

6406531891740. ✖

```
1 def add(self, rat):
2     out = Rational(1, 1)
3     out.num = self.num + rat.num
4     out.den = self.den + rat.den
5     return out
```

6406531891741. ✖

```
1 def add(self, rat):
2     out = Rational(1, 1)
3     out.num = self.num + rat.num
4     out.den = self.den + rat.den
5     out.reduce()
6     return out
```

6406531891742. ✖

```
1 def add(rat):
2     out = Rational(1, 1)
3     out.den = self.den * rat.den
4     out.num = self.num * rat.den + rat.num * self.den
5     out.reduce()
6     return out
```

6406531891743. ✔

```
1 def add(self, rat):
2     out = Rational(1, 1)
3     out.den = self.den * rat.den
4     out.num = self.num * rat.den + rat.num * self.den
5     out.reduce()
6     return out
```

Java

Section Id :	64065338401
Section Number :	4
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	24
Number of Questions to be attempted :	24

Section Marks :	100
Display Number Panel :	Yes
Group All Questions :	No
Enable Mark as Answered Mark for Review and Clear Response :	Yes
Maximum Instruction Time :	0
Sub-Section Number :	1
Sub-Section Id :	64065380933
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Number : 74 Question Id : 640653566045 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 0

Question Label : Multiple Choice Question

THIS IS QUESTION PAPER FOR THE SUBJECT "DIPLOMA LEVEL : PROGRAMMING CONCEPTS USING JAVA (COMPUTER BASED EXAM)"

ARE YOU SURE YOU HAVE TO WRITE EXAM FOR THIS SUBJECT?
CROSS CHECK YOUR HALL TICKET TO CONFIRM THE SUBJECTS TO BE WRITTEN.

(IF IT IS NOT THE CORRECT SUBJECT, PLS CHECK THE SECTION AT THE [TOP](#) FOR THE SUBJECTS REGISTERED BY YOU)

Options :

6406531891744.  YES

6406531891745.  NO

Sub-Section Number :	2
Sub-Section Id :	64065380934
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 75 Question Id : 640653566046 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the Java code given below.

```
1 class Example{
2     public void functionOne(){
3         // ...
4         functionTwo();
5         // ...
6     }
7     public void functionTwo(){
8         // ...
9     }
10    public static void functionThree(){
11        // ...
12        Example e = new Example();
13        e.functionOne();
14        // ...
15    }
16    public static void main(String[] args) {
17        // ...
18        functionThree();
19    }
20 }
```

During execution of Line 14 in the above code, the activation record of which method is at the top of the stack of activation records?

Options :

6406531891746. ✖ main

6406531891747. ✖ functionOne

6406531891748. ✖ functionTwo

6406531891749. ✓ functionThree

Question Number : 76 Question Id : 640653566047 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the following code.

```
class ContactInfo implements Cloneable{
    String email;
    // Constructor
    // Accessor method getEmail()
    // Mutator method setEmail()
    public ContactInfo clone() throws CloneNotSupportedException{
        return (ContactInfo)super.clone();
    }
}

class Student implements Cloneable{
    String name;
    ContactInfo ci;
    // Constructor
    // Mutator method setName()
    public Student clone() throws CloneNotSupportedException{
        Student s = (Student)super.clone();
        s.ci = s.ci.clone();
        return s;
    }
    public String toString(){
        return name + ":" + ci.getEmail();
    }
}

public class Test {
    public static void main(String[] args) {
        Student s1 = new Student("Rahul", new ContactInfo("mail"));
        try{
            Student s2 = s1.clone();
            s2.ci.setEmail("new mail");
            s2.setName("Sreeja");
            System.out.println(s1);
            System.out.println(s2);
        }
        catch(Exception e){
            System.out.println(e);
        }
    }
}
```

What will the output be?

Options :

6406531891750. ✓
Rahul:mail
Sreeja:new mail

6406531891751. ✗
Rahul:new mail
Sreeja:new mail

6406531891752. ✖ Sreeja:new mail
Sreeja:new mail

6406531891753. ✖ Sreeja:mail
Sreeja:new mail

Question Number : 77 Question Id : 640653566049 Question Type : MCQ Is Question

**Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction
Time : 0**

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the code given below

```
class Shape{
    public void area(){
        System.out.println("Area exists");
    }
    public void sides(){
        System.out.println("Sides may exist");
    }
}
class Circle extends Shape{
    public void area(){
        System.out.println("Circle area");
    }
}
class Rectangle extends Shape{
    public void area(){
        System.out.println("Rectangle area");
    }
    public void sides(){
        System.out.println("Four sides");
    }
}
public class Test{
    static void display(Shape[] s){
        for(int i = 0; i < s.length; i++){
            s[i].area();
            s[i].sides();
        }
    }
    public static void main(String[] args){
        Shape[] s = {new Circle(), new Rectangle()}; // LINE 1
        display(s);
    }
}
```

Choose the correct option.

Options :

This code generates the output:

Rectangle area

Four sides

Rectangle area

Four sides

6406531891758. ✖

This code generates the output:

```
Area exists  
Sides may exist  
Area exists  
Sides may exist
```

6406531891759. ✖

This code generates the output:

```
Circle area  
Sides may exist  
Rectangle area  
Four sides
```

6406531891760. ✔

Compilation error at LINE 1 because a reference variable of type Shape cannot refer to an object of class Rectangle

6406531891761. ✖

Question Number : 78 Question Id : 640653566050 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the code given below that checks whether two laptops are the same. Method `equals` is overridden to compare two `Laptop` objects as follows: If two laptops have the same serial number, then they are the same. Based on the given information, answer the question that follows.

```
class Laptop{
    private String sno;

    //Constructor to initialize instance variable

    public boolean equals(Object obj) {
        // CODE BLOCK
    }
}

public class Test {
    public static void main(String[] args) {
        Laptop l1 = new Laptop("A001");
        Laptop l2 = new Laptop("A001");
        if(l1.equals(l2))
            System.out.println("l1 and l2 are the same");
        else
            System.out.println("l1 and l2 are not same");
    }
}
```

Choose the correct option to fill in place of `CODE BLOCK` so that the output is:

l1 and l2 are the same

Options :

```
if(obj instanceof Laptop) {
    if(this.sno.equals(obj.sno))
        return true;
}
```

6406531891762. ✖ return false;

```
if(obj instanceof Laptop) {
    Laptop l = (Laptop) obj;
    if(this.sno.equals(l.sno))
        return true;
}
```

6406531891763. ✔ return false;

6406531891764. ✖

```
if(obj instanceof Laptop) {  
    Laptop l = obj;  
    if(this.sno.equals(l.sno))  
        return true;  
}  
return false;
```

```
        if(this.sno.equals(obj.sno))  
            return true;  
        return false;
```

6406531891765. ✖

Question Number : 79 Question Id : 640653566054 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the Java code given below, and answer the question that follows:

```
abstract class Op implements Runnable{
    static int mul = 1;
}
class Op1 extends Op{
    public void run() {
        if (mul != 3) {
            mul = mul * 2;
        }
    }
}
class Op2 extends Op{
    public void run() {
        if (mul != 2) {
            mul = mul * 3;
        }
    }
}
public class RaceCondition {
    public static void main(String[] args) {
        Op o1 = new Op1();
        Op o2 = new Op2();

        Thread t1 = new Thread(o1);
        Thread t2 = new Thread(o2);

        t1.start();
        t2.start();

        System.out.println(Op.mul);
    }
}
```

Choose the correct option.

Options :

6406531891778. ✖ The program will never generate the output: 1

6406531891779. ✖ The program will never generate the output: 3

6406531891780. ✖ The program will never generate the output: 2

6406531891781. ✖ The program will never generate the output: 6

6406531891782. ✔ The output can be 1 or 3 or 2 or 6.

Question Number : 80 Question Id : 640653566056 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the code given below.

```
class Rectangle{
    double length, width;
    public Rectangle(double l, double w) {
        length = l;
        width = w;
    }
    public Rectangle(Rectangle r){
        length = r.length;
        width = r.width;
    }
    public double perimeter(){
        return 2 * (length + width);
    }
}

public class ConTest {
    public static void main(String[] args) {
        Rectangle r1 = new Rectangle(6.0, 3.20);
        Rectangle r2 = new Rectangle(r1);
        Rectangle r3 = new Rectangle(r2);
        r1.width = 3.0;
        System.out.println(r1.perimeter());
        System.out.println(r2.perimeter());
        System.out.println(r3.perimeter());
    }
}
```

What will the output be?

Options :

6406531891787.

18.4
18.0
✖ 18.0

18.0
18.0
6406531891788. ✖ 18.0

18.0
18.4
6406531891789. ✔ 18.4

18.4
18.4
6406531891790. ✖ 18.4

Question Number : 81 Question Id : 640653566060 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0
Correct Marks : 4
Question Label : Multiple Choice Question

Consider the code given below.

```
import java.util.*;
public class MapTest {
    public static void main(String[] args) {
        var bowler= new TreeMap<String,Integer>();

        bowler.put("Umran", 5);
        bowler.put("Starc", 5);

        bowler.put("Starc", 3);
        bowler.put("Umran", 3 + bowler.getDefault("Umran", 0));

        for(Map.Entry<String, Integer> obj1:bowler.entrySet())
            System.out.println(obj1.getKey()+":"+obj1.getValue());
    }
}
```

What will the output be?

Options :

6406531891803. ✖ `Starc:3
Umran:3`

6406531891804. ✔ `Starc:3
Umran:8`

6406531891805. ✖ `Starc:8
Umran:8`

6406531891806. ✖ `Starc:5
Umran:5`

Question Number : 82 Question Id : 640653566061 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the Java code given below.

```
class InvalidIncomeException extends Exception{
    public InvalidIncomeException(String str) {
        super(str);
    }
}
class TaxPayer{
    String name;
    double income;
    public TaxPayer(String n, double i) {
        name = n;
        income = i;
    }
    public double getTax() throws InvalidIncomeException{
        if(income < 0)
            throw new InvalidIncomeException("Invalid income");
        else if(income > 700000)
            return 0.1*income;
        else
            return 0;
    }
}

public class ExceptionTest {
    public static void main(String[] args) {
        var obj1 = new TaxPayer("ABC", -700000);
        var obj2 = new TaxPayer("XYZ", 500000);
        try {
            System.out.println(obj1.getTax());
            System.out.println(obj2.getTax());
        } catch (InvalidIncomeException e) {
            System.out.println(e.getMessage());
        }
    }
}
```

What will the output be?

Options :

Invalid income

6406531891807. ✖ 0.0

6406531891808. ✔ Invalid income

0.0

6406531891809. ✖ Invalid income

0.0

6406531891810. ✖ 0.0

Question Number : 83 Question Id : 640653566062 Question Type : MCQ Is Question

**Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction
Time : 0**

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the Java code given below.

```
import java.io.*;
class MobilePay implements Serializable{
    private String userNo = "9999999999";
    private transient int pin = 0000;
    private String upiID = "9999999999@ybl";
    public MobilePay(String uno, int p, String id) {
        userNo = uno;
        pin = p;
        upiID = id;
    }
    public String toString() {
        return userNo + ", " + pin + ", " + upiID;
    }
}
public class SerialTest{
    public static void main(String[] args) throws Exception{
        var fos = new FileOutputStream("upi.txt");
        var os = new ObjectOutputStream(fos);
        os.writeObject(new MobilePay("9977665544", 5432, "9977665544@ybl"));

        var fis = new FileInputStream("upi.txt");
        var ois = new ObjectInputStream(fis);
        MobilePay mPay = (MobilePay)ois.readObject();
        System.out.println(mPay);
    }
}
```

What will the output be?

Options :

6406531891811. ✓ 9977665544, 0, 9977665544@ybl

6406531891812. ✖ 9977665544, 0000, 9977665544@ybl

6406531891813. ✖ 9977665544, 5432, 9977665544@ybl

6406531891814. ✖ null, 0, null

6406531891815. ✖ 9999999999, 0000, 9999999999@ybl

Question Number : 84 Question Id : 640653566063 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the code given below.

```
import java.util.*;
class Television{
    TreeMap<String, String> f = new TreeMap<String, String>();
    public Television() {
        f.put("LG", "Korea");
        f.put("Toshiba", "Japan");
    }
    public String getManufacturer(String t){
        return f.get(t);
    }
}
public class OptionalTest {
    public static void main(String[] args){
        Optional<String> op1 = Optional.ofNullable(new Television()
                                                .getManufacturer("Toshiba"));
        Optional<String> op2 = Optional.ofNullable(new Television()
                                                .getManufacturer("lg"));
        op1.ifPresent(n->System.out.println(n.toUpperCase()));
        op2.ifPresent(n->System.out.println(n.toUpperCase()));
    }
}
```

Choose the correct option.

Options :

This program terminates due to NullPointerException after printing the message:

6406531891816. ✖ JAPAN

6406531891817. ✖

This program generates the output:

JAPAN

null

6406531891818. ✖ This program generates no output.

This program generates the output:

6406531891819. ✔ JAPAN

Question Number : 85 Question Id : 640653566064 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the code given below.

```
import java.util.*;
import java.util.stream.*;
class Student{
    String id;
    double cgpa;
    //Constructor to initialize instance variables
    //Method toString() to return id of the student
}
public class CollectingTest {
    public static void main(String[] args){
        var sArr = new ArrayList<Student>();
        sArr.add(new Student("ID6789", 9.75));
        sArr.add(new Student("ID5345", 6.75));
        sArr.add(new Student("ID4378", 8.00));
        sArr.add(new Student("ID7790", 8.45));
        Map<Boolean, List<Student>> stuMap;
        stuMap = sArr.stream()
            .collect(Collectors.partitioningBy(s->s.cgpa >= 8.00));
        System.out.println(stuMap.get(true));
        System.out.println(stuMap.get(false));
    }
}
```

What will the output be?

Options :

6406531891820. ✖

[ID5345]

[ID6789, ID4378, ID7790]

6406531891821. ✖

[]

[ID5345]

6406531891822. ✔

[ID6789, ID4378, ID7790]

[ID5345]

6406531891823. ✖

[ID6789, ID4378, ID7790]

[]

Question Number : 86 Question Id : 640653566065 Question Type : MCQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the code given below.

Assume that file "data.txt" exists and contains the following text.

I love my country

Assume that there is no file named "newdata.txt".

```
import java.io.*;
import java.util.Scanner;
public class FileTest {
    public static void main(String[] args) throws Exception{
        try {
            FileInputStream in = new FileInputStream("data.txt");
            Scanner sc = new Scanner(in);
            String data = "";
            if(sc.hasNext())
                data = sc.nextLine();

            var out = new FileOutputStream("newdata.txt", false);
            var dout = new DataOutputStream(out);
            dout.writeBytes(data);
            System.out.println("Data written to newdata.txt successfully");
            out.close();
            dout.close();
            sc.close();
        }
        catch (FileNotFoundException e) {
            System.out.println("File not found.....");
        }
        catch (IOException e) {
            System.out.println("Sorry there is an IO Error");
        }
    }
}
```

Choose the correct option regarding the code.

Options :

6406531891824. ✖ This program terminates abnormally due to unhandled exceptions.

6406531891825. ✓ This program prints the message:
Data written to newdata.txt successfully

6406531891826. ✗ This program prints the message:
File not found.....

6406531891827. ✗ This program prints the message:
Sorry there is an IO Error

Question Number : 87 Question Id : 640653566066 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the Java code given below.

```
class Triangle{
    private double base, height;
    // Constructor Triangle(double b, double h) to initialize base, height
    public double getArea() {
        assert base > 0; //LINE 1
        assert height > 0; //LINE 2
        double area = 0.5 * base * height;
        assert area > 0; //LINE 3
        return area;
    }
}

public class AssertionTest {
    public static void main(String[] args) {
        Triangle obj = new Triangle(3.0, -4.0);
        System.out.println(obj.getArea());
    }
}
```

Choose the correct option when the program is executed as:
java -ea AssertionTest

Options :

6406531891828. ✖ This program throws AssertionError at LINE 2 after printing the value:
-6.0
6406531891829. ✖ This program throws AssertionError at LINE 1.
6406531891830. ✖ This program throws AssertionError at LINE 3 after printing the value:
-6.0
6406531891831. ✔ This program throws AssertionError at LINE 2.

Question Number : 88 Question Id : 640653566067 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the Java program given below.

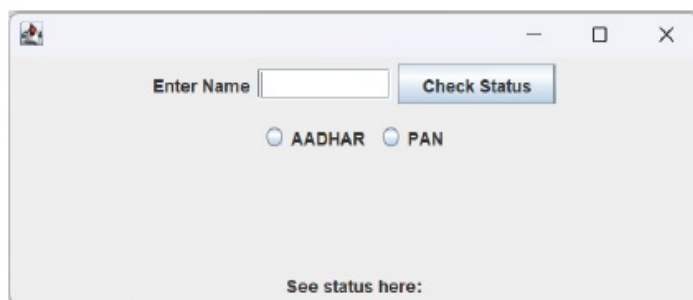
```
import javax.swing.*;
public class GUITest extends JFrame{
    JPanel fPanel, sPanel, tPanel;
    JLabel name, status;
    JRadioButton aadhar, pan;
    JTextField input;
    JButton button;
    public GUITest() {
        name = new JLabel("Enter Name");
        input = new JTextField(10);
        button = new JButton("Check Status");
        fPanel = new JPanel();
        // name, input and button are added to fPanel

        aadhar = new JRadioButton("AADHAR");
        pan = new JRadioButton("PAN");
        sPanel = new JPanel();
        // aadhar and pan are added to sPanel

        status = new JLabel("See status here:");
        tPanel = new JPanel();
        // status is added to tPanel

        setVisible(true);
        //CODE BLOCK
    }
    public static void main(String[] args) {
        new GUITest();
    }
}
```

Choose the correct option to be filled in place of CODE BLOCK such that the above program produces the GUI given below:



Options :

```
add(tPanel, "Center");
add(fPanel, "North");
add(sPanel, "South");
```

6406531891832. ✖

6406531891833. ✔

```
add(tPanel, "South");  
add(sPanel, "Center");  
add(fPanel, "North");
```

```
add(sPanel, "Center");  
add(tPanel, "East");  
add(fPanel, "West");
```

6406531891834. ✖

```
add(sPanel, "Center");  
add(tPanel, "West");  
add(fPanel, "East");
```

6406531891835. ✖

Question Number : 89 Question Id : 640653566068 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 4

Question Label : Multiple Choice Question

Consider the code given below.

```
class Example extends Thread{
    Thread t;
    Example(Thread t){
        this.t = t;
    }
    public void run(){
        try {
            t.join();
        } catch (InterruptedException ex) {

        }
        for(int i = 1; i < 3; i++){
            System.out.print(i + " ");
        }
    }
}

public class FClass{
    public static void main(String[] args) throws InterruptedException{
        Thread t1 = Thread.currentThread();
        Example t2 = new Example(t1);
        t2.start();
        for(int i = 3; i <= 5; i++){
            System.out.print(i + " ");
        }
    }
}
```

What will be the possible output?

Options :

6406531891836. ✖ 1 2 3 4 5

6406531891837. ✔ 3 4 5 1 2

6406531891838. ✖ 3 4 5

6406531891839. ✖ All of these

Sub-Section Id : 64065380935

Question Shuffling Allowed : Yes

Is Section Default? : null

Question Number : 90 Question Id : 640653566057 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 5

Question Label : Multiple Choice Question

Consider the code given below.

```
interface Chargable{
    void charge();
}
interface Supplyable{
    void supply();
}
class Bike{
    public Battery getBattery() {
        return new Battery();
    }
    private class Battery implements Chargable,Supplyable {
        public void charge() {
            System.out.println("Charges automatically while you ride");
        }
        public void supply() {
            System.out.println("Supplies current the bike requires");
        }
    }
}
public class PrivateTest {
    public static void main(String[] args) {
        //CODE BLOCK
    }
}
```

Choose the correct option to be filled in place of CODE BLOCK so that the output is:

Charges automatically while you ride
Supplies current the bike requires

Options :

6406531891791. ❌


```
Chargable obj1 = new Bike().getBattery();  
obj1.charge();  
obj1.supply();
```

```
Supplyable obj1 = new Bike().getBattery();  
obj1.charge();  
6406531891792. ✖ obj1.supply();
```

```
Battery obj1 = new Bike().getBattery();  
obj1.charge();  
6406531891793. ✖ obj1.supply();
```

```
Chargable obj1 = new Bike().getBattery();  
Supplyable obj2 = new Bike().getBattery();  
obj1.charge();  
6406531891794. ✔ obj2.supply();
```

Sub-Section Number :	4
Sub-Section Id :	64065380936
Question Shuffling Allowed :	Yes
Is Section Default? :	null

Question Number : 91 Question Id : 640653566048 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 5 Selectable Option : 0

Question Label : Multiple Select Question

Consider the code given below.

```
1 interface Travelable{
2     void travel();
3 }
4 interface Flyable{
5     void fly();
6 }
7 interface Seatable extends Travelable, Flyable{
8     void seat();
9 }
10 abstract class Vehicle implements Seatable {
11     public void seat(){
12         System.out.println("Vehicle seat");
13     }
14 }
15 class Airplane extends Vehicle{
16     public void travel(){
17         System.out.println("Plane travel");
18     }
19     public void fly(){
20         System.out.println("Plane fly");
21     }
22 }
23 public class Test{
24     public static void main(String[] args){
25         Travelable t = new Airplane();
26         t.travel();
27     }
28 }
```

Choose the correct option/s

Options :

6406531891754. ✖ Compilation error at LINE 7 because an interface cannot *extend* two interfaces.

6406531891755. ✔ The program generates the output: Plane travel

6406531891756. ✖ Compilation error in LINE 10 to LINE 14 because the abstract class Vehicle cannot implement an interface.

6406531891757. ✖ Compilation error at LINE 25 because a reference variable of interface Travelable cannot refer to an object of class Airplane.

Question Number : 92 Question Id : 640653566051 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 5 Selectable Option : 0

Question Label : Multiple Select Question

Consider the Java code given below that prints the recharge time of each charger in a list of chargers. From among the options, identify the appropriate function header for function display that takes as input a list of `Charger` objects and prints the recharge time of each.

```
import java.util.*;
abstract class Charger{
    abstract void rechargeTime();
}
class USBCharger extends Charger{
    public void rechargeTime() {
        System.out.println("USB recharge time");
    }
}
class LightningCharger extends Charger{
    public void rechargeTime() {
        System.out.println("Lightning charger recharge time");
    }
}
public class Test {
    // LINE 1: FUNCTION HEADER
    {
        // invokes method rechargeTime()
        // to print the recharge time of each charger
    }
    public static void main(String[] args) {
        List<Charger> clist = new ArrayList<Charger>();
        clist.add(new USBCharger());
        clist.add(new LightningCharger());
        display(clist);
    }
}
```

Choose the correct option(s).

Options :

6406531891766.

✖ public static void display(List<USBCharger> obj)

6406531891767. ✖ public static void display(List<LightningCharger> obj)

6406531891768. ✔ public static <T extends Charger> void display(List<T> obj)

6406531891769. ✔ public static void display(List<? extends Charger> obj)

Question Number : 93 Question Id : 640653566052 Question Type : MSQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 5 Selectable Option : 0

Question Label : Multiple Select Question

Which of the following statements counts the number of integers between 300 and 350 (including 300 and 350) that are not divisible by 3?

Options :

6406531891770. ✔ Stream.iterate(300, n -> n + 1).limit(51).filter(n -> n % 3 != 0).count();

6406531891771. ✖ Stream.iterate(300, n -> n + 1).filter(n -> n % 3 != 0).count();

6406531891772. ✖ Stream.filter(300, n-> n < 350, n -> n + 1).iterate(n -> n % 3 == 0).count();

6406531891773. ✔ Stream.iterate(300, n-> n <= 350, n -> n + 1).filter(n -> n % 3 != 0).count();

Question Number : 94 Question Id : 640653566053 Question Type : MSQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 5 Selectable Option : 0

Question Label : Multiple Select Question

Consider the code given below.

```
class Stock{
    int available = 40;
    public synchronized void request(String uname, int n){
        if(available >= n){
            available = available - n;
            System.out.println(uname + " ordered " + n + " item(s)");
        }
        else
            System.out.println(uname + " cannot order " + n + " item(s)");
    }
}

class Order implements Runnable{
    Stock st;
    String username;
    int nItems;
    public Order(Stock s, String u, int n){
        st = s;
        username = u;
        nItems = n;
    }
    public void run(){
        st.request(username, nItems);
    }
}

public class Test {
    public static void main(String[] args) {
        Stock stk = new Stock();
        Order od1 = new Order(stk, "Ram", 15);
        Order od2 = new Order(stk, "Sita", 30);
        Thread t1 = new Thread(od1);
        Thread t2 = new Thread(od2);
        t1.start();
        t2.start();
    }
}
```

From among the options given below, which are possible outputs of the given code?

Options :

6406531891774. ✓ Sita ordered 30 item(s)
Ram cannot order 15 item(s)

6406531891775. ✗ Ram ordered 15 item(s)
Sita ordered 30 item(s)

6406531891776. ✗ Sita cannot order 30 item(s)
Ram cannot order 15 item(s)

6406531891777. ✓ Ram ordered 15 item(s)
Sita cannot order 30 item(s)

Question Number : 95 Question Id : 640653566055 Question Type : MSQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 5 Selectable Option : 0

Question Label : Multiple Select Question

Consider the code given below.

```
import java.util.*;
class Student{
    private String name;
    private int total;
    public Student(String n, int t) {
        name = n;
        total = t;
    }
    public int getTotal(){
        return total;
    }
    public void print() {
        System.out.println(name + " : " + total);
    }
}

public class FClass{
    public static void main(String[] args) {
        var list = new ArrayList<Student>();
        list.add(new Student("Robin", 133));
        list.add(new Student("Indra", 176));
        list.add(new Student("Smita", 135));
        list.add(new Student("Rikki", 126));
        Collections.sort(list, _____); LINE 1
        for(var l: list)
            l.print();
    }
}
```

Identify the appropriate option(s) to fill in the blank at LINE 1, such that the output is:

Rikki : 126
Robin : 133
Smita : 135
Indra : 176

Options :

6406531891783. ✓ (a, b) -> a.getTotal() - b.getTotal()

6406531891784. ✗ (a, b) -> b.getTotal() - a.getTotal()

6406531891785. ✓ (Student a, Student b) -> a.getTotal() - b.getTotal()

6406531891786. ✖ (a, b) -> { return b.getTotal() - a.getTotal();}

Question Number : 96 Question Id : 640653566058 Question Type : MSQ Is Question

Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 5 Selectable Option : 0

Question Label : Multiple Select Question

Consider the code given below.

```
public class ListExample {  
    public static void main(String[] args) {  
        ----- //LINE 1  
        ----- //LINE 2  
        list1 = new ArrayList<String>();  
        list1.add("Apples");  
        list1.add("Grapes");  
        list2 = new LinkedList<String>(list1);  
        for(String s : list1)  
            System.out.println(s);  
        for(String s : list2)  
            System.out.println(s);  
    }  
}
```

If the code given above produces the output:

Apples
Grapes
Apples
Grapes

What should be the correct choice for LINE 1 and LINE 2?

Options :

```
AbstractSequentialList<String> list1;  
AbstractSequentialList<String> list2;
```

6406531891795. ✖

6406531891796. ✓
AbstractList<String> list1;
AbstractSequentialList<String> list2;

6406531891797. ✓
List<String> list1;
List<String> list2;

6406531891798. ✗
AbstractList<String> list1;
SequentialList<String> list2;

Question Number : 97 Question Id : 640653566059 Question Type : MSQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0
Correct Marks : 5 Selectable Option : 0
Question Label : Multiple Select Question

Consider the Java code given below.

```
import java.util.*;
public class SetTest {
    public static void main(String[] args) {
        var list = new LinkedList<String>();
        list.add("Modern");
        list.add("Application");
        list.add("Development");
        list.add("Development");
        //CODE BLOCK
        for(String str:list) {
            set1.add(str);
            set2.add(str);
        }
        for(String str:set1)
            System.out.print(str+" ");
        System.out.println();
        for(String str:set2)
            System.out.print(str+" ");
    }
}
```

Choose the correct option(s) to be filled in place of CODE BLOCK so that the program always generates the following output:

```
Application Development Modern
Modern Application Development
```

Options :

6406531891799. ✖ `var set1 = new HashSet<String>();`
`var set2 = new HashSet<String>();`

6406531891800. ✔ `var set1 = new TreeSet<String>();`
`var set2 = new LinkedHashSet<String>();`

6406531891801. ✖ `var set1 = new TreeSet<String>();`
`var set2 = new TreeSet<String>();`

6406531891802. ✖ `var set1 = new LinkedHashSet<String>();`
`var set2 = new LinkedHashSet<String>();`

MLT

Section Id :	64065338402
Section Number :	5
Section type :	Online
Mandatory or Optional :	Mandatory
Number of Questions :	21
Number of Questions to be attempted :	21
Section Marks :	100
Display Number Panel :	Yes
Group All Questions :	No
Enable Mark as Answered Mark for Review and Clear Response :	Yes
Maximum Instruction Time :	0
Sub-Section Number :	1
Sub-Section Id :	64065380937
Question Shuffling Allowed :	No
Is Section Default? :	null

Question Number : 98 Question Id : 640653566069 Question Type : MCQ Is Question Mandatory : No Calculator : None Response Time : N.A Think Time : N.A Minimum Instruction Time : 0

Correct Marks : 0

Question Label : Multiple Choice Question

THIS IS QUESTION PAPER FOR THE SUBJECT "DIPLOMA LEVEL : MACHINE LEARNING TECHNIQUES (COMPUTER BASED EXAM)"

ARE YOU SURE YOU HAVE TO WRITE EXAM FOR THIS SUBJECT?

CROSS CHECK YOUR HALL TICKET TO CONFIRM THE SUBJECTS TO BE WRITTEN.