

<b>ExamCode: 120-808</b>
<b>ExamName: Java SE 8 Programmer I</b>
<b>Vendor Name: Oracle</b>
<b><u>Edition = DEMO</u></b>

## Question: 1

Given:

```
class Product {
    double price;
}

public class Test {
    public void updatePrice(Product product, double price) {
        price = price * 2;
        product.price = product.price + price;
    }
    public static void main(String[] args) {
        Product prt = new Product();
        prt.price = 200;
        double newPrice = 100;

        Test t = new Test();
        t.updatePrice(prt, newPrice);
        System.out.println(prt.price + " : " + newPrice);
    }
}
```

What is the result?

- A. 200.0 : 100.0
- B. 400.0 : 200.0
- C. 400.0 : 100.0
- D. Compilation fails.

<b>Answer: B</b>
------------------

## Question: 2

Given the code fragment:

```
if (aVar++ < 10) {  
    System.out.println(aVar + " Hello World!");  
} else {  
    System.out.println(aVar + " Hello Universe!");  
}
```

What is the result if the integer aVar is 9?

- A. Hello World!
- B. Hello Universe!
- C. Hello World!
- D. Compilation fails.

**Answer: A**

### Question: 3

Given the code fragment:

```
public static void main(String[] args) {  
    String date = LocalDate  
        .parse("2014-05-04")  
        .format(DateTimeFormatter.ISO_DATE_TIME);  
    System.out.println(date);  
}
```

What is the result?

- A. May 04, 2014T00:00:00.000
- B. 2014-05-04T00:00: 00. 000
- C. 5/4/14T00:00:00.000
- D. An exception is thrown at runtime.

**Answer: B**

Explanation:

Reference:

<https://docs.oracle.com/javase/8/docs/api/java/time/format/DateTimeFormatter.html>  
(see predefined formatters)

### Question: 4

Given the code fragment:

```
public static void main(String[] args) {  
    Short s1 = 200;  
    Integer s2 = 400;  
    Long s3 = (long) s1 + s2;           //line n1  
    String s4 = (String) (s3 * s2);    //line n2  
    System.out.println("Sum is " + s4);  
}
```

What is the result?

- A. Sum is 600
- B. Compilation fails at line n1.
- C. Compilation fails at line n2.
- D. A ClassCastException is thrown at line n1.
- E. A ClassCastException is thrown at line n2.

**Answer: E**

### Question: 5

What is the name of the Java concept that uses access modifiers to protect variables and hide them within a class?

- A. Encapsulation
- B. Inheritance
- C. Abstraction
- D. Instantiation
- E. Polymorphism

**Answer: A**

Explanation:

Using the private modifier is the main way that an object encapsulates itself and hide data from the outside world.

Explanation:

Reference:

[http://www.tutorialspoint.com/java/java\\_access\\_modifiers.htm](http://www.tutorialspoint.com/java/java_access_modifiers.htm)

### Question: 6

Given the code fragment:

```
abstract class Planet {  
    protected void revolve() {           //line n1  
    }  
  
    abstract void rotate();              //line n2  
}  
  
class Earth extends Planet {  
    void revolve() {                     //line n3  
    }  
  
    protected void rotate() {           //line n4  
    }  
}
```

Which two modifications, made independently, enable the code to compile?

- A. Make the method at line n1 public.
- B. Make the method at line n2 public.
- C. Make the method at line n3 public.
- D. Make the method at line n3 protected.
- E. Make the method at line n4 public.

<b>Answer: B, C</b>
---------------------

## Question: 7

Given:

```

class Vehicle {
    String type = "4W";
    int maxSpeed = 100;

    Vehicle(String type, int maxSpeed) {
        this.type = type;
        this.maxSpeed = maxSpeed;
    }
}

class Car extends Vehicle {
    String trans;

    Car(String trans) {                //line n1
        this.trans = trans;
    }

    Car(String type, int maxSpeed, String trans) {
        super(type, maxSpeed);
        this.trans = trans;           //line n2
    }
}

```

And given the code fragment:

```

7. Car c1 = new Car("Auto");
8. Car c2 = new Car("4W", 150, "Manual");
9. System.out.println(c1.type + " " + c1.maxSpeed + " " + c1.trans);
10. System.out.println(c2.type + " " + c2.maxSpeed + " " + c2.trans);

```

What is the result?

- A. 4W 100 Auto  
4W 150 Manual
- B. Null 0 Auto  
4W 150 Manual
- C. Compilation fails only at line n1
- D. Compilation fails only at line n2
- E. Compilation fails at both line n1 and line n2

<b>Answer: B</b>
------------------

## Question: 8

Given the code fragment:

```

1. class X {
2.     public void printFileContent() {
3.         /* code goes here */
4.         throw new IOException();
5.     }
6. }
7. public class Test {
8.     public static void main(String[] args) {
9.         X xobj = new X();
10.        xobj.printFileContent();
11.    }
12. }

```

Which two modifications should you make so that the code compiles successfully?

- ☐ A) Replace line 8 with `public static void main(String[] args) throws Exception {`
- ☐ B) Replace line 10 with:
 

```

      try {
          xobj.printFileContent();
      }
      catch(Exception e) { }
      catch(IOException e) { }
      
```
- ☐ C) Replace line 2 with `public void printFileContent() throws IOException {`
- ☐ D) Replace line 4 with `throw new IOException("Exception raised");`
- ☐ E) At line 11, insert `throw new IOException();`

- A. Option A
- B. Option B
- C. Option C
- D. Option D
- E. Option E

<b>Answer: D, E</b>
---------------------

## Question: 9

Given the following two classes:

```
public class Customer {
    ElectricAccount acct = new ElectricAccount();

    public void useElectricity(double kWh) {
        acct.addKWh(kWh);
    }
}

public class ElectricAccount {
    private double kWh;
    private double rate = 0.07;
    private double bill;

    //line n1
}
```

How should you write methods in the ElectricAccount class at line n1 so that the member variable bill is always equal to the value of the member variable kWh multiplied by the member variable rate? Any amount of electricity used by a customer (represented by an instance of the customer class) must contribute to the customer's bill (represented by the member variable bill) through the method useElectricity method. An instance of the customer class should never be able to tamper with or decrease the value of the member variable bill.

- ☐ A) 

```
public void addKWh(double kWh) {
    this.kWh += kWh;
    this.bill = this.kWh*this.rate;
}
```
- ☐ B) 

```
public void addKWh(double kWh) {
    if (kWh > 0){
        this.kWh += kWh;
        this.bill = this.kWh * this.rate;
    }
}
```
- ☐ C) 

```
private void addKWh(double kWh) {
    if (kWh > 0) {
        this.kWh += kWh;
        this.bill = this.kWh*this.rate;
    }
}
```
- ☐ D) 

```
public void addKWh(double kWh) {
    if(kWh > 0) {
        this.kWh += kWh;
        setBill(this.kWh);
    }
}

public void setBill(double kWh) {
    bill = kWh*rate;
}
```

- A. Option A  
B. Option B  
C. Option C  
D. Option D

<b>Answer: A, C</b>
---------------------

### Question: 10

Given the code fragment:



```
public static void main(String[] args) {  
    StringBuilder sb = new StringBuilder(5);  
    String s = "";  
  
    if (sb.equals(s)) {  
        System.out.println("Match 1");  
    } else if (sb.toString().equals(s.toString())) {  
        System.out.println("Match 2");  
    } else {  
        System.out.println("No Match");  
    }  
}
```

What is the result?

- A. Match 1
- B. Match 2
- C. No Match
- D. A NullPointerException is thrown at runtime.

<b>Answer: B</b>
------------------