OCA PREP QUESTIONS
SET – 8

1. What will be the output of compiling and executing the Test class?

```
public class Test {
    public static void main(String[] args) {
        int a = 5;
        int x = 10;
        switch(x) {
            case 10:
                a *= 2;
            case 20:
                a *= 3;
            case 30:
                a *= 4;
        }
        System.out.println(a);
    }
}
```

A. 10
B. 120
C. 5
D. 30

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

```
public class Test {
    public static void main(String[] args) {
        StringBuilder sb = new StringBuilder("Good"); //Line 3
        change(sb); //Line 4
        System.out.println(sb); //Line 5
    }

    private static void change(StringBuilder s) {
        s.append("_Morning"); //Line 9
    }
}
```

A. _Morning
B. None
C. Good_Morning
D. Good

3. Below is the code of Test.java file:

```
import java.util.ArrayList;
import java.util.List;

abstract class Animal {}
```

```
class Dog extends Animal{}

public class Test {
     public static void main(String [] args) {
         List<Animal> list = new ArrayList<Dog>();
         list.add(0, new Dog());
         System.out.println(list.size() > 0);
     }
}
```

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

A. Runtime exception
B. true
C. false
D. Compilation error

4. Consider codes of 3 java files:

```
//Class1.java
package com.rock.oca;

import java.io.FileNotFoundException;

public class Class1 {
     public void read() throws FileNotFoundException {}
}
```

```
//Class2.java
package com.rock.oca;

public class Class2 {
     String Class2;
     public void Class2() {}
}
```

```
//Class3.java
package com.rock.oca;

public class Class3 {
     private void print() {
         private String msg = "HELLO";
         System.out.println(msg);
     }
}
```

Which of the following statement is true?

A. Class1.java and Class2.java compile successfully
B. Class2.java and Class3.java compile successfully
C. Class1.java and Class3.java compile successfully
D. Only Class1.java compiles successfully
E. Only Class2.java compiles successfully
F. Only Class3.java compiles successfully

5. Consider below code:

```
//Test.java
import java.time.LocalDate;
import java.time.format.DateTimeFormatter;

public class Test {
     public static void main(String [] args) {
         LocalDate date1 = LocalDate.parse("1947-08-15", DateTimeFormatter.ISO_DATE);
         LocalDate date2 = LocalDate.parse("1947-08-15",
                                                DateTimeFormatter.ISO_LOCAL_DATE);
         LocalDate date3 = LocalDate.of(1947, 8, 15);

         System.out.println(date1.equals(date2) + " : " + date2.equals(date3));
     }
}
```

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

A. true : true
B. false:true
C. Runtime exception
D. true:false
E. false:false

6. Consider below code:

```
//Test.java
import java.util.ArrayList;
import java.util.List;

public class Test {
     public static void main(String[] args) {
         List<String> dryFruits = new ArrayList<>();
         dryFruits.add("Walnut");
         dryFruits.add("Apricot");
         dryFruits.add("Almond");
         dryFruits.add("Date");

         for(String dryFruit : dryFruits) {
             if(dryFruit.startsWith("A")) {
                 dryFruits.remove(dryFruit);
             }
         }

         System.out.println(dryFruits);
     }
}
```

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

A. An exception is thrown at runtime
B. Compilation error
C. [Walnut,Date]
D. [Walnut,Apricot,Almond,Date]

7. Consider below code:

```java
public class Test {

    private static void add(int i, int j) {
        System.out.println("int version");
    }

    private static void add(Integer i, Integer j) {
        System.out.println("Integer version");
    }

    public static void main(String[] args) {
        add(10, 20);
    }

}
```

Which modifications, done independently, print "Integer version" on to the console?

Select 3 options.

A. Remove add(int i, int j) method declaration and definition
B. Replace add(10,20); by add(null,null);
C. Replace add(10,20); by add(10.0,20.0);
D. Replace add(10,20); by add(new Integer(10),new Integer(20));

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

```java
public class Test {
    public static void main(String[] args) {
        char [][] arr = {
                {'A', 'B', 'C'},
                {'D', 'E', 'F'},
                {'G', 'H', 'I'}
        };

        for(int i = 0; i < arr.length; i++) {
            for(int j = 0; j < arr[i].length; j++) {
                System.out.print(arr[i][1]);
            }
            System.out.println();
        }
    }
}
```

A. BBB
EEE
HHH

B. AAA
DDD
GGG

C. CCC
FFF
III

D. ABC
DEF
GHI

9. Consider codes below:

```java
//A.java
package com.rock.oca;

public class A {
     public void print() {
         System.out.println("A");
     }
}
```

```java
//B.java
package com.rock.oca;

public class B extends A {
     public void print() {
         System.out.println("B");
     }
}
```

```java
//C.java
package com.rock.oca;

public class C extends A {
     public void print() {
         System.out.println("C");
     }
}
```

```java
//Test.java
package com.rock.oca.test;

import com.rock.oca.*;

public class Test {
     public static void main(String[] args) {
         A obj1 = new C();
         A obj2 = new B();
         C obj3 = (C)obj1;
         C obj4 = (C)obj2;
         obj3.print();
     }
}
```

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

A. A

B. B

C. Compilation error

D. ClassCastException is thrown at runtime

E. C

10. Consider below code of Test.java file:

```java
class Document {
    int pages;
    Document(int pages) {
        this.pages = pages;
    }
}

class Word extends Document {
    String type;
    Word(String type) {
        super(20); //default pages
        /*INSERT-1*/
    }

    Word(int pages, String type) {
        /*INSERT-2*/
        super.pages = pages;
    }
}

public class Test {
    public static void main(String[] args) {
        Word obj = new Word(25, "TEXT");
        System.out.println(obj.type + "," + obj.pages);
    }
}
```

Currently above code causes compilation error.

Which of the options can successfully print TEXT,25 on to the console?

A. Replace /*INSERT-1*/ with:
super.type=type;
Replace /*INSERT-2*/ with:
this(type);

B. Replace /*INSERT-1*/ with:
this.type=type;
Replace /*INSERT-2*/ with:
this(type);

C. Replace /*INSERT-1*/ with:
   super.type=type;
   Replace /*INSERT-2*/ with:
   super(type);

D. Replace /*INSERT-1*/ with:
   this(type);
   Replace /*INSERT-2*/ with:
   this.type=type;