

1. Given:

```
public abstract class Customer {
    private String name;
    public Customer (String name) {
        this.name = name;
    }
    public String getName() { return name; }
    public abstract void buy();
}
```

Which two statements are true about Customer?

- A) The Customer class cannot be extended.
- B) The Customer class cannot be instantiated.
- C) Subclasses of Customer cannot override getName() method.
- D) Concrete subclasses of Customer must use a default constructor.
- E) Concrete subclasses of Customer must implement the buy() method.
- F) Subclasses of Customer must implement the buy() method.

2. Given:

```
class Toy {
    double price;
    String color;
    Toy(String color, double price) {
        this.color = color;
        this.price = price;
    }
    public double getPrice() {
        return price;
    }
    public String getColor() {
        return color;
    }
}
```

And given the code fragment:

```
List<Toy> toys = new ArrayList<>();
toys.add(new Toy("red", 10));
toys.add(new Toy("yellow", 10));
toys.add(new Toy("red", 10));
double totalPrice = toys.stream()
    .filter(e -> e.getColor() == "red")
    /* Line n1 */
    .sum();
System.out.println("Total Price of Red Toys: " + totalPrice);
```

Which code fragment can be inserted at Line n1 to enable the code to print Total Price of Red Toys: 20.0?

- A) `.flatMap(e -> e.getPrice())`
- B) `.mapToDouble(e -> e.getPrice())`
- C) `.map(e -> e.getPrice())`
- D) `.peek(e -> e.getPrice())`

3. Given the code fragment:

```
class MyResource1 implements AutoCloseable {
    public void close() throws IOException {
        System.out.print("1 ");
    }
}
class MyResource2 implements Closeable {
    public void close() throws IOException {
        throw new IOException();
    }
}
public class TestRes {
    public static void main(String[] args) {
        try (MyResource1 r1 = new MyResource1();
            MyResource2 r2 = new MyResource2();) {
            System.out.print("T ");
        } catch (IOException ioe) {
            System.out.print("IOE ");
        } finally {
            System.out.print("F ");
        }
    }
}
```

What is the result?

- A) T 1 IOE F
- B) T IOE F
- C) T IOE 1 F
- D) Compilation fails.

4. Assuming that the `TestResult.txt` file exists and given the code fragment:

```
public class TestReadFile {  
  
    public void readFile(String fName) throws IOException {  
        // Line n1  
        Stream<String> content = Files.lines(p);  
        content.forEach(s1 -> System.out.println(s1));  
    }  
  
    public static void main(String[] args) throws IOException {  
        TestReadFile trf = new TestReadFile();  
        trf.readFile("TestResult.txt ");  
    }  
  
}
```

Which code fragment at Line n1 compiles?

- A) `Path p = new Path(fName);`
- B) `Path p = Paths.get(fName);`
- C) `Path p = Paths.toPath(fName);`
- D) `Path p = Paths.get(new File(fName));`

5. Which class definition compiles?

- A)

```
class CallerThread1 implements Callable<String> {  
    public String call() throws Exception { return "thread";}  
}
```
- B)

```
class CallerThread2 implements Callable {  
    public void call() {}  
}
```
- C)

```
class CallerThread3 extends Callable {  
    public void call() throws IOException {}  
}
```
- D)

```
class CallerThread4 implements Callable<String> {  
    public String call(String s) { return "thread";}  
}
```
- E)

```
class CallerThread5 extends Callable<String> {  
    public void callable(String s) throws Exception {}  
}
```

6. Given the code fragment:

```
Queue<String> products = new ArrayDeque<String>();  
products.add("p1");  
products.add("p2");  
products.add("p3");  
System.out.print(products.peek());  
System.out.print(products.poll());  
System.out.println("");  
products.forEach(s -> System.out.print(s));
```

What is the result?

- A. p1p1
p2p3
- B. p1p2
p1p2p3
- C. p1p2
p3
- D. p1p1
p1p2p3

7. Given the code fragment:

```
try (Connection con = DriverManager.getConnection(url, uname, pwd)) {  
    Statement stmt = con.createStatement();  
    System.out.print(stmt.executeUpdate("INSERT INTO Emp VALUES (500, 'Murray')"));  
}
```

Assuming the SQL query executes properly, what is the result?

- A) true
- B) false
- C) 1

D) 0

8. Given the code fragment:

```
public class TestFun {  
    public static void main(String[] args) {  
        List<Integer> nums = Arrays.asList(1,2,3,4,5);  
        // Line n1  
    }  
}
```

Which code fragment can be inserted at Line n1 to enable the code to print 2 4?

- A) `nums.peek(n -> n%2 == 0)`
 `.forEach(s -> System.out.print(" "+s));`
- B) `nums.filter(n -> n%2 == 0)`
 `.forEach(s -> System.out.print(" "+s));`
- C) `nums.map(n -> n%2 == 0)`
 `.forEach(s -> System.out.print(" "+s));`
- D) `nums.stream()`
 `.filter(n -> n%2 == 0)`
 `.forEach(s -> System.out.print(" "+s));`

Answers:

1. B and E
2. B
3. A
4. B
5. A
6. A
7. C
8. D