

Introduction to Java Programming, Tenth Edition, Y. Daniel Liang

This quiz is for students to practice. A large number of additional quiz is available for instructors from the Instructor's Resource Website.

Chapter 12 Exception Handling and Text I/O

Please send suggestions and errata to Dr. Liang at y.daniel.liang@gmail.com. Indicate which book and edition you are using.
Thanks!

Section 12.3 Exception Types

12.1 A Java exception is an instance of _____.

- ☐ A. RuntimeException
- ☐ B. Exception
- ☐ C. Error
- ☒ D. Throwable
- ☐ E. NumberFormatException

Your answer is correct



12.2 An instance of _____ describes system errors. If this type of error occurs, there is little you can do beyond notifying the user and trying to terminate the program gracefully.

- ☐ A. RuntimeException
- ☐ B. Exception
- ☒ C. Error
- ☐ D. Throwable
- ☐ E. NumberFormatException

Your answer is correct



12.3 An instance of _____ describes the errors caused by your program and external circumstances. These errors can be caught and handled by your program.

- ☐ A. RuntimeException
- ☒ B. Exception
- ☐ C. Error
- ☐ D. Throwable
- ☐ E. NumberFormatException

Your answer is correct



12.4 An instance of _____ describes programming errors, such as bad casting, accessing an out-of-bounds array, and numeric errors..

- ☒ A. RuntimeException
- ☐ B. Exception
- ☐ C. Error
- ☐ D. Throwable
- ☐ E. NumberFormatException

Your answer is correct



12.5 The following code causes Java to throw _____.

```
int number = Integer.MAX_VALUE + 1;
```

- ☐ A. RuntimeException

- ☐ B. Exception
- ☐ C. Error
- ☐ D. Throwable
- ☒ E. no exceptions



Your answer is correct

[Click here to show an explanation](#)

12.6 An instance of _____ are unchecked exceptions.

- ☒ A. RuntimeException
- ☐ B. Exception
- ☒ C. Error
- ☐ D. Throwable
- ☒ E. NumberFormatException



Your answer is correct

[Click here to show an explanation](#)

12.7 What exception type does the following program throw?

```
public class Test {  
    public static void main(String[] args) {  
        System.out.println(1 / 0);  
    }  
}
```

- ☒ A. ArithmeticException
- ☐ B. ArrayIndexOutOfBoundsException
- ☐ C. StringIndexOutOfBoundsException
- ☐ D. ClassCastException
- ☐ E. No exception



Your answer is correct

12.8 What exception type does the following program throw?

```
public class Test {  
    public static void main(String[] args) {  
        int[] list = new int[5];  
        System.out.println(list[5]);  
    }  
}
```

- ☐ A. ArithmeticException
- ☒ B. ArrayIndexOutOfBoundsException
- ☐ C. StringIndexOutOfBoundsException
- ☐ D. ClassCastException
- ☐ E. No exception



Your answer is correct

12.9 What exception type does the following program throw?

```
public class Test {  
    public static void main(String[] args) {  
        String s = "abc";  
        System.out.println(s.charAt(3));  
    }  
}
```

- ☐ A. ArithmeticException
- ☐ B. ArrayIndexOutOfBoundsException

- ☒ C. StringIndexOutOfBoundsException
☐ D. ClassCastException
☐ E. No exception

Your answer is correct



12.10 What exception type does the following program throw?

```
public class Test {  
    public static void main(String[] args) {  
        Object o = new Object();  
        String d = (String)o;  
    }  
}
```

- ☐ A. ArithmeticException
☐ B. ArrayIndexOutOfBoundsException
☐ C. StringIndexOutOfBoundsException
☒ D. ClassCastException
☐ E. No exception

Your answer is correct



12.11 What exception type does the following program throw?

```
public class Test {  
    public static void main(String[] args) {  
        Object o = null;  
        System.out.println(o.toString());  
    }  
}
```

- ☐ A. ArithmeticException
☐ B. ArrayIndexOutOfBoundsException
☐ C. StringIndexOutOfBoundsException
☐ D. ClassCastException
☒ E. NullPointerException

Your answer is correct



12.12 What exception type does the following program throw?

```
public class Test {  
    public static void main(String[] args) {  
        Object o = null;  
        System.out.println(o);  
    }  
}
```

- ☐ A. ArithmeticException
☐ B. ArrayIndexOutOfBoundsException
☐ C. StringIndexOutOfBoundsException
☒ D. No exception
☐ E. NullPointerException

Your answer is correct



Section 12.4 More on Exception Handling

12.13 A method must declare to throw _____.

- ☐ A. unchecked exceptions
☒ B. checked exceptions
☐ C. Error

☐ D. RuntimeException

Your answer is correct



12.14 Which of the following statements are true?

- ☒ A. You use the keyword throws to declare exceptions in the method heading.
- ☒ B. A method may declare to throw multiple exceptions.
- ☒ C. To throw an exception, use the key word throw.
- ☒ D. If a checked exception occurs in a method, it must be either caught or declared to be thrown from the method.

Your answer is correct



12.15 Analyze the following code:

```
public class Test {  
    public static void main(String[] args)  
        throws MyException {  
        System.out.println("Welcome to Java");  
    }  
}  
  
class MyException extends Error {  
}
```

- ☒ A. You should not declare a class that extends Error, because Error raises a fatal error that terminates the program.
- ☐ B. You cannot declare an exception in the main method.
- ☐ C. You declared an exception in the main method, but you did not throw it.
- ☐ D. The program has a compile error.

Your answer is correct

[Click here to show an explanation](#)



12.16 Analyze the following code:

```
public class Test {  
    public static void main(String[] args) {  
        try {  
            String s = "5.6";  
            Integer.parseInt(s); // Cause a NumberFormatException  
  
            int i = 0;  
            int y = 2 / i;  
        }  
        catch (Exception ex) {  
            System.out.println("NumberFormatException");  
        }  
        catch (RuntimeException ex) {  
            System.out.println("RuntimeException");  
        }  
    }  
}
```

- ☐ A. The program displays NumberFormatException.
- ☐ B. The program displays RuntimeException.
- ☐ C. The program displays NumberFormatException followed by RuntimeException.
- ☒ D. The program has a compile error.

Your answer is correct

[Click here to show an explanation](#)



12.17 Analyze the following program.

```
public class Test {  
    public static void main(String[] args) {  
        try {
```

```
String s = "5.6";
Integer.parseInt(s); // Cause a NumberFormatException

int i = 0;
int y = 2 / i;
System.out.println("Welcome to Java");
}
catch (Exception ex) {
    System.out.println(ex);
}
}
```

- ☒ A. An exception is raised due to Integer.parseInt(s);
- ☐ B. An exception is raised due to 2 / i;
- ☐ C. The program has a compile error.
- ☐ D. The program compiles and runs without exceptions.



Your answer is correct

[Click here to show an explanation](#)

12.18 What is displayed on the console when running the following program?

```
public class Test {
    public static void main(String[] args) {
        try {
            method();
            System.out.println("After the method call");
        }
        catch (NumberFormatException ex) {
            System.out.println("NumberFormatException");
        }
        catch (RuntimeException ex) {
            System.out.println("RuntimeException");
        }
    }

    static void method() {
        String s = "5.6";
        Integer.parseInt(s); // Cause a NumberFormatException

        int i = 0;
        int y = 2 / i;
        System.out.println("Welcome to Java");
    }
}
```

- ☒ A. The program displays NumberFormatException.
- ☐ B. The program displays NumberFormatException followed by After the method call.
- ☐ C. The program displays NumberFormatException followed by RuntimeException.
- ☐ D. The program has a compile error.
- ☐ E. The program displays RuntimeException.



Your answer is correct

12.19 What is displayed on the console when running the following program?

```
public class Test {
    public static void main(String[] args) {
        try {
            method();
            System.out.println("After the method call");
        }
        catch (RuntimeException ex) {
            System.out.println("RuntimeException");
        }
        catch (Exception ex) {
            System.out.println("Exception");
        }
    }

    static void method() throws Exception {
        try {
```

```
String s = "5.6";
Integer.parseInt(s); // Cause a NumberFormatException

int i = 0;
int y = 2 / i;
System.out.println("Welcome to Java");
}
catch (RuntimeException ex) {
    System.out.println("RuntimeException");
}
catch (Exception ex) {
    System.out.println("Exception");
}
}
```

- ☐ A. The program displays RuntimeException twice.
- ☐ B. The program displays Exception twice.
- ☒ C. The program displays RuntimeException followed by After the method call.
- ☐ D. The program displays Exception followed by RuntimeException.
- ☐ E. The program has a compile error.

Your answer is correct



Section 12.5 The finally Clause

12.20 What is wrong in the following program?

```
public class Test {
    public static void main (String[] args) {
        try {
            System.out.println("Welcome to Java");
        }
    }
}
```

- ☐ A. You cannot have a try block without a catch block.
- ☒ B. You cannot have a try block without a catch block or a finally block.
- ☐ C. A method call that does not declare exceptions cannot be placed inside a try block.
- ☐ D. Nothing is wrong.

Your answer is correct



12.21 What is displayed on the console when running the following program?

```
public class Test {
    public static void main (String[] args) {
        try {
            System.out.println("Welcome to Java");
        }
        finally {
            System.out.println("The finally clause is executed");
        }
    }
}
```

- ☐ A. Welcome to Java
- ☒ B. Welcome to Java followed by The finally clause is executed in the next line
- ☐ C. The finally clause is executed
- ☐ D. None of the above

Your answer is correct



12.22 What is displayed on the console when running the following program?

```
public class Test {
    public static void main (String[] args) {
        try {
            System.out.println("Welcome to Java");
            return;
        }
    }
}
```

```
    }  
    finally {  
        System.out.println("The finally clause is executed");  
    }  
}
```

- ☐ A. Welcome to Java
- ☒ B. Welcome to Java followed by The finally clause is executed in the next line
- ☐ C. The finally clause is executed
- ☐ D. None of the above



Your answer is correct

[Click here to show an explanation](#)

12.23 What is displayed on the console when running the following program?

```
public class Test {  
    public static void main(String[] args) {  
        try {  
            System.out.println("Welcome to Java");  
            int i = 0;  
            int y = 2 / i;  
            System.out.println("Welcome to HTML");  
        }  
        finally {  
            System.out.println("The finally clause is executed");  
        }  
    }  
}
```

- ☐ A. Welcome to Java, then an error message.
- ☒ B. Welcome to Java followed by The finally clause is executed in the next line, then an error message.
- ☐ C. The program displays three lines: Welcome to Java, Welcome to HTML, The finally clause is executed, then an error message.
- ☐ D. None of the above.



Your answer is correct

12.24 What is displayed on the console when running the following program?

```
public class Test {  
    public static void main(String[] args) {  
        try {  
            System.out.println("Welcome to Java");  
            int i = 0;  
            double y = 2.0 / i;  
            System.out.println("Welcome to HTML");  
        }  
        finally {  
            System.out.println("The finally clause is executed");  
        }  
    }  
}
```

- ☐ A. Welcome to Java.
- ☐ B. Welcome to Java followed by The finally clause is executed in the next line.
- ☒ C. The program displays three lines: Welcome to Java, Welcome to HTML, The finally clause is executed.
- ☐ D. None of the above.



Your answer is correct

[Click here to show an explanation](#)

12.25 What is displayed on the console when running the following program?

```
public class Test {  
    public static void main(String[] args) {  
        try {
```

```
        System.out.println("Welcome to Java");
        int i = 0;
        int y = 2/i;
        System.out.println("Welcome to Java");
    }
    catch (RuntimeException ex) {
        System.out.println("Welcome to Java");
    }
    finally {
        System.out.println("End of the block");
    }
}
```

- ☐ A. The program displays Welcome to Java three times followed by End of the block.
- ☒ B. The program displays Welcome to Java two times followed by End of the block.
- ☐ C. The program displays Welcome to Java three times.
- ☐ D. The program displays Welcome to Java two times.

Your answer is correct



12.26 What is displayed on the console when running the following program?

```
public class Test {
    public static void main(String[] args) {
        try {
            System.out.println("Welcome to Java");
            int i = 0;
            int y = 2/i;
            System.out.println("Welcome to Java");
        }
        catch (RuntimeException ex) {
            System.out.println("Welcome to Java");
        }
        finally {
            System.out.println("End of the block");
        }

        System.out.println("End of the block");
    }
}
```

- ☐ A. The program displays Welcome to Java three times followed by End of the block.
- ☐ B. The program displays Welcome to Java two times followed by End of the block.
- ☒ C. The program displays Welcome to Java two times followed by End of the block two times.
- ☐ D. You cannot catch RuntimeException errors.

Your answer is correct



12.27 What is displayed on the console when running the following program?

```
public class Test {
    public static void main(String[] args) {
        try {
            System.out.println("Welcome to Java");
            int i = 0;
            int y = 2/i;
            System.out.println("Welcome to Java");
        }
        finally {
            System.out.println("End of the block");
        }

        System.out.println("End of the block");
    }
}
```

- ☐ A. The program displays Welcome to Java three times followed by End of the block.
- ☐ B. The program displays Welcome to Java two times followed by End of the block.
- ☐ C. The program displays Welcome to Java two times followed by End of the block two times.

- ☒ D. The program displays Welcome to Java and End of the block, and then terminates because of an unhandled exception.

Your answer is correct



Section 12.6 When to Use Exceptions

12.28 Which of the following is not an advantage of Java exception handling?

- ☐ A. Java separates exception handling from normal processing tasks.
- ☒ B. Exception handling improves performance.
- ☐ C. Exception handling makes it possible for the caller's caller to handle the exception.
- ☐ D. Exception handling simplifies programming because the error-reporting and error-handling code can be placed at the catch block.

Your answer is correct



12.29 Analyze the following code:

```
public class Test {
    public static void main(String[] args) {
        try {
            int zero = 0;
            int y = 2/zero;
            try {
                String s = "5.6";
                Integer.parseInt(s); // Cause a NumberFormatException
            }
            catch(Exception e) {
            }
        }
        catch(RuntimeException e) {
            System.out.println(e);
        }
    }
}
```

- ☐ A. A try-catch block cannot be embedded inside another try-catch block.
- ☒ B. A good programming practice is to avoid nesting try-catch blocks, because nesting makes programs difficult to read. You can rewrite the program using only one try-catch block.
- ☐ C. The program has a compile error because Exception appears before RuntimeException.
- ☐ D. None of the above.

Your answer is correct



[Click here to show an explanation](#)

Section 12.7 Rethrowing Exceptions

12.30 What is displayed on the console when running the following program?

```
public class Test {
    public static void main(String[] args) {
        try {
            method();
            System.out.println("After the method call");
        }
        catch (RuntimeException ex) {
            System.out.println("RuntimeException");
        }
        catch (Exception ex) {
            System.out.println("Exception");
        }
    }

    static void method() throws Exception {
        try {
            String s = "5.6";
            Integer.parseInt(s); // Cause a NumberFormatException

            int i = 0;
        }
    }
}
```

```
        int y = 2 / i;
        System.out.println("Welcome to Java");
    }
    catch (NumberFormatException ex) {
        System.out.println("NumberFormatException");
        throw ex;
    }
    catch (RuntimeException ex) {
        System.out.println("RuntimeException");
    }
}
```

- ☐ A. The program displays NumberFormatException twice.
- ☐ B. The program displays NumberFormatException followed by After the method call.
- ☒ C. The program displays NumberFormatException followed by RuntimeException.
- ☐ D. The program has a compile error.

Your answer is correct



Section 12.10 The File Class

12.31 What are the reasons to create an instance of the File class?

- ☒ A. To determine whether the file exists.
- ☒ B. To obtain the properties of the file such as whether the file can be read, written, or is hidden.
- ☒ C. To rename the file.
- ☒ D. To delete the file.
- ☐ E. To read/write data from/to a file

Your answer is correct



12.32 Which of the following returns the path separator character?

- ☐ A. File.pathSeparator
- ☒ B. File.pathSeparatorChar
- ☐ C. File.separator
- ☐ D. File.separatorChar
- ☐ E. None of the above.

Your answer is correct



12.33 Which of the following statements creates an instance of File on Window for the file c:\temp.txt?

- ☐ A. new File("c:\temp.txt")
- ☒ B. new File("c:\\temp.txt")
- ☐ C. new File("c:/temp.txt")
- ☐ D. new File("c://temp.txt")

Your answer is correct



12.34 Which of the following statements are true?

- ☐ A. If a file (e.g., c:\temp.txt) does not exist, new File("c:\\temp.txt") returns null.
- ☐ B. If a directory (e.g., c:\liang) does not exist, new File("c:\\liang") returns null.
- ☐ C. If a file (e.g., c:\temp.txt) does not exist, new File("c:\\temp.txt") creates a new file named c:\temp.txt.
- ☐ D. If a directory (e.g., c:\liang) does not exist, new File("c:\\liang") creates a new directory named c:\liang.
- ☒ E. None of the above.

Your answer is correct



Section 12.11 Text I/O

12.35 Which class contains the method for checking whether a file exists?

- ☒ A. File
- ☐ B. PrintWriter
- ☐ C. Scanner
- ☐ D. System

Your answer is correct



12.36 Which class do you use to write data into a text file?

- ☐ A. File
- ☒ B. PrintWriter
- ☐ C. Scanner
- ☐ D. System

Your answer is correct



12.37 Which class do you use to read data from a text file?

- ☐ A. File
- ☐ B. PrintWriter
- ☒ C. Scanner
- ☐ D. System

Your answer is correct



12.38 Which method can be used to write data?

- ☐ A. close
- ☒ B. print
- ☐ C. exist
- ☐ D. rename

Your answer is correct



12.39 Which method can be used to read a whole line from the file?

- ☐ A. next
- ☒ B. nextLine
- ☐ C. nextInt
- ☐ D. nextDouble

Your answer is correct



12.40 Which of the following statements are correct?

```
I:
try (PrintWriter output = new PrintWriter("output.txt")) {
    output.println("Welcome to Java");
}

II:
try (PrintWriter output = new PrintWriter("output.txt");) {
    output.println("Welcome to Java");
}

III:
PrintWriter output;
```

```
try (output = new PrintWriter("output.txt");) {  
    output.println("Welcome to Java");  
}  
  
IV:  
try (PrintWriter output = new PrintWriter("output.txt");) {  
    output.println("Welcome to Java");  
}  
finally {  
    output.close();  
}
```

- ☒ A. I
☒ B. II
☒ C. III
☐ D. IV



Your answer is correct

12.41 Which of the following statements are correct?

```
I:  
File file = new File("input.txt");  
try (Scanner input = new Scanner(file)) {  
    String line = input.nextLine();  
}  
  
II:  
try (File file = new File("input.txt");  
    Scanner input = new Scanner(file);) {  
    String line = input.nextLine();  
}  
  
III:  
File file;  
try (file = new File("input.txt");  
    Scanner input = new Scanner(file);) {  
    String line = input.nextLine();  
}  
  
IV:  
File file;  
Scanner input;  
try (file = new File("input.txt");  
    input = new Scanner(file);) {  
    String line = input.nextLine();  
}
```

- ☒ A. I
☐ B. II
☐ C. III
☐ D. IV



Your answer is correct

[Click here to show an explanation](#)

12.42 Which method can be used to create an input object for file temp.txt?

- ☐ A. new Scanner("temp.txt")
☐ B. new Scanner(temp.txt)
☒ C. new Scanner(new File("temp.txt"))
☐ D. new Scanner(File("temp.txt"))



Your answer is correct

12.43 Suppose you enter 34.3 57.8 789, then press the ENTER key. Analyze the following code.

```
Scanner input = new Scanner(System.in);  
int v1 = input.nextInt();  
int v2 = input.nextInt();
```

```
String line = input.nextLine();
```

- ☐ A. After the last statement is executed, v1 is 34.
- ☒ B. The program has a runtime error because 34.3 is not an integer.
- ☐ C. After the last statement is executed, line contains characters '7', '8', '9', '\n'.
- ☐ D. After the last statement is executed, line contains characters '7', '8', '9'.



Your answer is correct

[Click here to show an explanation](#)

12.44 Suppose you enter 34.3 57.8 789, then press the ENTER key. Analyze the following code.

```
Scanner input = new Scanner(System.in);
double v1 = input.nextDouble();
double v2 = input.nextDouble();
String line = input.nextLine();
```

- ☐ A. After the last statement is executed, line contains characters '7', '8', '9'.
- ☐ B. After the last statement is executed, line contains characters '7', '8', '9', '\n'.
- ☐ C. After the last statement is executed, line contains characters ' ', '7', '8', '9', '\n'.
- ☒ D. After the last statement is executed, line contains characters ' ', '7', '8', '9'.



Your answer is correct

[Click here to show an explanation](#)

12.45 Suppose you enter 34.3, the ENTER key, 57.8, the ENTER key. Analyze the following code.

```
1 Scanner input = new Scanner(System.in);
2 double v1 = input.nextDouble();
3 double v2 = input.nextDouble();
4 String line = input.nextLine();
```

- ☒ A. After line 2 is executed, v1 is 34.3.
- ☒ B. After line 3 is executed, v2 is 57.8.
- ☒ C. After line 4 is executed, line contains an empty string.
- ☐ D. After line 4 is executed, line is null.
- ☐ E. After line 4 is executed, line contains character "\n".



Your answer is correct

[Click here to show an explanation](#)

12.46 Suppose you enter 34.3, the ENTER key, 57.8, the ENTER key, abc, the Enter key. Analyze the following code.

```
1 Scanner input = new Scanner(System.in);
2 double v1 = input.nextDouble();
3 double v2 = input.nextDouble();
4 String line = input.nextLine();
```

- ☒ A. After line 2 is executed, v1 is 34.3.
- ☒ B. After line 3 is executed, v2 is 57.8.
- ☒ C. After line 4 is executed, line contains an empty string.
- ☐ D. After line 4 is executed, line is null.
- ☐ E. After line 4 is executed, line contains character "abc".



Your answer is correct

[Click here to show an explanation](#)

12.47 Which method can be used to create an output object for file temp.txt?

- ☒ A. new PrintWriter("temp.txt")
- ☐ B. new PrintWriter(temp.txt)
- ☒ C. new PrintWriter(new File("temp.txt"))

☐ D. `new PrintWriter(File("temp.txt"))`

Your answer is correct



Section 12.12 Reading Data from the Web

12.48 To create an `InputStream` to read from a file on a Web server, you use the method _____ in the `URL` class.

- ☐ A. `getInputStream();`
- ☐ B. `obtainInputStream();`
- ☒ C. `openStream();`
- ☐ D. `connectStream();`

Your answer is correct

