Java Software Development Exercise 6

1. Which of these are core interfaces in the collections framework? Select the three correct answers.
   (A) Set<E>
   (B) Bag<E>
   (C) LinkedList<E>
   (D) Collection<E>
   (E) Map<K,V>

2. Which of these implementations are provided by the `java.util` package? Select the two correct answers.
   (A) HashList<E>
   (B) HashMap<K,V>
   (C) ArraySet<E>
   (D) ArrayMap<K,V>
   (E) TreeMap<K,V>

3. Which sequence of digits will the following program print?
```java
import java.util.ArrayList;
import java.util.LinkedList;
import java.util.List;
public class Lists {
    public static void main(String[] args) {
        List<String> list = new ArrayList<String>();
        list.add("1");
        list.add("2");
        list.add(1, "3");
        List<String> list2 = new LinkedList<String>(list);
        list.addAll(list2);
        list2 = list.subList(2, 5);
        list2.clear();
        System.out.println(list);
    }
}
```
Select the one correct answer.
   (A) [1, 3, 2]
   (B) [1, 3, 3, 2]
   (C) [1, 3, 2, 1, 3, 2]
   (D) [3, 1, 2]
   (E) [3, 1, 1, 2]
   (F) None of the above.

4. Which statements are true about maps? Select the two correct answers.
   (A) The return type of the `values()` method is `Set`.
   (B) Changes made in the set view returned by `keySet()` will be reflected in the original map.
   (C) The `Map` interface extends the `Collection` interface.
   (D) All keys in a map are unique.
   (E) All `Map` implementations keep the keys sorted.

5. Which code, when inserted independently at (1), will result in the following output from the program: {1=Odd, 2=Even, 3=Odd}?
```java
import java.util.Map;
import java.util.TreeMap;
```

```
public class StringMap {
    public static void main(String[] args) {
        Map<Integer, String> myMap = new TreeMap<Integer, String>();
        for (Integer key : new int[] {1, 2, 1, 3, 1, 2, 3, 3}) {
            // (1) INSERT CODE HERE ...
        }
        System.out.println(myMap);
    }
    private static String toggle(String str) {
        if (str.equals("Odd"))
            str = str.replace("Odd", "Even");
        else
            str = str.replace("Even", "Odd");
        return str;
    }
}
```

Select the one correct answer.

(A) `String value = myMap.get(key);`
   `myMap.put(key, (value == null) ? "Odd" : StringMap.toggle(value));`

(B) `String value = myMap.get(key);`
   `if (value == null)`
   `  value = "Odd";`
   `else`
   `  StringMap.toggle(value);`
   `myMap.put(key, value);`

(C) `String value = myMap.get(key);`
   `if (!myMap.containsKey(key))`
   `  myMap.put(key, "Odd");`
   `else`
   `  StringMap.toggle(value);`

(D) All of the above.

Answer

1. (A), (D), and (E)
`Set`, `Collection`, and `Map` are core interfaces in the collections framework. `LinkedList` is a class that implements the `List` interface. There is no class or interface named Bag.

2. (B) and (E)
The `java.util` package provides map implementations named `HashMap` and `TreeMap`. It does not provide any implementations named `HashList`, `ArraySet`, and `ArrayMap`.

3. (A)
[1, 3, 2] is printed. First, "1" and "2" are appended to an empty list. Next, "3" is inserted between "1" and "2", and then the list is duplicated. The original list is concatenated with the copy. The sequence of elements in the list is now "1", "3", "2", "1", "3", "2". Then a sublist view allowing access to elements from index 2 to index 5 (exclusive) is created (i.e., the subsequence "2", "1", "3"). The sublist is cleared, thus removing the elements. This is reflected in the original list and the sequence of elements is now "1", "3", "2".

4. (B) and (D)
Although all the keys in a map must be unique, multiple identical values may exist. Since values are not unique, the `values()` method returns a `Collection` instance and not a `Set` instance. The collection objects returned by the `keySet()`, `entrySet()`, and `values()` methods are backed by the original `Map` object. This means that changes made in one are reflected in the other. Although implementations of `SortedMap` keep the entries sorted on the keys, this is not a requirement for classes that implement `Map`. For instance, the entries in a `HashMap` are not sorted.

5. (A)
Strings are immutable. In (B) and (C) the argument value in the call to the method `toggle()` refers to the old string after completion of the call, so the value in the map is not updated with the new string.