
}
```

---

1. glass,end
2. glass,plastic,end
3. The code does not compile.

4. None of the above

34. What is the result of the following?

```
String[] nycTourLoops = new String[] { "Downtown", "Uptown", "Brook"
String[] times = new String[] { "Day", "Night" };
for (int i = 0, j = 0; i < nycTourLoops.length
 && j < times.length; i++; j++)
{
 System.out.print(nycTourLoops[i] + " " + times[j] + "-");
}
```

1. Downtown Day-
2. Downtown Day-Uptown Night-
3. The code does not compile.
4. The code compiles but throws an exception at runtime.

35. What is the result of the following when run with `java peregrine.TimeLoop September 3 1940`?

```
package peregrine;
public class TimeLoop {
 public static void main(String[] args) {
 for (int i = args.length; i>0; i--)
 System.out.println(args[i]);
 }
}
```

1. September
2. 1940
3. The code does not compile.
4. None of the above

36. What is the output of the following?

```
public class Shoelaces {
 public static void main(String[] args) {
 String tie = null;
 while (tie == null)
 tie = "shoelace";
 System.out.print(tie);
 }
}
```

1. null
2. shoelace
3. shoelaceshoelace
4. None of the above

37. The following code outputs a single letter x. What happens if you remove lines 25 and 28?

```
23: String race = "";
24: loop:
25: do {
26: race += "x";
27: break loop;
28: } while (true);
29: System.out.println(race);
```

1. It prints an empty string.
2. It still outputs a single letter x.
3. It no longer compiles.
4. It becomes an infinite loop.

38. What is the output of the following code?

```
package chicago;
public class Loop {
 private static int count;
 private static String[] stops = new String[] { "Washington",
 "Monroe", "Jackson", "LaSalle" };
 public static void main(String[] args) {
 while (count < stops.length) {
```

```

 if (stops[count++].length() < 8) {
 continue;
 }
 }
 System.out.println(count);
}

```

1. 1
2. 2
3. 4
4. The code does not compile.

39. What is the output of the following?

```

StringBuilder builder = new StringBuilder();
String str = new String("Leaves growing");
do {
 System.out.println(str);
} while (builder);
System.out.println(builder);

```

1. Leaves growing
2. This is an infinite loop.
3. The code does not compile.
4. The code compiles but throws an exception at runtime.

40. What is the result of the following code?

```

6: int count = 0;
7: do {
8: do {
9: count++;
10: } while (count < 2);
11: break;
12: } while (true);
13: System.out.println(count);

```

1. 2
2. 3
3. The code does not compile.
4. This is an infinite loop.

41. Fill in the blank so this code compiles and does not cause an infinite loop.

```

t: while (true) {
 f: while(true) {

 }
}

```

1. break;
2. break f;
3. break t;
4. None of the above

42. What is the result of the following?

```

String[] nycTourLoops = new String[] { "Downtown", "Uptown", "Brook."
String[] times = new String[] { "Day", "Night" };
for (int i = 0, j = 0; i < nycTourLoops.length
 && j < times.length; i++, j++)
{
 System.out.print(nycTourLoops[i] + " " + times[j] + "-");
}

```

1. Downtown Day-
2. Downtown Day-Uptown Night-
3. The code does not compile.
4. The code compiles but throws an exception at runtime.



43. How many lines does the following code output?

---

```
import java.util.*;
public class Exams {
 public static void main(String[] args) {
 List<String> exams = Arrays.asList("OCA", "OCP");
 for (String e1 : exams)
 for (String e2 : exams)
 System.out.println(e1 + " " + e2);
 }
}
```

---

1. One
2. Four
3. The code does not compile.
4. The code compiles but throws an exception at runtime.

44. Which of the following best describes the flow of execution in this for loop if beta always returns false?

---

```
for (alpha; beta; gamma) {
 delta;
}
```

---

1. alpha
2. alpha, beta
3. alpha, beta, gamma
4. None of the above

45. Which of the following best describes the flow of execution in this for loop if the loop body is run exactly once?

---

```
for (alpha; beta; gamma) {
 delta;
}
```

---

1. alpha, delta, gamma, beta
2. alpha, beta, delta, gamma, beta
3. alpha, delta, gamma, alpha, beta
4. alpha, beta, delta, gamma, alpha, beta

46. Which of the following iterates a different number of times than the others?

1. for (int k=0; k < 5; k++) {}
2. for (int k=1; k <= 5; k++) {}
3. int k=0; do { } while(k++ < 5)
4. int k=0; while (k++ < 5) {}

47. What is the output of the following?

---

```
public class Shoelaces {
 public static void main(String[] args) {
 String tie = null;
 while (tie == null);
 tie = "shoelace";
 System.out.print(tie);
 }
}
```

---

1. null
2. shoelace
3. shoelaceshoelace
4. None of the above

48. What is the output of the following?

---

```
12: int result = 8;
13: for: while (result > 7) {
14: result++;
15: do {
16: result--;
17: } while (result > 5);
18: break for;
```

```
19: }
20: System.out.println(result);
```

1. 5
2. 8
3. The code does not compile.
4. The code compiles but throws an exception at runtime.

49. What is the output of the following?

```
boolean balloonInflated = false;
do {
 if (!balloonInflated) {
 balloonInflated = true;
 System.out.print("inflate-");
 }
} while (balloonInflated);
System.out.println("done");
```

1. done
2. inflate-done
3. The code does not compile.
4. This is an infinite loop.

50. Which of the following can fill in the blank to have the code compile successfully?

```
package nyc;
public class TouristBus {
 public static void main(String... args) {
 String[] nycTourLoops = new String[] { "Downtown", "Uptown", "I"
 String[] times = new String[] { "Day", "Night" };
 for (_____ i < 1; i++, j++)
 System.out.println(nycTourLoops[i] + " " + times[j]);
 }
}
```

1. int i=0; j=0;
2. int i=0, j=0;
3. int i=0; int j=0;
4. int i=0, int j=0;



◀ PREV  
Chapter 4 Creating and Using Arrays

NEXT ▶  
Chapter 6 Working with Methods and Encapsulation