

## Day 26

**Assignment 7: Timezone** Create a timezone converter that takes a time in one timezone and converts it to another timezone in java code and explanation.

**A)**

*Sure, here's a Java program that converts a time from one timezone to another timezone:*

*Java code:*

```
import java.time.LocalDateTime;

import java.time.ZoneId;

import java.time.ZonedDateTime;

import java.time.format.DateTimeFormatter;

import java.util.Scanner;


public class TimezoneConverter {

    public static void main(String[] args) {

        Scanner scanner = new Scanner(System.in);


        // Prompt the user to input the time and timezone

        System.out.print("Enter the time (HH:mm:ss): ");

        String inputTime = scanner.nextLine();


        System.out.print("Enter the timezone of the input time: ");

        String inputTimezone = scanner.nextLine();


        System.out.print("Enter the target timezone: ");

        String targetTimezone = scanner.nextLine();
```

```

// Parse the input time string into a LocalDateTime object

LocalDateTime localDateTime = LocalDateTime.parse(inputTime,
DateTimeFormatter.ofPattern("HH:mm:ss"));

// Convert LocalDateTime to ZonedDateTime with the input timezone

ZonedDateTime inputZonedDateTime = ZonedDateTime.of(localDateTime,
ZoneId.of(inputTimezone));

// Convert ZonedDateTime to ZonedDateTime with the target timezone

ZonedDateTime targetZonedDateTime =
inputZonedDateTime.withZoneSameInstant(ZoneId.of(targetTimezone));

// Format the result

String resultTime = targetZonedDateTime.format(DateTimeFormatter.ofPattern("HH:mm:ss"));

// Display the converted time

System.out.println("Time in " + targetTimezone + ": " + resultTime);

scanner.close();
}
}

```

### **Explanation:**

We import necessary classes from `java.time` package for working with dates, times, and timezones, and `java.util.Scanner` for user input.

The user is prompted to input the time in the format "HH:mm

*", the timezone of the input time, and the target timezone.*

*We parse the input time string into a `LocalDateTime` object using the `DateTimeFormatter`.*

*We convert the `LocalDateTime` to a `ZonedDateTime` object with the input timezone using `ZonedDateTime.of(inputTimezone)`.*

*We then convert the `ZonedDateTime` object to another `ZonedDateTime` object with the target timezone using `withZoneSameInstant(ZonedDateTime.of(targetTimezone))`.*

*Finally, we format the result and display the converted time in the target timezone.*