Exceptions & IO Workbook

Answer the Following

1. What if there is a `break` or `return` statement in try block followed by finally block?

2. Explain different types of exceptions in Java.

3. How to create custom exceptions?

4. Why runtime exceptions are not checked?

5. What is the use of finally block?

6. What is meant by Stream and what are the types of Streams and classes of the Streams?

7. What is the difference between the Reader/Writer class hierarchy and the InputStream/OutputStream class hierarchy?

8. What an I/O filter?

9. What is serialization?

State whether the following are True/False

1. The File class contains a method that changes the current working directory. [ ]
2. Readers have methods that can read and return floats and doubles. [ ]
3. The InputStream and OutputStream classes are byte-oriented. [ ]
4. The Serializable interface extends the Externalizable interface. [ ]
5. Writer classes can be used to write Unicode characters to output streams. [ ]
6. We can have catch block with out try block. [ ]
7. Exceptions can be caught or rethrown to a calling method. [ ]
8. The toString ( ) method in the user-defined exception class is overridden.[ ]

Multiple Choice Questions

1. When does Exceptions in Java arises in code sequence?
   (a) Run time
   (b) Compilation time
   (c) Can occur any time
   (d) None

2. Which of these keywords must be used to monitor for exceptions?

   (a) try
   (b) finally
   (c) catch
   (d) throw

3. Which of these keywords is used to manually throw an exception?

   (a) throw
   (b) throws
   (c) catch
   (d) finally

4. Which of these class is used to read from byte array?

   (a) InputStream.
   (b) BufferedInputStream.
   (c) ArrayInputStream.
   (d) ByteArrayInputStream.

5. Which of these classes are used by byte streams for input and output operation?

   (a) InputStream
   (b) InputOutputStream
   (c) Reader
   (d) All the above

6. Which of these class contains the methods print() and println()?

   (a) System
   (b) System.out
   (c) BufferedOutputStream
   (d) PrintStream

7. Which of these classes are used by character streams output operations?

   (a) InputStream
   (b) Writer
   (c) ReadStream
   (d) InputOutputStream

Exercises

- Write the expected output, or compiler errors if any, for each of the following programs in the box provided below each program.
- Then execute the programs and check your answers.
- Then answer the questions given below.

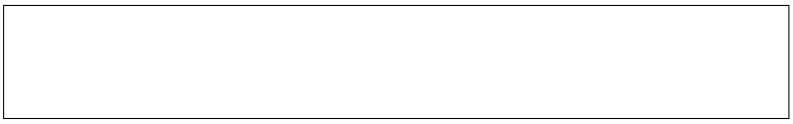
Program 1

```
1  public class Tester {
2      public static void main(String[] args) {
3          System.out.print(''1'');
4          try {
5              return;
6          } catch (Exception e) {
7              System.out.print(''2'');
8          } finally {
9              System.out.print(''3'');
10         }
11         System.out.print(''4'');
12     }
13 }
```

Q1: What is the result of compiling and running the above code?

Program 2

```
1  import java.io.IOException;
2  class AirPlane {
3      public AirPlane() throws IOException, RuntimeException {
4          System.out.println(''AirPlane'');
5      }
6  }
7  class AirJet extends AirPlane {
8
9  }
10 public class Tester {
11     public static void main(String args[]) throws IOException {
12         new AirPlane();
13     }
14 }
```



Q1: What is the expected output of compiling and running the above code?

Program 3

```
1  public class Tester {
2      static void method() {
3          throw new Exception();
4      }
5      public static void main(String[] args) {
6          try {
7              method();
8          } catch (Throwable e) {
9              try {
10                 throw new Exception() ;
11             } catch (Exception ex) {
12             System.out.print(``exception'');
13             } finally {
14                 System.out.print(``finally'');
15             }
16         }
17     }
18 }
```

Q1: What is the result of compiling and running the above code?

Program 4

```
1  interface Foldable {
2      public void fold() throws Exception ;
3  }
4  class Paper implements Foldable {
5      public void fold() {
6          System.out.print(``Fold'');
7      }
8  }
9  public class Tester {
10     public static void main(String args []) {
```

```
        Foldable obj1 = new Paper();
        obj1.fold();
        Paper obj2 = new Paper();
        obj2.fold();
    }
}
```

Q1: What is the output of the above code?

Program 5

```
class Father {
    public Father() throws RuntimeException {
        System.out.print(``Father'');
        throw new RuntimeException();
    }
}
class Son extends Father {
    public Son() throws RuntimeException {
        System.out.print(``Son'');
    }
}
public class Tester {
    public static void main(String[] args) {
        new Son();
    }
}
```

Q1: What is the expected output of compiling and running this code?

Program 6

```
try {
    File file = new File(``file.dat'');
    file.createNewFile();
}
catch (// INSERT EXCEPTION TYPE e) {
    e.printStackTrace();
}
```

Q1: Given the above code segment enclosed within a try/catch block, what valid Exception types can be catched causing no compilation error?

Program 7

```
1  try {
2      int x = 0;
3      int y = 5 / x;
4  }
5  catch (Exception e) {
6      System.out.println(``Exception'');
7  }
8  catch (ArithmeticException ae) {
9      System.out.println(``Arithmetic Exception'');
10 }
11     System.out.println(``finished'');
```

Q1: What will be the output of the program?

Program 8

```
1  public class Foo {
2      public static void main(String[] args) {
3          try {
4              return;
5          }
6          finally {
7              System.out.println(``Finally'');
8          }
9      }
10 }
```

Q1: What will be the output of the program?