Java Software Development Quiz 10

1. Which one of these methods in the File class will return the name of the entry, excluding the specification of the directory in which it resides?

Select the one correct answer.

(a) getAbsolutePath()

(b) getName()

(c) getParent()

(d) getPath()

(e) None of the above.

2. Given the following program:

```
import java.io.File;
import java.io.IOException;
public final class Filing {
   public static void main (String[] args) throws IOException {
      File file = new File("./documents","../book/../chapter1");
      System.out.println(file.getPath());
      System.out.println(file.getAbsolutePath());
      System.out.println(file.getCanonicalPath());
      System.out.println(file.getName());
      System.out.println(file.getParent());
   }
}
```

Assume that the current or working directory has the absolute path "/wrk". Which lines below will not be included in the output from the program?

Select the two correct answers.

(a) ./documents/../book/../chapter1

(b) ./documents/book/chapter1

(c) /wrk/./documents/../book/../chapter1

(d) /wrk/documents/book/chapter1

(e) /wrk/chapter1

(f) chapter1

(g) ./documents/../book/..

3. Which of these classes provides methods for writing binary representations of Java primitive values?

Select the two correct answers.

(a) DataOutputStream

(b) FileOutputStream

(c) ObjectOutputStream

(d) PrintStream

(e) BufferedOutputStream

4. Which of the following best describes the data written by an ObjectOutputStream?

Select the one correct answer.

(a) Bytes and other Java primitive types.

(b) Object hierarchies.

(c) Object hierarchies and Java primitive types.

(d) Single objects.

(e) Single objects and Java primitive types.

5. Given the following code:

```
public class Person {
   protected String name;
   Person() { }
   Person(String name) { this.name = name; }
}
```

```
import java.io.Serializable;
public class Student extends Person implements Serializable {
   private long studNum;
      Student(String name, long studNum) {
      super(name);
      this.studNum = studNum;
   }
   public String toString() { return "(" + name + ", " + studNum + ")"; }
}
```

```
import java.io.*;

public class RQ800_10 {
   public static void main(String args[]) throws IOException,
         ClassNotFoundException {
      FileOutputStream outputFile = new FileOutputStream("storage.dat");
      ObjectOutputStream outputStream=new ObjectOutputStream(outputFile);
      Student stud1 = new Student("Aesop", 100);
      System.out.print(stud1);
      outputStream.writeObject(stud1);
```

```
        outputStream.flush();
        outputStream.close();
        FileInputStream inputFile = new FileInputStream("storage.dat");
        ObjectInputStream inputStream = new ObjectInputStream(inputFile);
        Student stud2 = (Student) inputStream.readObject();
        System.out.println(stud2);
        inputStream.close();
    }
}
```

Which statement about the program is true?

Select the one correct answer.

(a) It fails to compile.

(b) It compiles, but throws an exception at runtime.

(c) It prints (Aesop, 100)(Aesop, 100).

(d) It prints (Aesop, 100)(null, 100).

(e) It prints (Aesop, 100)(, 100).

6. Given the following code:

```
import java.io.Serializable;
public class Person implements Serializable {
   protected transient String name;
   Person(String name) { this.name = name; }
}
```

```
public class Student extends Person {
   private static int numOfStudents;
   private long studNum;
   Student(String name, long studNum) {
      super(name);
      this.studNum = studNum;
      ++numOfStudents;
   }
   public String toString() {
      return "(" + name + ", " + studNum + ", " + numOfStudents + ")";
   }
}
```

```
import java.io.*;
public class RQ800_80 {
   public static void main(String args[])
      throws IOException, ClassNotFoundException {
      FileOutputStream outputFile = new FileOutputStream("storage.dat");
      ObjectOutputStream outputStream=new ObjectOutputStream(outputFile);
      Student stud1 = new Student("Aesop", 100);
      System.out.print(stud1);
      outputStream.writeObject(stud1);
```

```
        outputStream.flush();
        outputStream.close();
        Student student = new Student("Mowgli", 300);
        FileInputStream inputFile = new FileInputStream("storage.dat");
        ObjectInputStream inputStream = new ObjectInputStream(inputFile);
        Student stud2 = (Student) inputStream.readObject();
        System.out.println(stud2);
        inputStream.close();
    }
}
```

Which statement about the program is true?

Select the one correct answer.

(a) It fails to compile.

(b) It compiles, but throws an exception at runtime.

(c) It prints (Aesop, 100, 1)(Aesop, 100, 1).

(d) It prints (Aesop, 100, 1)(null, 100, 2).

(e) It prints (Aesop, 100, 1)(null, 100, 1).

Answer

1. (b)

The method getName() can be used on a File object to return the name of the entry excluding the specification of the directory in which the entry resides.

2. (b) and (d)

Compiling and running the program results in the following output:

```
./documents/../book/../chapter1
/wrk/./documents/../book/../chapter1
/wrk/chapter1
chapter1
./documents/../book/..
```

3. (a) and (c)

Classes that implement the DataOutput interface, i.e., DataOutputStream and ObjectOutputStream, provide methods for writing binary representations of primitive values. The output stream classes FileOutputStream, PrintStream, and BufferedOutputStream do not provide such methods.

4. (c)

A ObjectOutputStream can write both objects and Java primitive types, as it implements the ObjectInput and the DataInput interfaces. The serialization mechanism will follow object references and can write whole hierarchies of objects.

5. (d)

During deserialization, the default constructor of the superclass Person is called, because this superclass is not Serializable.

6. (d)

The field name in the Person class is transient, and the field numOfStudents in the Student class is static. During serialization of a Student object, neither of these fields are serialized. After deserialization, the value of the field name is null, but the value of the static field numOfStudents has been incremented because a second Student object has been created.