Java Basics 5 Assignment 1

Michael Moreland

October 29, 2021

1. Which class would you use to store your birthday in years, months, days, seconds, and nanoseconds?
You can use the LocalDate (years, months, and days) and LocalTime (seconds and nanoseconds) classes.

2. Given a random date, how would you find the date of the previous Thursday?

1. Save the random date as a LocalDate object

```
LocalDate randomDateObject = LocalDate.of(1998, Month.OCTOBER, 31);
```

2. On that object, call the "getDayOfWeek()" method

```
DayOfWeek randomDay = randomDateObject.getDayOfWeek();
```

   This method returns an object of class DayOfWeek.

   Alternatively, you can get the day of the week by passing the date as a parameter in the ".from()" DayOfWeek method

```
DayOfWeek randomDay = DayOfWeek.from(randomDateObject);
```

3. Create an integer variable that contains the difference in days between the given random date's day and the target previous Thursday

```
int differenceInDays = randomDay.compareTo(DayOfWeek.THURSDAY);
```

   Each day of the week is associated with an integer: Monday with 1, Tuesday with 2, Wednesday with 3, Thursday with 4... up to Sunday with 7. This lets us know that:

   (a) A positive integer is returned if the random date is a Friday (+1), Saturday (+2), or Sunday (+3) with respect to Thursday.
   (b) A zero means there is zero difference in days, i.e. that random date is a Thursday.
   (c) A negative integer is returned if the random date is a Monday (-3), Tuesday (-2), and Wednesday (-1).

4. Instantiate a LocalDate object to be Thursday's date and call the ".minusDays()" method on the random date object

```
LocalDate thursdayDate;

if (differenceInDays >= 0) {
    thursdayDate = randomDateObject.minusDays(differenceInDays);
}
else {
    differenceInDays += 7;
    thursdayDate = randomDateObject.minusDays(differenceInDays);
}
```

If the difference in days was ≥ 0, then subtracting that difference from the random date gets the previous Thursday.
Else, if the difference in days was < 0, then we first have to ***add*** 7 days to the difference in days and ***then*** finally subtract that difference from the random date. Without this added step, we get the next Thursday from the random date, not the previous Thursday.

5. Finally, that results with

```
7  public class dateprac {
8      public static void main(String[] args) {
9
10         LocalDate randomDateObject = LocalDate.of(2021, Month.OCTOBER,29);
11         DayOfWeek randomDay = DayOfWeek.from(randomDateObject);
12
13         int differenceInDays = randomDay.compareTo(DayOfWeek.THURSDAY);
14
15         LocalDate thursdayDate;
16
17         if (differenceInDays >= 0) {
18             thursdayDate = randomDateObject.minusDays(differenceInDays);
19         }
20         else {
21             differenceInDays += 7;
22             thursdayDate = randomDateObject.minusDays(differenceInDays);
23         }
24
25         System.out.println("Random date is " + randomDateObject + " that is a " + randomDay);
26         System.out.println("Date of the previous Thursday is " + thursdayDate);
27
28     }
29
30 }
31
```

Problems | Declaration | Console | Git Staging | JUnit | Call Hierarchy

```
<terminated> dateprac [Java Application] C:\Users\mmore\Desktop\sts-4.12.0.RELEASE\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_16.0.2.v20210721-1
Random date is 2021-10-29 that is a FRIDAY
Date of the previous Thursday is 2021-10-28
```

```
Random date is 2021-10-27 that is a WEDNESDAY
Date of the previous Thursday is 2021-10-21
```

3. What is the difference between a ZoneId and a ZoneOffset?
A ZoneId specifies a time zone identifier, meaning it represents a specific geographic area's time zone.
A ZoneOffset offsets that time zone from UTC time.

4. How would you convert an Instant to a ZonedDateTime? How would you convert a ZonedDateTime to an Instant?

```
Instant instant = Instant.now();
ZonedDateTime zdt = instant.atZone(ZoneId.of("America/Los_Angeles"));
```

```
ZonedDateTime zdt = ZonedDateTime.now(ZoneId.of("America/Los_Angeles"));
Instant instant = zdt.toInstant();
```

5. Write an example that, for a given year, reports the length of each month within that year.