

Exercise: MyDate, version 5

MyDate
- day : int - month : int - year : int
+ MyDate(day : int, month : int, year : int) + MyDate() + set(day : int, month : int, year : int) : void + getDay() : int + getMonth() : int + getYear() : int + getMonthName() : String + isLeapYear() : boolean + stepForwardOneDay() : void + stepForward(days : int) : void + numberOfDaysInMonth() : int + isBefore(other : MyDate) : boolean + yearsBetween(other : MyDate) : int + toString() : String

Create a new module and name it MyDate_v5 copying the files from MyDate_v4 (or MyDate_v3)

Modify class MyDate:

- a) Add a zero-argument constructor setting the date to today's date.

Hint: Use the following code as a guideline for how to get the current date.

```
import java.time.LocalDate;
public class CalendarTest
{
    public static void main(String[] args)
    {
        LocalDate today = LocalDate.now();

        System.out.println("Day = " + today.getDayOfMonth());
        System.out.println("Month = " + today.getMonthValue());
        System.out.println("Year = " + today.getYear());
    }
}
```

Note that the zero-arguments constructor sets the three instance variables and do NOT print out.

- b) Add a method `stepForward` updating the date to a new date a number of days in the future. The number of days is given as argument. Hint: the easiest solution is to make a loop and call method `stepForwardOneDay` in the loop body.

Modify the test application named MyDateTest such that you demonstrates class MyDate's new capabilities.