

Java Coding Challenge

Candidate name :

Date :

# Coding Test guidelines

We invite applicant to relax and enjoy solving day to day working experience based java problems and share information about roles and responsibilities. It is highly encouraged to use Java 8 version or best known methods to solve these Challenges.

## 1 Fibonacci Series

Given method call is : fibonacci(0, 1, 10, 1)

Print first 10 numbers in fibonacci series : 0, 1, 1, 2, 3, 5, 8, 13, 21, 34

**/\*\***

**\* Replace line 8 with one line of code.**

**\* Hint : Recursion in java is a process in which a method calls itself continuously**

**\*\*/**

1. **public static void fibonacci(int num1, int num2, int limit, int count) {**
2. **int total = num1 + num2;**
3. **if (count <= limit) {**
4. **System.out.print(num1 + " ");**
5. **num1 = num2;**
6. **num2 = total;**
7. **count = count + 1;**
8. **//TEST1 : Write one line of code to complete fibonacci series - Hint : Use recursion**
9. **} else {**
10. **return;**
11. **}**
12. **}**

## 2 Correct the following program

* What exception do you expect in the following scenario?
* Best collection to use in Hashmaps to avoid this Exception.
* Correct the program by replacing lines 7-13 to remove “number” from given list of “Integers”. Given input=3, explain your approach to fix the code.

1. public static List<Integer> removeValue(int number) {
2. List<Integer> list = new ArrayList<>();
3. list.add(1);
4. list.add(2);
5. list.add(3);
6. Iterator<Integer> it = list.iterator();
7. while (it.hasNext()) {
8. Integer value = it.next();
9. System.out.println("List Value:" + value);
10. if (value.equals(number)) {
11. list.remove(value);
12. }
13. }
14. return list;
15. }

## 3 Evaluate output of the following program and set daemon thread

* What are thread priorities? What is max thread priority?
* Different ways to write a thread.
* What is a daemon thread?
* Correct the following program and make t1 a daemon thread.

1. public class Test1 extends Thread{
2. public static void main(String[] args) {
3. Test1 t1=new Test1();
4. t1.setPriority(12**);**
5. //Make this a daemon thread
6. }
7. }

## 4 Sort and reverse a list of strings

* Complete line 2,3 to sort and reverse list of countries. Hint : You may use collections API.

/\*\*

\* Sort and reverse a List of countries.

\*/

1. **public static void sortAndReverseList(List<String> countriesList) {**
2. //Sort list
3. //Reverse list
4. }

## 5 Evaluate output of following program

* Define Set
* Predict outputs in line 8, 10.

1. **public static void testSet() {**
2. List<String> fruits = new ArrayList<String>();
3. fruits.add("Apple");
4. fruits.add("Mango");
5. fruits.add("Guava");
6. fruits.add("Mango");
7. Set<String> fruitSet = new HashSet<>(fruits);
8. **System.*out.println(fruitSet); //Predict Output :***
9. fruitSet = new TreeSet<>(fruits);
10. **System.*out.println(fruitSet);//Predict Output :***
11. **}**

## 6 Design patterns

* List some of java design patterns you may have used.
* Define singleton pattern and explain highlighted lines of code.

1. class Singleton
2. {
3. private volatile static Singleton obj;
4. private Singleton() {}
5. public static Singleton getInstance()
6. {
7. if (obj == null) {
8. synchronized (Singleton.class) {
9. if (obj==null) **{**
10. obj = new Singleton();
11. }
12. }
13. return obj;
14. }
15. }

## 7 JAVA 8

* List some new features of Java 8.
* Can you write an example lambda expression to stream any collection.

## 8 Explain comparator and usage with JAVA 8

* Explain comparator
* Complete following program to sort list of users by username first and then user ID given User class has fields username, userid.

1 public void sortUsers() {

2 User user1 = new User("Ben", 1);

3 User user2 = new User("Anna", 2);

4 User user3 = new User("Madonna", 3);

5 User user4 = new User("Siri", 4);

6 List<User> users = new ArrayList<>();

7 users.add(user1);

8 users.add(user2);

9 users.add(user3);

10 users.add(user4);

11 // Complete code to compare by first name and then last name

13 System.out.println("Sorted users list " + sortedList);

14 }

## 9 Spring MVC and Spring Microservices features

* List some features of Spring MVC and microservices.
* Do you have any experience with Spring microservices development?
* In terms of deployment and release, how are microservices better than deploying a

Spring mvc project run on servers like tomcat in a production environment?

## 10 Skills

Please explain following skills relevance to your work and a very brief explanation about list of topics.

* Git code repository and Gerrit code review.
* Maven build tool.
* ORM experience : Hibernate, Spring JDBC template.
* Cloud computing experience.
* PL/SQL coding and sql query writing experience.
* Unix/Linux skills and experience.
* Applicant is flexible to learn and adapt to a dynamic work environment.
* Best practices in coding :
  + Naming convention of methods, variables to have camel casing, package naming, coding comments.
  + Designing to interface or abstract design pattern application in code.
  + Peer review of code and Code reviewing experience.
  + Test case writing and code quality tools like sonarqube.