Q1: Answer the following [3 marks]

(a) 3 short questions of 1 mark each [3 marks]

i) Describe difference of LocalTime and LocalDateTime in Java? [1 mark]
Answer: LocalTime represents only the time of day (e.g., 14:30:00) without any date or time zone information. LocalDateTime, on the other hand, represents both a date and a time (e.g., 2025-04-02T14:30:00) without a time zone. Essentially, LocalDateTime combines LocalDate (date) and LocalTime (time) into a single object.

ii) Implement how to convert a long to a String in Java? [1 mark]
Answer:

long number = 123456789L;

String str = String.valueOf(number);

System.out.println(str); // Output: 123456789

iii) What is the purpose of the container class in Bootstrap? [1 mark]
Answer: The container class in Bootstrap is used to create a responsive, fixed-width container that centers the content on the page with padding on the sides. It ensures that the content is not stretched to the full width of the viewport and provides a consistent layout across different screen sizes. For example:

<div class="container">

  <!-- Content here -->

</div>

(b) Objective type/MCQs/True-False/Fill in blanks (7 questions of 1 mark each) [7 marks]

1) What is the default format of LocalDateTime when converted to a string?

Answer: b) YYYY-MM-DD HH:mm:ss

2) True or False: The clearfix class in Bootstrap helps to clear floated elements.
Answer: True

3) Which method is used to convert a string to a Date?

Answer: b) SimpleDateFormat.parse(String s)

4) Fill in the blanks: The Bootstrap ___ class is used to create buttons, and ___ is used to add alerts.
Answer:
The Bootstrap <mark>btn</mark> class is used to create buttons, and <mark>alert</mark> is used to add alerts.

5) In JSON, what is used to separate key-value pairs?

Answer: a) : (colon)

6) What Bootstrap utility class makes an element invisible but still takes up space?

Answer: a) invisible

7) Which tool in the Selenium suite is used for writing automated test scripts?

Answer: c) Selenium WebDriver

Q2: Answer the following (2 or 3 mark questions) [4 marks]

(a) Two Questions of 2 Marks [4 marks]

1) What is the difference between OffsetTime and OffsetDateTime in Java?
Answer:

- **OffsetTime** represents a time with an offset from UTC/Greenwich (e.g., 14:30:00+02:00). It only includes the time and the offset, without a date.

- **OffsetDateTime** represents a full date and time with an offset from UTC/Greenwich (e.g., 2025-04-02T14:30:00+02:00). It includes both the date and time, along with the offset.
  In summary, OffsetDateTime is more comprehensive as it includes the date, while OffsetTime is limited to time and offset.

2) Describe what is JSON, and how is it different from XML?
Answer:

- **JSON (JavaScript Object Notation)** is a lightweight data-interchange format that is easy for humans to read and write, and for machines to parse and generate. It uses key-value pairs and supports data structures like objects and arrays (e.g., {"name": "John", "age": 25}).

- **XML (Extensible Markup Language)** is a markup language that defines a set of rules for encoding documents in a format that is both human-readable and machine-readable. It uses tags to define elements (e.g., <person><name>John</name><age>25</age></person>).
  Differences:
  - JSON is more concise and less verbose than XML, which requires opening and closing tags.
  - JSON is primarily used for data exchange (e.g., in APIs), while XML is often used for document storage and configuration.
  - JSON supports arrays natively, while XML does not.
  - JSON parsing is generally faster than XML parsing.

Q3: Attempt any TWO [10 marks]

(i) Demonstrate a Java program to convert a binary string to a decimal value.

Answer:

```java
public class BinaryToDecimal {

  public static void main(String[] args) {

    String binary = "1010"; // Example binary string

    int decimal = convertBinaryToDecimal(binary);

    System.out.println("Binary: " + binary + " -> Decimal: " + decimal);

  }


  public static int convertBinaryToDecimal(String binary) {

    int decimal = 0;

    int power = 0;

    // Iterate over the binary string from right to left

    for (int i = binary.length() - 1; i >= 0; i--) {

      if (binary.charAt(i) == '1') {
```

```
        decimal += Math.pow(2, power); // Add 2^power if the bit is 1

    }

        power++; // Increment the power for the next bit

    }

    return decimal;

  }

}
```

Output

Binary: 1010 -> Decimal: 10

(ii) Implement buttons using Bootstrap, and where is it typically used? [5 marks]

Answer:

```html
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Bootstrap Buttons</title>

    <!-- Bootstrap CSS -->

    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css" rel="stylesheet">

</head>

<body>

    <div class="container mt-3">

        <h2>Bootstrap Buttons Example</h2>

        <button class="btn btn-primary">Primary Button</button>

        <button class="btn btn-secondary">Secondary Button</button>

        <button class="btn btn-success">Success Button</button>
```
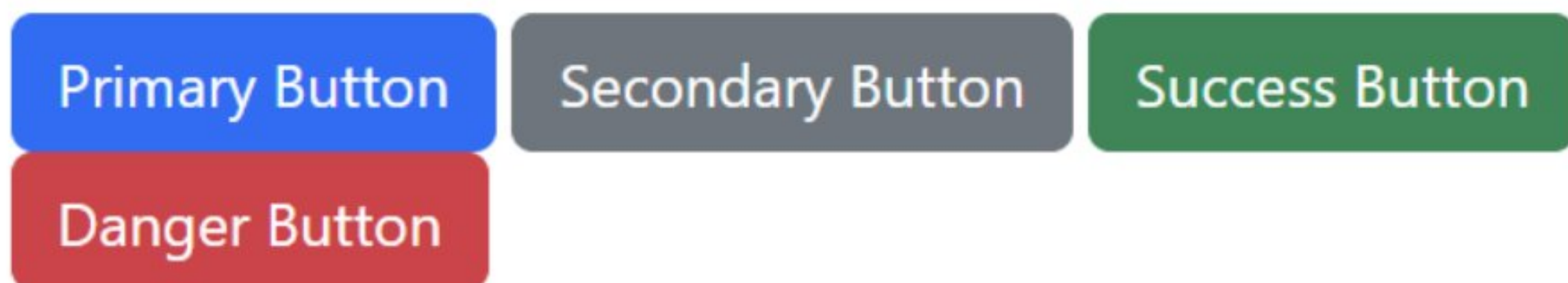
```
    <button class="btn btn-danger">Danger Button</button>

  </div>

  <!-- Bootstrap JS (optional for interactivity) -->

  <script
src="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/js/bootstrap.bundle.min.js"></s
cript>

</body>

</html>
```

Output

Bootstrap Buttons Example

| Primary Button | Secondary Button | Success Button |

| Danger Button |

Q4: Construct the following [10 marks]

(a) Create a Java method that takes a LocalDateTime object and returns the number of days left until the end of the year.

Answer:

```java
import java.time.LocalDateTime;

import java.time.temporal.ChronoUnit;


public class DaysUntilEndOfYear {

  public static long daysUntilEndOfYear(LocalDateTime dateTime) {

    // Get the current year from the LocalDateTime

    int year = dateTime.getYear();

    // Create a LocalDateTime for the end of the year (Dec 31, 23:59:59)
```

```java
        LocalDateTime endOfYear = LocalDateTime.of(year, 12, 31, 23, 59, 59);

        // Calculate the difference in days

        return ChronoUnit.DAYS.between(dateTime, endOfYear);

    }


    public static void main(String[] args) {

        // Example: Current date and time

        LocalDateTime now = LocalDateTime.of(2025, 4, 2, 14, 30); // April 2, 2025

        long daysLeft = daysUntilEndOfYear(now);

        System.out.println("Days left until the end of the year: " + daysLeft);

    }

}
```

Output

Days left until the end of the year: 273

(b)Construct a star rating system using Bootstrap utilities. What classes would be helpful for this? [5 marks]

Answer:

```html
<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Star Rating System</title>

    <!-- Bootstrap CSS -->

    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.0/dist/css/bootstrap.min.css"
rel="stylesheet">

    <!-- Font Awesome for star icons -->
```

```html
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/5.15.4/css/all.min.css">
    <style>
        .star-rating .fas { color: gold; }
        .star-rating .far { color: gray; }
    </style>
</head>
<body>
    <div class="container mt-3">
        <h2>Star Rating System</h2>
        <div class="star-rating">
            <i class="fas fa-star"></i> <!-- Filled star -->
            <i class="fas fa-star"></i> <!-- Filled star -->
            <i class="fas fa-star"></i> <!-- Filled star -->
            <i class="far fa-star"></i> <!-- Empty star -->
            <i class="far fa-star"></i> <!-- Empty star -->
        </div>
    </div>
</body>
</html>
```

output

Star Rating System
★★★☆☆