Java IO Fundamentals - 0

Given code of Test.java file:

```
import java.io.*;
import java.time.LocalDate;
import java.util.Optional;

public class Test {
    public static void main(String[] args) throws IOException,
        ClassNotFoundException {
        Optional<LocalDate> optional =
        Optional.of(LocalDate.of(2018, 12, 1));
        try (ObjectOutputStream oos = new ObjectOutputStream(new
        FileOutputStream(("F:\date.ser")));
             ObjectInputStream ois = new ObjectInputStream(new
        FileInputStream("F:\date.ser")))
        {
            oos.writeObject(optional);

            Optional<?> object = (Optional<?>)ois.readObject();
            System.out.println(object.get());
        }
    }
}
```

F: is accessible for reading/writing and currently doesn't contain any files/directories.

What will be the result of compiling and executing Test class?

A - 01-12-2018

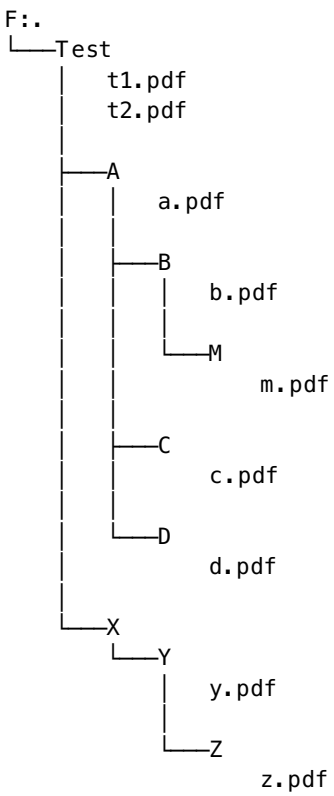
B - 2018-12-01

C - Compilation error

D - Runtime Exception

Java IO Fundamentals - 1

Below is the directory structure of “F:/Test” directory:



Given code of Test.java file:

```
import java.io.File;
import java.io.IOException;

public class Test {
    public static void main(String[] args) throws IOException {
        deleteFiles(new File("F:\Test"), ".pdf");
    }

    public static void deleteFiles(File dir, String extension)
        throws IOException {
        File[] list = dir.listFiles();
        if (list != null && list.length > 0) {
            for (File file : list) {
                if (file.isDirectory()) {
                    deleteFiles(file, extension);
                } else if (file.getName().endsWith(extension)) {
                    file.delete();
                }
            }
        }
```

```
    }
}
```

There is full permission to list/create/delete files and directories in F:.
What will be the result of compiling and executing Test class?

A - Only t1.pdf and t2.pdf will not get deleted, other pdf files will be successfully deleted

B - t1.pdf, t2.pdf and all the pdf files under ‘A’ and its sub-directories will be deleted successfully

C - All the pdf files in ‘Test’ directory and its sub-directories will be deleted successfully

D - Only t1.pdf and t2.pdf will get deleted

E - t1.pdf, t2.pdf and all the pdf files under ‘X’ and its sub-directories will be deleted successfully

Java IO Fundamentals - 2

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) throws IOException {
        try (FileInputStream fis = new
        FileInputStream("F:\orig.png");
             FileOutputStream fos = new
        FileOutputStream("F:\copy.png")) {
            int res;
            byte [] arr = new byte[500000]; //Line 10
            while((res = fis.read(arr)) != -1){ //Line 11
                fos.write(arr); //Line 12
            }
        }
    }
}
```

F: is accessible for reading/writing and contains ‘orig.png’ file.

Will above code create exact copy of ‘orig.png’ file?

A - No

B - Yes

Java IO Fundamentals - 3

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) throws IOException {
        File file = new File("F:\temp.dat");
        try(
                DataOutputStream os = new DataOutputStream(new
         FileOutputStream(file));
                DataInputStream is = new DataInputStream((new
         FileInputStream(file)))
        ) {
            os.writeChars("JAVA");
            System.out.println(is.readChar());
        }
    }
}
```

What will be the result of compiling and executing Test class?

A - A

B - None of the other options.

C - V

D - J

Java IO Fundamentals - 4

Given code of Test.java file:

```
import java.io.Console;
import java.util.Optional;

public class Test {
    public static void main(String[] args) {
        Optional<Console> optional =
         Optional.ofNullable(System.console());
        if(optional.isPresent()) {
            System.out.println(optional.get());
        }
    }
}
```

Which of the following statement are correct regarding above code?

A - Above code may throw NullPointerException.

B - Above code will never throw NullPointerException.

C - Above code will always print some output on to the console.

Java IO Fundamentals - 5

Imagine below path exists:

```
F:.
└───A
    └───B
```

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) {
        File dir = new File("F:" + File.separator + "A" +
         File.separator + "B");
        System.out.println(dir.getParentFile().getParent());
    }
}
```

What will be the result of compiling and executing Test class?

A - F:\

B - NullPointerException is thrown at runtime

C - F:\A

D - Compilation error

Java IO Fundamentals - 6

Given code of Test.java file:

```
import java.io.*;

class Counter implements Serializable {
    private static int count = 0;
    public Counter() {
        count++;
    }

    public static int getCount() {
        return count;
    }
}

public class Test {
    public static void main(String[] args) throws IOException,
        ClassNotFoundException {
        Counter ctr = new Counter();
        try( ObjectOutputStream oos = new ObjectOutputStream(
                new FileOutputStream("C:\Counter.dat")) ){
            oos.writeObject(ctr);
        }

        new Counter(); new Counter();

        try( ObjectInputStream ois = new ObjectInputStream(
                new FileInputStream("C:\Counter.dat")) ){
            ctr = (Counter)ois.readObject();
            System.out.println(Counter.getCount());
        }
    }
}
```

There is full permission to list/create/delete files and directories in C:.

What will be the result of compiling and executing Test class?

A - Runtime Exception

B - 1

C - 3

D - 2

Java IO Fundamentals - 7

Imagine below path exists:

```
F:.
└───A
    └───B
```

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) {
        File dir = new File("F:" + File.separator + "A" +
         File.separator + "B");
        System.out.println(/*INSERT*/);
    }
}
```

Which of the following replaces /*INSERT*/, such that on execution 'F:\' is displayed on to the console? Select 2 options.

A - dir.getParentFile().getParentFile()

B - dir.getParent().getParentFile( )

C - dir.getParent().getParent()

D - dir.getParentFile().getParent( )

Java IO Fundamentals - 8

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) throws IOException {
        BufferedReader br = new BufferedReader(new
         InputStreamReader(System.in));
        System.out.print("Enter any number between 1 and 10: ");
        int num = br.read();
        System.out.println(num);
    }
}
```

On execution above code prompts user with following message:

Enter any number between 1 and 10:

Which of the following statement is true if user types 2 and press Enter?

A - Runtime Exception

B - 2 will be printed on to the console.

C - 2 will not be printed on to the console.

Java IO Fundamentals - 9

Given code of Test.java file:

```
import java.io.IOException;
import java.io.PrintWriter;

public class Test {
    public static void main(String[] args) {
        try(PrintWriter bw = new PrintWriter("F:\test.txt"))
        {
            bw.close();
            bw.write(1);
        } catch(IOException e) {
            System.out.println("IOException");
        }
    }
}
```

F: is accessible for reading/writing purposes.

Which of the following statement is true about above code?

A - Class Test compiles and executes fine and no output is displayed on to the console

B - test.txt file will be successfully created and 1 will be written to it

C - On execution, IOException is printed on to the console

D - Compilation error

Java IO Fundamentals - 10

Which of the following will correctly accept and print the entered password on to the console?

A -

```
Console console = System.console();
String pwd = console.readPassword("Enter Password: ");
System.out.println(pwd);
```

B -

```
Console console = System.console();
char [] pwd = console.readPassword("Enter Password: ");
System.out.println(new String(pwd));
```

C -

```
Console console = new Console(System.in);
char [] pwd = console.readPassword("Enter Password: ");
System.out.println(new String(pwd));
```

D -

```
Console console = new Console(System.in);
String pwd = console.readPassword("Enter Password: ");
System.out.println(pwd);
```

Java IO Fundamentals - 11

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) throws IOException {
        BufferedWriter bw = new BufferedWriter(new
         FileWriter("F:\temp.tmp"));
        try(BufferedWriter writer = bw) { //Line 8

        } finally {
           bw.flush(); //Line 11
        }
    }
}
```

What will be the result of compiling and executing Test class?

A - Line 11 causes Runtime exception

B - Line 11 causes Compilation error

C - Line 8 causes Runtime exception

D - Line 8 causes Compilation error

Java IO Fundamentals - 12

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) {
        try(BufferedWriter bw = new BufferedWriter(new
        FileWriter("F:\test.txt")))
        {
            bw.close();
            bw.newLine();
        } catch(IOException e) {
            System.out.println("IOException");
        }
    }
}
```

F: is accessible for reading/writing purposes.

Which of the following statement is true about above code?

A - On execution, IOException is printed on to the console

B - Class Test compiles and executes fine and no output is displayed on to the console

C - Compilation error

Java IO Fundamentals - 13

Given code of Test.java file:

```
import java.io.*;

class Student {
    private String name;
    private int age;

    Student(String name, int age) {
        this.name = name;
        this.age = age;
    }

    public String getName() {
        return name;
    }

    public int getAge() {
        return age;
    }

    public void setName(String name) {
        this.name = name;
    }

    public void setAge(int age) {
        this.age = age;
    }
}

public class Test {
    public static void main(String[] args) throws IOException,
         ClassNotFoundException {
        Student stud = new Student("John", 20);

        try( ObjectOutputStream oos = new ObjectOutputStream(
                new FileOutputStream("C:\Student.dat")) ){
            oos.writeObject(stud);
        }

        try( ObjectInputStream ois = new ObjectInputStream(
                new FileInputStream("C:\Student.dat")) ){
            stud = (Student)ois.readObject();
            System.out.printf("%s : %d", stud.getName(),
         stud.getAge());
        }
    }
}
```

There is full permission to list/create/delete files and directories in C:.

What will be the result of compiling and executing Test class?

A - Runtime Exception

B - John : 20

C - Compilation error

D - null : 0

Java IO Fundamentals - 14

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) throws IOException {
        BufferedOutputStream bos = new BufferedOutputStream(
                new FileOutputStream("F:\file.tmp"));
        bos.write(2);
        bos.close();
    }
}
```

Which of the following statement regarding above code is true?

A - There is a chance of resource leak

B - Above code causes compilation error

C - There is no chance of resource leak

Java IO Fundamentals - 15

Below is the content of 'F:\message.txt': sdaleteftdeagncedk

message.txt file contains secret message received.

Below code is for decoding the secret message.

```
import java.io.*;

public class Test {
    public static void main(String[] args) throws IOException {
        try (InputStreamReader reader =
                    new InputStreamReader(new
         FileInputStream("F:\message.txt"))) {
            while (reader.ready()) {
                reader.skip(1);
                reader.skip(1);
                System.out.print((char) reader.read());
            }
        }
    }
}
```

What will be the result of compiling and executing Test class?

A - sledge

B - defend

C - None of the other options

D - attack

Java IO Fundamentals - 16

Given code of Test.java file:

```
public class Test {
    public static void main(String[] args) {
        System.out.format("A%nB%nC");
    }
}
```

What will be the result of compiling and executing Test class?

A - A

B - Runtime Exception

C -

```
A
B
C
```

D - ABC

Java IO Fundamentals - 17

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) throws IOException {
        File f1 = new File("F:\f1.txt");
        FileWriter fw = new FileWriter("F:\dir\f2.txt");
        PrintWriter pw = new PrintWriter("F:\f3.txt");
    }
}
```

F: is accessible for reading/writing and currently doesn't contain any files/directories.

On executing Test class, how many physical files will be created on the disc?

A - 2

B - 1

C - 3

D - 0

Java IO Fundamentals - 18

Given code of Test.java file:

```
public class Test {
    public static void main(String[] args) {
        System.out.printf("%2$d + %1$d", 10, 20);
    }
}
```

What will be the result of compiling and executing Test class?

A - 10 + 20

B - 30

C - 20 +10

D - None of the other options

Java IO Fundamentals - 19

F: is accessible for reading/writing. Currently there is no ‘err.log’ file under F:.

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) throws IOException {
        System.setOut(new PrintStream("F:\err.log"));
        try {
            System.out.println("ONE");
            System.out.println(1 / 0);
        } catch (ArithmeticException e) {
            System.err.println("TWO");
        }
    }
}
```

What will be the result of compiling and executing Test class?

A - err.log will be created but it will not have any texts inside.

B - No err.log file will be created.

C - err.log file will be created and it will contain following texts: ONE

D - err.log file will be created and it will contain following texts: TWO

E - err.log file will be created and it will contain following texts:

```
ONE
TWO
```

Java IO Fundamentals - 20

Given code of Test.java file:

```
import java.io.*;

class Student implements Serializable {
    private transient String name;
    private int age;

    Student(String name, int age) {
        this.name = name;
        this.age = age;
    }

    public String getName() {
        return name;
    }

    public int getAge() {
        return age;
    }

    public void setName(String name) {
        this.name = name;
    }

    public void setAge(int age) {
        this.age = age;
    }
}

public class Test {
    public static void main(String[] args) throws IOException,
         ClassNotFoundException {
        Student stud = new Student("John", 20);

        try( ObjectOutputStream oos = new ObjectOutputStream(
                new FileOutputStream("C:\Student.dat")) ){
            oos.writeObject(stud);
        }

        try( ObjectInputStream ois = new ObjectInputStream(
                new FileInputStream("C:\Student.dat")) ){
            stud = (Student)ois.readObject();
            System.out.printf("%s : %d", stud.getName(),
         stud.getAge());
        }
    }
}
```

There is full permission to list/create/delete files and directories in C:.

What will be the result of compiling and executing Test class?

A - null : 0

B - John : 20

C - Runtime Exception

D - Compilation error

E - null : 20

Java IO Fundamentals - 21

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) throws IOException {
        Console console = System.console();
        String name = console.readLine("What's your name? ");
        System.out.printf("You entered: %s", name);
    }
}
```

Which of the following statement is correct regarding above program, if it is executed from the command line?

A - It waits for 1 min for the user input and then terminates

B - It compiles fine but can cause NullPointerException at runtime

C - It waits indefinitely for the user input after displaying the text: What's your name?

D - It compiles fine and will never cause NullPointerException at runtime

E - It causes compilation failure

Java IO Fundamentals - 22

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) {
        File dirs = new File("F:\A\B\C");
        System.out.println(dirs.mkdirs());
        File dir = new File("F:\A");
        System.out.println(dir.mkdir());
        System.out.println(dir.delete());
    }
}
```

F: is accessible for reading/writing and currently doesn't contain any files/directories. What will be the result of compiling and executing Test class?

A -

```
false
true
true
```

B -

```
true
false
false
```

C -

```
false
false
false
```

D -

```
true
true
true
```

E -

```
true
false
true
```

Java IO Fundamentals - 23

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) {
        File dir = new File("F:" +
         System.getProperty("path.separator") + "A");
        dir.mkdir();
    }
}
```

F: is accessible for reading/writing and currently doesn't contain any files/directories.

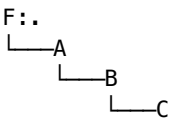
Will above code create directory 'A' inside F:?

A - No

B - Yes

Java IO Fundamentals - 24

F: is accessible for reading/writing and currently doesn't contain any directories.



Which of the code snippet allows to create below directory structure under F:?

A -

```
File file = new File("F:\A\B\C");
file.mkdirs();
```

B -

```
File file = new File("F:\A\B\C");
file.createNewDirectories();
```

C -

```
File file = new File("F:\A\B\C");
file.mkdir();
```

D -

```
File file = new File("F:\A\B\C");
file.createNewDirectory();
```

Java IO Fundamentals - 25

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) {
        Console console = System.console();
        if(console != null) {
            console.format("%d %<x", 10);
        }
    }
}
```

What will be the output of compiling and executing Test class from command prompt? javac Test.java java Test

A - 10

B - 10 10

C - 10 a

D - 10 12

E - Runtime Exception

Java IO Fundamentals - 26

Given code of Test.java file:

```
import java.io.*;

class Person {
    private String name;
    private int age;

    public Person(String name, int age) {
        this.name = name;
        this.age = age;
    }

    public String getName() {
        return name;
    }

    public int getAge() {
        return age;
    }
}

class Student extends Person implements Serializable {
    private String course;

    public Student(String name, int age, String course) {
        super(name, age);
        this.course = course;
    }

    public String getCourse() {
        return course;
    }
}

public class Test {
    public static void main(String[] args) throws IOException,
        ClassNotFoundException {
        Student stud = new Student("John", 20, "Computer Science");
        try (ObjectOutputStream oos = new ObjectOutputStream(new
        FileOutputStream(("F:\stud.ser")));
            ObjectInputStream ois = new ObjectInputStream(new
        FileInputStream("F:\stud.ser")))
        {
            oos.writeObject(stud);

            Student s = (Student) ois.readObject();
            System.out.printf("%s, %d, %s", s.getName(), s.getAge(),
        s.getCourse());
        }
    }
}
```

F: is accessible for reading/writing and currently doesn’t contain any files/directories.

What will be the result of compiling and executing Test class?

A - Runtime Exception

B - John, 20, Computer Science

C - null, 0, null

D - null, 0, Computer Science

Java IO Fundamentals - 27

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) throws IOException {
        File f1 = new File("F:\f1.txt");
        FileWriter fw = new FileWriter("F:\f2.txt");
        PrintWriter pw = new PrintWriter("F:\f3.txt");
    }
}
```

F: is accessible for reading/writing and currently doesn't contain any files/directories.

On executing Test class, how many physical files will be created on the disc?

A - 3

B - 0

C - 2

D - 1

Java IO Fundamentals - 28

Given code of Test.java file:

```
import java.io.*;

public class Test {
    public static void main(String[] args) {
        Console console = System.console();
        if(console != null) {
            console.format("%d %x", 10);
        }
    }
}
```

What will be the output of compiling and executing Test class from command prompt?

javac Test.java

java Test

A - Runtime Exception

B - 10 a

C - 10

D - 10 10

E - 10 12

Java IO Fundamentals - 29

Which of the following methods a class must implement/override to implement java.io.Serializable interface?

Select ALL that apply

A - public void serialize(Object);

B - public Object readObject();

C - None of the other options

D - public void writeObject(Object);

E - public Object deserialize();

Java IO Fundamentals - 30

Given code of Test.java file:

```
import java.io.*;

class Person {
    private String name;
    private int age;

    public Person(){}

    public Person(String name, int age) {
        this.name = name;
        this.age = age;
    }

    public String getName() {
        return name;
    }

    public int getAge() {
        return age;
    }
}

class Student extends Person implements Serializable {
    private String course;

    public Student(String name, int age, String course) {
        super(name, age);
        this.course = course;
    }

    public String getCourse() {
        return course;
    }
}

public class Test {
    public static void main(String[] args) throws IOException,
        ClassNotFoundException {
        Student stud = new Student("John", 20, "Computer Science");
        try (ObjectOutputStream oos = new ObjectOutputStream(new
        FileOutputStream(("F:\stud.ser")));
            ObjectInputStream ois = new ObjectInputStream(new
        FileInputStream("F:\stud.ser")))
        {
            oos.writeObject(stud);

            Student s = (Student) ois.readObject();
            System.out.printf("%s, %d, %s", s.getName(), s.getAge(),
        s.getCourse());
        }
```

```
    }
}
```

F: is accessible for reading/writing and currently doesn't contain any files/directories.

What will be the result of compiling and executing Test class?

A - null, 0, null

B - John, 20, Computer Science

C - Runtime Exception

D - null, 0, Computer Science

Java IO Fundamentals - 31

Given code of Test.java file:

```
import java.io.*;

class Student implements Serializable {
    private String name;
    private int age;

    Student(String name, int age) {
        this.name = name;
        this.age = age;
    }

    public String getName() {
        return name;
    }

    public int getAge() {
        return age;
    }

    public void setName(String name) {
        this.name = name;
    }

    public void setAge(int age) {
        this.age = age;
    }
}

public class Test {
    public static void main(String[] args) throws IOException,
         ClassNotFoundException {
        Student stud = new Student("John", 20);

        try( ObjectOutputStream oos = new ObjectOutputStream(
                new FileOutputStream("C:\Student.dat")) ){
            oos.writeObject(stud);
        }

        stud.setName("James");
        stud.setAge(21);

        try( ObjectInputStream ois = new ObjectInputStream(
                new FileInputStream("C:\Student.dat")) ){
            stud = (Student)ois.readObject();
            System.out.printf("%s : %d", stud.getName(),
         stud.getAge());
        }
    }
}
```

There is full permission to list/create/delete files and directories in C:.

What will be the result of compiling and executing Test class?

A - John : 20

B - Runtime Exception

C - James : 21

Java IO Fundamentals - 32

Given code of Test.java file:

```
public class Test {
    public static void main(String[] args) {
        System.out.printf("%2$d + %1$d", 10, 20, 30);
    }
}
```

What will be the result of compiling and executing Test class?

A - 30

B - 10 + 20

C - None of the other options

D - 20 + 10

H - In format string, format specifier are just replaced.

2$ means 2nd argument, which is 20 and 1$ means 1st argument, which is 10.

Hence 'System.out.printf("%2*d*+d", 10, 20);' prints '20 + 10' on to the console.

Having more arguments than the format specifiers is OK, extra arguments are ignored but having less number of arguments than format specifiers throws MissingFormatArgumentException at runtime.

NOTE: System.out.printf(...) is same as System.out.format(...).