

# 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?
  - A. `do-while` loop
  - B. `for` (traditional)
  - C. `for-each`
  - D. `while`
2. Which type of loop is best known for using an index or counter?
  - A. `do-while` loop
  - B. `for` (traditional)
  - C. `for-each`
  - D. `while`
3. Which type of loop is guaranteed to have the body execute at least once?
  - A. `do-while` loop
  - B. `for` (traditional)
  - C. `for-each`
  - D. `while`
4. Which of the following can loop through an array without referring to the elements by index?
  - A. `do-while` loop
  - B. `for` (traditional)
  - C. `for-each`
  - D. `while`
5. What keyword is used to end the current loop iteration and proceed execution with the next iteration of that loop?
  - A. `break`
  - B. `continue`
  - C. `end`
  - D. `skip`
6. What keyword is used to proceed with execution immediately after a loop?
  - A. `break`
  - B. `continue`

C. `end`

D. `skip`

7. Which type of loop has three segments within parentheses?

A. `do-while` loop

B. `for` (traditional)

C. `for-each`

D. `while`

8. Which of the following statements is/are true?

I. A traditional `for` loop can iterate through an array starting from index 0.

II. A traditional `for` loop can iterate through an array starting from the end.

A. Only I

B. Only II

C. Both statements

D. Neither statement

9. Which of the following statements is/are true?

I. A `for-each` loop can iterate through an array starting from index 0.

II. A `for-each` loop can iterate through an array starting from the end.

A. Only I

B. Only II

C. Both statements

D. Neither statement

10. Which type of loop has a `boolean` condition that is first checked after a single iteration through the loop?

A. `do-while` loop

B. `for` (traditional)

C. `for-each`

D. `while`

11. What does the following code output?

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

A. 0

- B. The code does not compile.
- C. The loops complete with no output.
- D. 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) + ",");
```

- A. can,cup,
- B. cup,can,
- C. The code does not compile.
- D. 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(); type++) {  
        System.out.print(bottles.get(type) + ",");  
        break;  
    }  
    System.out.print("end");  
}
```

- A. glass,end
- B. glass,plastic,end
- C. The code does not compile.
- D. None of the above

14. What does the following code output?

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

- A. aa
- B. aaa
- C. The loops complete with no output.
- D. 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");
    }
}
```

- A. args
- B. argsargs
- C. The code does not compile.
- D. 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);
    }
}
```

- A. 1
- B. 2
- C. 4
- D. 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);
```

- A. 2
- B. 3
- C. The code does not compile.
- D. This is an infinite loop.

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

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

- A. `segmentA`
- B. `segmentB`
- C. `segmentC`
- D. All of the above

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

- A. One
- B. Two
- C. Three
- D. Four

10. 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) + ",");
```

- A. `can, cup,`
- B. `cup, can,`
- C. The code does not compile.
- D. None of the above

11. What happens when running the following code?

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

- A. It completes successfully without output.
- B. It outputs `helium` once.
- C. It keeps outputting `helium`.
- D. The code does not compile.

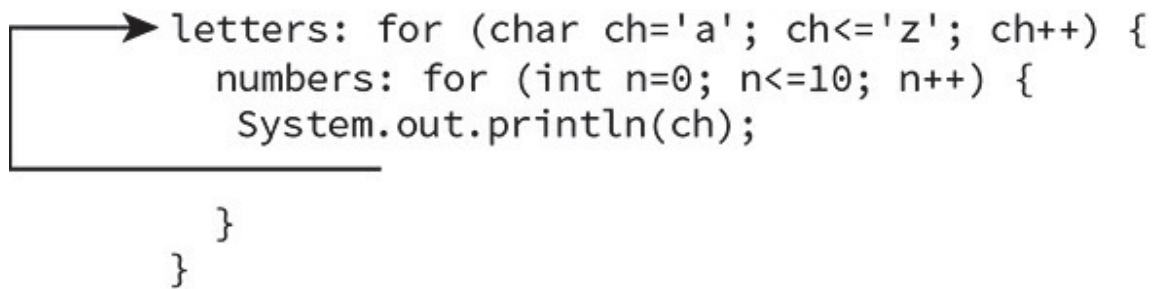
12. Which of the following is equivalent to this code snippet given an array of `String` objects?

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

- A. `for (String f = fun) System.out.println(f);`
- B. `for (String f : fun) System.out.println(f);`
- C. `for (String = fun) System.out.println(it);`
- D. None of the above

13. 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;
```



- A. None
- B. One
- C. Two
- D. Three

14. 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;
```

- A. None
- B. One
- C. Two
- D. Three

15. What does the following code output?

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

- A. 0
- B. The code does not compile.
- C. The loops completes with no output.

D. 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) {  
}
```

A. `ArrayList<Integer>`

B. `int[]`

C. `StringBuilder`

D. 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");
```

A. done

B. inflate-done

C. The code does not compile.

D. This is an infinite loop.

28. What does the following code output?

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

A. ab

B. abab

C. The loop completes with no output.

D. This is an infinite loop.

29. What describes the order in which the three expressions appear in a `for` loop?

A. `boolean` conditional, initialization expression, update statement

B. initialization expression, `boolean` conditional, update statement

C. initialization expression, update statement, `boolean` conditional



D. 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());
```

A. 3

B. 4

C. The code does not compile.

D. 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);
```

A. 1

B. 2

C. 3

D. 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]);
```

A. `for (String f = fun) System.out.println(f);`

B. `for (String f : fun) System.out.println(f);`

C. `for (String f fun) System.out.println(it);`

D. 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");  
    }
```

- A. glass,end
- B. glass,plastic,end
- C. The code does not compile.
- D. None of the above

34. What is the result of the following?

```
String[] nycTourLoops = new String[] { "Downtown", "Uptown", "Brooklyn" };  
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] + "-");  
}
```

- A. Downtown Day-
- B. Downtown Day-Uptown Night-
- C. The code does not compile.
- D. 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]);  
    }  
}
```

- A. September
- B. 1940
- C. The code does not compile.
- D. 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);  
    }  
}
```

- A. null
- B. shoelace
- C. shoelaceshoelace
- D. None of the above

7. 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);
```

- A. It prints an empty string.
- B. It still outputs a single letter x.
- C. It no longer compiles.
- D. It becomes an infinite loop.

8. 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);
    }
}
```

- A. 1
- B. 2
- C. 4
- D. The code does not compile.

9. 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);
```

- A. Leaves growing
- B. This is an infinite loop.
- C. The code does not compile.
- D. The code compiles but throws an exception at runtime.

10. 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);
```

- A. 2
- B. 3
- C. The code does not compile.
- D. This is an infinite loop.

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

```
t: while (true) {  
    f: while(true) {  
        _____  
    }  
}
```

- A. break;
- B. break f;
- C. break t;
- D. None of the above

12. What is the result of the following?

```
String[] nycTourLoops = new String[] { "Downtown", "Uptown", "Brooklyn" };  
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] + "-");  
}
```

- A. Downtown Day-

- B. Downtown Day-Uptown Night-
- C. The code does not compile.
- D. The code compiles but throws an exception at runtime.

13. 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);
    }
}
```

- A. One
- B. Four
- C. The code does not compile.
- D. The code compiles but throws an exception at runtime.

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

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

- A. `alpha`
- B. `alpha, beta`
- C. `alpha, beta, gamma`
- D. None of the above

15. 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;
}
```

- A. `alpha, delta, gamma, beta`
- B. `alpha, beta, delta, gamma, beta`
- C. `alpha, delta, gamma, alpha, beta`
- D. `alpha, beta, delta, gamma, alpha, beta`

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

- A. `for (int k=0; k < 5; k++) {}`
- B. `for (int k=1; k <= 5; k++) {}`
- C. `int k=0; do { } while(k++ < 5)`
- D. `int k=0; while (k++ < 5) {}`

17. 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);
    }
}
```

- A. `null`
- B. `shoelace`
- C. `shoelaceshoelace`
- D. None of the above

18. 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);
```

- A. 5
- B. 8
- C. The code does not compile.
- D. The code compiles but throws an exception at runtime.

19. 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");
```

- A. done
- B. inflate-done
- C. The code does not compile.
- D. This is an infinite loop.

10. 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", "Brooklyn"
    };
        String[] times = new String[] { "Day", "Night" };
        for (_____ i < 1; i++, j++)
            System.out.println(nycTourLoops[i] + " " + times[j]);
    }
}
```

- A. int i=0; j=0;
- B. int i=0, j=0;
- C. int i=0; int j=0;
- D. int i=0, int j=0;