**Q1**. Given:

class Plane {

static String s = "-";

public static void main(String[] args) {

new Plane().s1();

System.out.println(s);

}

void s1() {

try {

s2();

} catch (Exception e) {

s += "c";

}

}

void s2() throws Exception {

s3();

s += "2";

s3();

s += "2b";

}

void s3() throws Exception {

throw new Exception();

}

}

What is the result?

A. -

**B. -c**

C. -c2

D. -2c

E. -c22b

F. -2c2b

G. -2c2bc

H. Compilation fails

**Response**: **B. -c**

**Q2.** Given:

try { int x = Integer.parseInt("two"); }

Which could be used to create an appropriate catch block? (Choose all that apply.)

A. ClassCastException

B. IllegalStateException

**C. NumberFormatException**

**D. IllegalArgumentException**

E. ExceptionInInitializerError

F. ArrayIndexOutOfBoundsException

**Response:**

**C. NumberFormatException**

**D. IllegalArgumentException**

**Q3.** Given:

class Emu {

static String s = "-";

public static void main(String[] args) {

try {

throw new Exception();

} catch (Exception e) {

try {

try {

throw new Exception();

} catch (Exception ex) {

s += "ic ";

}

throw new Exception();

} catch (Exception x) {

s += "mc ";

} finally {

s += "mf ";

}

} finally {

s += "of ";

}

System.out.println(s);

}

}

What is the result?

A. -ic of

B. -mf of

C. -mc mf

D. -ic mf of

**E. -ic mc mf of**

F. -ic mc of mf

G. Compilation fail

**Response:**

**E. -ic mc mf of**

**Q4.** Given:

1. class SubException extends Exception {}

2. class SubSubException extends SubException {}

3. public class CC {

4. void doStuff() throws SubException {}

5. }

6. class CC2 extends CC {

7. void doStuff() throws SubSubException {}

8. }

9. class CC3 extends CC {

10. void doStuff() throws Exception {}

11. }

12. class CC4 extends CC {

13. void doStuff(int x) throws Exception {}

14. }

15. class CC5 extends CC {

16. void doStuff() {}

17. }

What is the result? (Choose all that apply.)

A. Compilation succeeds

B. Compilation fails due to an error on line 8

C. **Compilation fails due to an error on line 10**

D. Compilation fails due to an error on line 13

E. Compilation fails due to an error on line 16

**Response:**

C. **Compilation fails due to an error on line 10**

**Q5.** Given:

public class OverAndOver {

static String s = "";

public static void main(String[] args) {

try {

s += "1";

throw new Exception();

} catch (Exception e) {

s += "2";

} finally {

s += "3";

doStuff();

s += "4";

}

System.out.println(s);

}

static void doStuff() {

int x = 0;

int y = 7 / x;

}

}

What is the result?

A. 12

B. 13

C. 123

D. 1234

E.Compilation fails

F.123 followed by an exception

G.1234 followed by an exception

H.**An exception is thrown with no other output**

**Response**:

H.**An exception is thrown with no other output**

**Q6.** Given:

public class Gotcha {

public static void main(String[] args) {

// insert code here

}

void go() {

go();

}

}

And given the following three code fragments:

I. new Gotcha().go();

II. try { new Gotcha().go(); }

catch (Error e) { System.out.println("ouch"); }

III. try { new Gotcha().go(); }

catch (Exception e) { System.out.println("ouch"); }

When fragments I–III are added, independently, at comment line, which are true?

(Choose all that apply.)

A. Some will not compile

B. They will all compile

C. All will complete normally

D. None will complete normally

E. Only one will complete normally

**F. Two of them will complete normally**

**Response**:

**F. Two of them will complete normally**

**Q7.** Given:

public class Frisbee {

// insert code here

int x = 0;

System.out.println(7 / x);

}

}

And given the following four code fragments:

I. public static void main(String[] arg) {

II. public static void main(String[] arg) throws Exception {

III. public static void main(String[] arg) throws IOException {

IV. public static void main(String[] arg) throws RunTimeException {

If the four fragments are inserted independently at comment line, which are true? (Choose all that apply.)

A. All four will compile and execute without exception

B. All four will compile and execute and throw an exception

C. Some, but not all, will compile and execute without exception

**D. Some, but not all, will compile and execute and throw an exception**

**E. When considering fragments II, III, and IV, of those that will compile, adding a try/catch block around line 4 will cause compilation to fail**

**Q8.** Given:

class MyException extends Exception {}

class Tire {

void doStuff() {}

}

public class Retread extends Tire {

public static void main(String[] args) {

new Retread().doStuff();

}

// insert code here

System.out.println(7 / 0);

}

}

And given the following four code fragments:

I. void doStuff() {

II. void doStuff() throws MyException {

III.void doStuff() throws RuntimeException {

IV. void doStuff() throws ArithmeticException {

When fragments I–IV are added, independently, at line 10, which are true? (Choose all that apply.)

A. None will compile

B. They will all compile

C. Some, but not all, will compile

D. All of those that compile will throw an exception at runtime

E. None of those that compile will throw an exception at runtime

F. Only some of those that compile will throw an exception at runtime