

1. What will be the output of the following code?

```
public class Test {  
    public static void main(String[] args) {  
        try {  
            throw new ArithmeticException("Arithmetic exception occurred");  
        } catch (ArithmeticException e) {  
            System.out.println(e.getMessage());  
        }  
    }  
}
```

- Options: A) Arithmetic exception occurred
B) Exception occurred
C) ArithmeticException
D) RuntimeException
-

2. What will be the output of the following code?

```
public class Test {  
    public static void main(String[] args) throws Exception {  
        throw new Exception("Checked Exception");  
    }  
}
```

- Options: A) Checked Exception
B) Exception is thrown at runtime
C) Compilation error due to unhandled exception
D) No output
-

3. What is the main difference between checked and unchecked exceptions in Java?

- Options: A) Checked exceptions must be explicitly caught or declared, while unchecked exceptions do not
B) Checked exceptions are inherited from `Error` class, unchecked from `Exception`
C) Unchecked exceptions are objects of `Throwable`, checked are not
D) There is no difference
-

4. Which of the following is an unchecked exception?

- Options: A) `IOException`
B) `SQLException`
C) `ArithmeticException`
D) `FileNotFoundException`
-

5. What will happen if we throw a checked exception without declaring it in the method signature?

- Options: A) Compilation error
B) Runtime error

- C) No error, the exception is handled implicitly
 - D) The program will not compile until the exception is handled
-

6. What is the correct way to declare that a method throws a checked exception?

- Options:
- A) `public void method() throw Exception`
 - B) `public void method() throws Exception`
 - C) `public void method() throws (Exception)`
 - D) `public void method() exception throws`
-

7. What is the purpose of the `throw` keyword in Java?

- Options:
- A) It is used to declare an exception
 - B) It is used to handle exceptions
 - C) It is used to create and throw an exception explicitly
 - D) It is used to catch exceptions
-

8. Which of the following will cause a compilation error?

```
public class Test {  
    public static void main(String[] args) throws Exception {  
        throw new NullPointerException();  
    }  
}
```

- Options:
- A) `NullPointerException` is a checked exception
 - B) `NullPointerException` is an unchecked exception
 - C) It must be declared with `throws` in the method signature
 - D) The program will compile and run correctly
-

9. What is the output of the following code?

```
public class Test {  
    public static void main(String[] args) {  
        try {  
            throw new Exception("Checked Exception");  
        } catch (Exception e) {  
            System.out.println(e.getMessage());  
        }  
    }  
}
```

- Options:
- A) Checked Exception
 - B) Exception thrown
 - C) Exception caught
 - D) Compilation error
-

10. What will be the output of the following code?

```
public class Test {
    public static void main(String[] args) {
        try {
            throw new RuntimeException("Unchecked exception");
        } catch (RuntimeException e) {
            System.out.println(e.getMessage());
        }
    }
}
```

- Options: A) Unchecked exception
 B) RuntimeException
 C) Exception
 D) No output
-

11. Which of the following exceptions are unchecked?

- Options: A) IOException
 B) ClassNotFoundException
 C) ArithmeticException
 D) InterruptedException
-

12. What is the output of the following code?

```
public class Test {
    public static void main(String[] args) throws Exception {
        throw new ClassNotFoundException("Checked Exception");
    }
}
```

- Options: A) Compilation error because ClassNotFoundException is not handled
 B) RuntimeException
 C) ClassNotFoundException is thrown
 D) Checked Exception
-

13. What will happen if a method throws an unchecked exception and it is not caught?

- Options: A) Compilation error
 B) Runtime error
 C) No issue, the program continues executing
 D) Exception is automatically caught
-

14. Can we throw an exception from a method that does not declare it in its throws clause?

- Options: A) Yes, only unchecked exceptions can be thrown
 B) Yes, but only if the exception is a runtime exception
 C) No, only checked exceptions can be thrown
 D) No, the method must declare the exception
-

15. Which of the following is the correct way to throw a custom exception in Java?

- Options:
- A) `throw new MyException("Custom Exception")`
 - B) `throw MyException("Custom Exception")`
 - C) `throw MyException()`
 - D) `throws new MyException("Custom Exception")`
-

16. Which of the following is a checked exception?

- Options:
- A) `NullPointerException`
 - B) `ArithmeticException`
 - C) `IOException`
 - D) `RuntimeException`
-

17. What happens if a method does not handle an exception but throws it using the `throws` keyword?

- Options:
- A) The program will terminate immediately
 - B) The exception must be caught or declared by the calling method
 - C) The exception is automatically caught by the JVM
 - D) The code will compile without error
-

18. Which of the following is correct regarding `throw` and `throws`?

- Options:
- A) `throw` is used for declaring exceptions, `throws` is used to actually throw an exception
 - B) `throw` is used for throwing an exception, `throws` is used for declaring exceptions
 - C) `throws` can only be used with checked exceptions
 - D) `throw` can only be used with unchecked exceptions
-

19. Consider the following method:

```
public void myMethod() throws IOException {  
    throw new IOException("IO Exception occurred");  
}
```

Which of the following is true?

- Options:
- A) The method must be surrounded by a try-catch block to catch the `IOException`
 - B) The method must declare the exception using the `throws` keyword
 - C) This code will cause a compilation error because `IOException` is unchecked
 - D) `IOException` will be caught by the JVM automatically
-

20. What is the correct statement about throwing multiple exceptions?

- Options:
- A) A method can throw multiple exceptions, which must be handled by the calling method
 - B) A method can throw multiple exceptions, but it must use multiple `throw` statements
 - C) A method cannot throw multiple exceptions
 - D) A method can throw only one exception at a time
