System Requirements Specification Index

For

Date and Time Formatting

Version 1.0

IIHT Pvt. Ltd.
fullstack@iiht.com

TABLE OF CONTENTS

<h1 style="text-align: center; color: red;">USE CASE DESCRIPTION</h1>
<h2 style="text-align: center;">System Requirements Specification</h2>

1 PROJECT ABSTRACT

You are tasked with formatting and parsing date and time values using DateTimeFormatter and working with different formats.

2 ASSESSMENT TASKS

Task 1: **Format Current Date and Time:**
- Use `LocalDateTime.now()` to get the current date and time and store it in a variable named `now`.
- Define a date-time format using: `DateTimeFormatter.ofPattern("dd/MM/yyyy HH:mm:ss")` and store the formatted date in a variable named `dateFormatter`.
- Format the current date and time using `.format(dateFormatter)`.
- **Print** the result: `"Formatted Date and Time:"` followed by the formatted date and time.

Task 2: **Parse Date Strings:**
- Declare and initialize a string variable `dateStr` with the value `"2025-01-02 12:30 PM"`.
- Define a date-time format using: `DateTimeFormatter.ofPattern("yyyy-MM-dd hh:mm a")` and store it in a variable named `formatter`.
- Parse `dateStr` into a `LocalDateTime` object using: `LocalDateTime.parse(dateStr, formatter)` and store the result in a variable named `dateTime`.
- **Print** the result: `"Parsed Date and Time:"` followed by the parsed date and time.

Task 3: **Format Date and Time with Time Zones:**
- Get the current date and time in the `"Asia/Tokyo"` time zone using `ZonedDateTime.now(ZoneId.of("Asia/Tokyo"))` and store it in a variable named `nowInTokyo`.
- Define a time zone format using: `DateTimeFormatter.ofPattern("yyyy-MM-dd HH:mm:ss Z")` and store it in a variable named `timeZoneFormatter`.
- Format the Tokyo time zone date-time using: `.format(timeZoneFormatter)` and store the formatted result in a variable named `formattedDateInTokyo` of

`String` datatype.
- **Print** the result: `"Formatted Date and Time in Tokyo:"` followed by the formatted date and time.

Expected Output:

Formatted Date and Time: 05/02/2025 12:48:08
Parsed Date and Time: 2025-01-02T12:30
Formatted Date and Time in Tokyo: 2025-02-05 21:48:08 +0900

(**Note**: The actual output values will vary depending on the current date and time of execution.)

3   TEMPLATE CODE STRUCTURE

3.1 PACKAGE: COM.YAKSHA.ASSIGNMENT.DATETIMEFORMATTINGANDPARSINGASSIGNMENT Resources

Class/Interface	Description	Status
DateTimeFormattingAnd ParsingAssignment (class)	● Main class demonstrating date and time formatting and parsing using `java.time.format.Date TimeFormatter`. Includes examples of formatting dates, parsing date strings, and formatting dates for specific time zones.	Need to be implemented.

4   EXECUTION STEPS TO FOLLOW

1. **All actions like build, compile, running application, running test cases will be through Command Terminal.**
2. **To open the command terminal the test takers, need to go to Application menu (Three horizontal lines at left top) Terminal New Terminal.**
3. **This editor Auto Saves the code.**

4. These are time bound assessments the timer would stop if you logout and while logging in back using the same credentials the timer would resume from the same time it was stopped from the previous logout.

5. To run your project use command:
   sudo JAVA_HOME=$JAVA_HOME /usr/share/maven/bin/mvn compile exec:java
   -Dexec.mainClass="com.yaksha.assignment.DateTimeFormattingAndParsingAssignment"

   *If it asks for the password, provide password : pass@word1

6. To test your project test cases, use the command
   sudo JAVA_HOME=$JAVA_HOME /usr/share/maven/bin/mvn test

   *If it asks for the password, provide password : pass@word1