CORE_JAVA_DXC_2022_002

Employee Id: _____
Duration: 3.0 Hrs **Max. Marks 60**

Instructions:

1. First 15 minutes are provided to read & understand the test paper and 15 minutes for recheck whatever code we have written.
2. Next 30 minutes for Question Part1 and 2.0Hrs for Part2
3. Create a project with Collection followed by your first name and employee ID.
4. Take care of Plagiarism
5. Referring Java Documentation is allowed.
6. **Marks will only be awarded to working functionalities**

Question Part1

Q1. Multiple Choice Questions:
Q1.1 What is printed as a result ofexecuting the following code segment?

```
List<Integer> 1st = new ArrayList<Integer>();
for (int k 1; k <= 6: k++)
        Ist.add(new Integer(k)):
for (int k 0; k < 3; k++){
        Integer i Ist.remove(k);
        Ist.add(i);
}
for (Integer i : 1st)
        System.out.print(i);
```

(A)123456 (B)456123 (C)456321 (D)246135
(E)IndexOutOfBoundsException

Q1.2 Classes Salsa and Swing implement an interface Dance.

```
        perform(new Salsa());
        perform(new Swing());
```

If both call are valid, which of the following headers of the perform method(s) in a class Dancer will compile successfully?
1. Two methods:

```
        public void perform(Salsa dance)
        public void perform (Swing dance)
```

II. public void perform (Dance dance)
III. public void perform(Object dance)
(A)I only (B)II on1y (C)I and II only (D) II and III only
(E) I, II, and III

Q1.3 Consider the following method.

```
/** Rearranges the elements in words according to
 *  the values stored in an integer array indices,
 *  so that the element of words at index indices[k] is moved to the element at index k.
 *  * Precondition: words.size() indices.length
 */  public void permute(List<String> words, int[] indices) {
ArrayList<String> temp = new ArrayList<String>();
< missing code >
```

For example, after executing the code segment

```
List<String> words = new ArrayList<String>();
words.add("I"):
words.add("am");
words.add("Sam");
int[] indices {2, 0, 1);
permute (words, indices);
```

words will become the list ["S am ", II I", " am"]. Which ofthe following code segments could replace < missing code> in the permute method?

```
for (String word : words)
1. temp.add(word);
for (int k 0; k < indices.length; k++) words.set(k, temp.get(indices[k]»;
II. for (int j : indices) temp.add(words.get(j));
for (int k = 0; k < indices.length; k++) words.set(k, temp.get(k));
III. while (words. size () > 0) temp.add(words.remove(O));
for (int j : indices) words.add(temp.get(j));
```

(A)I only (B) II only (C) I and II only (D) II and III only
(E) I, II, and III

Q1.4 What is printed as a result ofexecuting the following code segment?

```
ArrayList<String> digits new ArrayList<String>();
for (int k = 0; k <= 9; k++)
digits.add("" + k);
for (int k = 0; k <= 4; k++){
        String d1 = digits.remove(k);
        String d2 = digits.remove(k);
        digits.add(k, d1 + "+" + d2);
}
System.out.println(digits);
```

(A)[0+1, 1+2, 2+3, 3+4, 4+5] (B)[0+1, 2+3, 4+5, 6+7, 8+9]
(C)[0+1, 1+2, 2+3, 3+4, 5, 6, 7, 8, 9] (D)[0+1, 1+2, 2+3, 3+4, 4+5, 6, 7, 8, 9]
(E)[0+0, 1+1, 2+2, 3+3, 4+4, 5, 6, 7, 8, 9]

Q1.5 Consider the following interface TV and class MyTV.

```
public interface TV
{
void tuneTo(String channel);
}
public class MyTV implements TV {
 private ArrayList<String> myFavoriteChannels;
public MyTV(ArrayList<String> channels) {
/* implementation not shown */ }
public void tuneTo(int k) {
/* implementation not shown */
public void tuneTo(int k, String name) {
/* implementation not shown */ }
}
```

One ofthem has one or more errors and won't compile properly. Which ofthe following best describes the compiler errors reported for the code that is shown?
(A)In the TV interface, the tuneTo method header is missing the keyword public
(B)MyTV should be declared abstract; it does not define tuneTo (String)
(C)tuneTo is defined more than once in MyTV
(D)Cannot convert int to String in the tuneTo method in MyTV
(E)Two errors: (1) tuneTo is defined more than once and (2) cannot convert int to String in the tuneTo (int) method in MyTV

Q1.6 & Q1.7refer to the following class Game and the incomplete class ChessGame.

```
public class Game
{
private String gameName:
private List<String> players:
public Game(String name)
{
gameName =name;
players = new ArrayList<String>();
public Game(String name, String[] people)
{
gameName= name;
players= new ArrayList<String>();
for (String nm : people)
players.add(nm);
public void addPlayer(String name) { players.add(name); }
public String getPlayer(int k)
{
return players.get(k-l);
}
public toString ()
{
return gameName + " game" + players.toString()
}
}
public class ChessGame extends Game
{
public ChessGame(String white, String black)
{
< missing code>
}
}
```

Consider the following code segment in a `Game`'s client class.

```
String[] players = {"Annette", "Bertrand",
                            "Claude", "Danielle"};
Game game = new Game("Bauernschnapsen", players);
System.out.println( < missing expression >);
```

Which of the following can replace < *missing expression*> so that the code results in printing `"Annette"`?

(A) `game.getPlayer(0)`
(B) `game.getPlayer(1)`
(C) `game.players.get(0)`
(D) `game.players.get(1)`
(E) `game.getPlayers().get(0)`

Which of the following can replace < *missing code* > in `ChessGame`'s constructor so that the statement

```
System.out.println(new ChessGame("Deep Blue",
                                          "Kasparov"));
```

prints

```
Chess game [Deep Blue, Kasparov]
```

I.
```
super("Chess", white, black);
```

II.
```
super("Chess");
super.addPlayer(white);
super.addPlayer(black);
```

III.
```
String[] players = {black, white};
super("Chess", players);
```

(A) I only
(B) II only
(C) I and II only
(D) II and III only
(E) I, II, and III

Question Part2

1. Write a program to show CRUD Operation on student data using ArrayList with menu in main.

2. Create a Product class with Product Id & Product Name. Write a program to accept information of 10 products and store that in HashTable. Search a particular product in the Hash Table. Remove a particular product id and product name from the Hash Table.
   The product list is as follows:

Product Id	Product Name
P001	Maruti800
P002	MarutiZen
P003	MarutiEsteem

3. Implement vector class for this problem
   1. Create an Employee class which will have details like EmployeeNo, EmployeeName and Address. You should pass value for EmployeeNo, EmployeeName and Address through constructor.
   2. Create a method addInput( ) which will add employee details to vector.
   3. Create method display( ) which should display all data from vector using Iterator.

4. Write a program having user interface like
   1. accept first name and surname
   2. display total name
   3. exit

   Option1: should accept First Name and SurName from command prompt and save that to Vector object
   Option2: it has to display how many names entered in the vector object
   This menu should be repeated untill users selects exit.
   To store first name and surname, create a class Name with these two attributes.