Date/Time API (JSR 310)

1. Given the following code snippet, what will be the output?

```
LocalDateTime dateTime = LocalDateTime.of(2024, 7, 30, 14, 30);
LocalDateTime newDateTime = dateTime.plusHours(3).minusMinutes(15);
System.out.println(newDateTime);
```

a) `2024-07-30T17:15`
b) `2024-07-30T17:45`
c) `2024-07-30T15:15`
d) `2024-07-30T14:45`

2. What is the result of the following code snippet?

```
ZonedDateTime zonedDateTime =
ZonedDateTime.now(ZoneId.of("America/New_York"));
ZonedDateTime utcDateTime =
zonedDateTime.withZoneSameInstant(ZoneId.of("UTC"));
System.out.println(zonedDateTime);
System.out.println(utcDateTime);
```

a) `Same local date and time for both, different time zone`
b) `Same date and time, same time zone`
c) `Different date and time, same time zone`
d) `Different date and time for different time zones`

Generic Classes

3. Consider the following code snippet. What will be the output?

```
List<? extends Number> list = Arrays.asList(1, 2.5, 3L);
System.out.println(list.get(1));
```

a) `2.5`
b) `2`
c) `Number`
d) `ClassCastException`

4. What will happen if you try to add an element to a `List<? extends Number>`?

a) The operation will succeed with no issues.
b) It will compile but throw a runtime exception.
c) It will not compile.
d) The added element will be of type `Object`.

Collections Framework

5. What is the result of the following code snippet?

```
Set<Integer> set = new HashSet<>(Arrays.asList(1, 2, 3));
set.add(3);
System.out.println(set.size());
```

a) 3
b) 2
c) 4
d) 1

6. What will be the result of the following code snippet?

```
Map<String, Integer> map = new HashMap<>();
    map.put("a", 1);
    map.put("b", 2);
    map.put("a", 3);
    System.out.println(map.size());
```

a) 3
b) 2
c) 1
d) 4

Working with Stacks

7. What is the output of the following code snippet?

```
Stack<String> stack = new Stack<>();
    stack.push("A");
    stack.push("B");
    stack.push("C");
    stack.pop();
    stack.push("D");
  System.out.println(stack.peek());
```

a) B
b) C
c) D
d) A

8. How does the stack class in handle concurrent access?

a) It is inherently thread-safe.
b) It uses `synchronized` methods but is not fully thread-safe.
c) It uses explicit locking mechanisms.
d) It does not handle concurrent access.

Working with Queues

9. What is the result of the following code snippet?

```java
Queue<Integer> queue = new PriorityQueue<>(Arrays.asList(3, 1, 4, 2));
queue.add(5);
System.out.println(queue.poll());
```

a) 1
b) 2
c) 3
d) 5

10. What is the time complexity of `offer` operation in a `PriorityQueue`?

a) O(1)
b) O(log n)
c) O(n)
d) O(n log n)

Sorting & Searching Algorithms

11. What is the average-case time complexity of Merge Sort?

a) O(n)
b) O(n log n)
c) O(n^2)
d) O(log n)

12. Given the following code snippet, what will be the result?

```java
int[] array = {10, 7, 8, 9, 1, 5};
    quickSort(array, 0, array.length - 1);
    System.out.println(Arrays.toString(array));

    void quickSort(int[] arr, int low, int high) {
        if (low < high) {
            int pi = partition(arr, low, high);
            quickSort(arr, low, pi - 1);
            quickSort(arr, pi + 1, high);
        }
    }
```

```java
        int partition(int[] arr, int low, int high) {
            int pivot = arr[high];
            int i = (low - 1);
            for (int j = low; j < high; j++) {
                if (arr[j] <= pivot) {
                    i++;
                    int temp = arr[i];
                    arr[i] = arr[j];
                    arr[j] = temp;
                }
            }
            int temp = arr[i + 1];
            arr[i + 1] = arr[high];
            arr[high] = temp;
            return i + 1;

        }
```

a) [1, 5, 7, 8, 9, 10]
b) [10, 7, 8, 9, 1, 5]
c) [1, 5, 7, 8, 9, 10]
d) [1, 5, 7, 8, 10, 9]

Input & Output Streams

13. What will be the result of the following code snippet?

```java
try (FileOutputStream fos = new FileOutputStream("testfile.txt")) {
            fos.write("Hello".getBytes());
        }
        FileInputStream fis = new
FileInputStream("testfile.txt");
        byte[] data = fis.readAllBytes();
        System.out.println(new String(data));
```

a) Hello
b) null
c) Hello with extra bytes
d) Runtime Exception

14. What will be the output if you read from a `BufferedReader` using `readLine()` method?

```java
BufferedReader reader = new BufferedReader(new FileReader("testfile.txt"));
String line = reader.readLine();
System.out.println(line);
```

a) Reads and prints the entire content of the file
b) Reads the first line of the file
c) Reads and prints the file content line by line
d) Returns `null` if file is empty

Multi-Threading

15. What will be the output of the following code snippet?

```java
Runnable task1 = () -> {
    for (int i = 0; i < 5; i++) {
        System.out.print("1");
    }
};
Runnable task2 = () -> {
    for (int i = 0; i < 5; i++) {
        System.out.print("2");
    }
};
Thread t1 = new Thread(task1);
Thread t2 = new Thread(task2);
t1.start();
t2.start();
t1.join();
t2.join();
```

a) `1111122222`
b) `2222111111`
c) `1212121212`
d) Output may vary

16. Which of the following is true about thread synchronization?

a) `synchronized` blocks ensure that only one thread can execute a block of code at a time.
b) `synchronized` ensures that threads execute in parallel without locking.
c) Synchronization increases the performance of multi-threaded applications.
d) `synchronized` blocks work on a thread-local basis.

JDBC API

17. What is the result of executing the following SQL query?

```java
Statement stmt = connection.createStatement();
stmt.executeUpdate("INSERT INTO students (id, name) VALUES (1,
'Monika')");
ResultSet rs = stmt.executeQuery("SELECT * FROM students");
```

a) A new row is inserted, and `ResultSet` will contain the new row.
b) `ResultSet` will be empty since `executeQuery` is used after an `INSERT` statement.
c) `executeUpdate` and `executeQuery` cannot be used together.
d) The `INSERT` statement will throw an exception.

18. What is the purpose of `DatabaseMetaData` in JDBC?

a) To obtain information about database metadata and capabilities
b) To modify the database schema
c) To handle database transactions
d) To execute SQL queries

Performing Unit Testing using JUnit4

19. What is the purpose of the `@Ignore` annotation in JUnit4?

a) To skip a test method during execution
b) To run a test method multiple times
c) To ensure that a test method fails
d) To mark a test method as deprecated

20. What will be the output of the following code snippet?

```
@Test(timeout = 1000)
public void testMethod() throws InterruptedException
{
    Thread.sleep(2000);
}
```

a) Test will pass because of the timeout
b) Test will fail because it takes too long
c) Test will not compile
d) Test will be skipped

21. What is the output of the following code snippet using JUnit assertions?

```
assertTrue(5 > 3);
assertFalse(3 > 5);
assertEquals("abc", "abc");
assertNotNull(new Object());
```

a) All assertions will pass
b) All assertions will fail
c) `assertTrue` and `assertFalse` will pass, others will fail
d) `assertEquals` will fail

Advanced and Tricky Questions

22. What will be the result of the following code snippet when executed?

```
List<String> list = Arrays.asList("a", "b", "c");
list.set(1, "d");
System.out.println(list);
```

a) `[a, d, c]`
b) `[a, b, c]`
c) `UnsupportedOperationException`
d) `IndexOutOfBoundsException`

23. What happens if you try to use the `HashSet` class with a custom class that does not override `equals` and `hashCode` methods?

a) The `HashSet` will behave as expected.
b) The `HashSet` will throw a `NullPointerException`.
c) The `HashSet` will not maintain unique elements properly.
d) The `HashSet` will not compile.

24. What is the result of the following code snippet?

```
String str = new String("test");
String str2 = "test";
System.out.println(str == str2);
```

a) `true`
b) `false`
c) `Compilation Error`
d) `Runtime Error`

25. Consider the following code snippet. What is the output?

```
StringBuilder sb = new StringBuilder("");
    sb.append(" Programming");
    sb.insert(4, " Language");
    System.out.println(sb);
```

a) `Language Programming`
b) `LanguageProgramming`
c) `Programming Language`
d) `Programming Language`