

### 3 Arbeiten mit den Standardklassen LocalDate und LocalTime

- 3.1. Gegeben ist die Main-Methode in der Klasse A0301KalenderTest\_v2 (siehe Moodle)  
In dieser Main-Methode werden verschiedene Klassenmethoden aus der Klasse  
A0301KalenderKlasse\_v2 aufgerufen.

#### Aufgabe:

Erstellen Sie die Klasse A0301KalenderKlasse\_v2 mit den Klassenmethoden, die in der Tabelle unten aufgelistet sind.

Zum Testen gehen Sie am besten so vor, dass Sie in der gegebenen Klasse in Main zunächst alles auskommentieren und dann versuchen Anweisung für Anweisung zum Laufen zu bringen, indem Sie die notwendigen Methoden implementieren.

String getDate(int iJahr, int iMonat, int iTag, int iFormat)	liefert das übergebene Datum im Format iFormat: 0: "Freitag, 18.Dezember 2009" 1: "18/12/09" 2: "18.Dezember 2009" 3: "18.Dezember.09" 4: "18.12.09" 5: "18/12/2009" 6: "2009-12-18" 7: "Dezember 09" 8: "18-Dezember-09" 9: "Dez.09"
String getDate(int iJahr, int iMonat, int iTag)	liefert das übergebene Datum im langen Format (FormatNr. 0)
String getAktuellesDatum(int iFormat) → ruft die Methode getDate(...) für das aktuelle Datum auf	liefert das aktuelle Datum im Format iFormat → benutzt die Methode getDate(...)
String getAktuellesDatum() → ruft die Methode getAktuellesDatum(0) auf	liefert das aktuelle Datum im langen Format (FormatNr. 0)
String getWochentag(int iJahr, int iMonat, int iTag)	
String getAktWochentag() → ruft die Methode getWochentag(...) für das aktuelle Datum auf	
String getAktuelleUhrzeit()	Liefert die aktuelle Uhrzeit im Format "hh:mm:ss"
int getAktStunde() int getAktMinute() int getAktSekunde()	
void druckeKalenderMonat(int iJahr, int iMonat)	z.B. 12/2000: Dezember 2000 Mo Di Mi Do Fr Sa So 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
void druckeAktKalenderMonat()	

```
public class A0301KalenderKlasse_v2
{
    public static String getDatum(int iJahr, int iMonat, int iTag, int iFormat)
    {
        String sDatumFormatiert;

        DateTimeFormatter einFormat;
        LocalDate datum = LocalDate.of(iJahr, iMonat, iTag);

        switch (iFormat)
        {
            case 0:
                einFormat = DateTimeFormatter.ofPattern("EEEE, dd. MMMM yyyy");
                sDatumFormatiert = datum.format(einFormat);
                break;
            case 1:
                einFormat = DateTimeFormatter.ofPattern("EE, dd. MMM. yy");
                sDatumFormatiert = datum.format(einFormat);
                break;
            case 2:
                einFormat = DateTimeFormatter.ofPattern("dd/MM/yy");
                sDatumFormatiert = datum.format(einFormat);
                break;
            case 3:
                einFormat = DateTimeFormatter.ofPattern("dd. MMMM yy");
                sDatumFormatiert = datum.format(einFormat);
                break;
            case 4:
                einFormat = DateTimeFormatter.ofPattern("d.M.yy");
                sDatumFormatiert = datum.format(einFormat);
                break;
            case 5:
                einFormat = DateTimeFormatter.ofPattern("d/M/yyyy");
                sDatumFormatiert = datum.format(einFormat);
                break;
            case 6:
                einFormat = DateTimeFormatter.ofPattern("yyyy-M-d");
                sDatumFormatiert = datum.format(einFormat);
                break;
            case 7:
                einFormat = DateTimeFormatter.ofPattern("MMMM yyyy");
                sDatumFormatiert = datum.format(einFormat);
                break;
            case 8:
                einFormat = DateTimeFormatter.ofPattern("d-MMMM-yy");
                sDatumFormatiert = datum.format(einFormat);
                break;
            case 9:
                einFormat = DateTimeFormatter.ofPattern("MMM. yyyy");
                sDatumFormatiert = datum.format(einFormat);
                break;
            default:
                sDatumFormatiert = null;
                break;
        }
        return (sDatumFormatiert);
    }
}
```

```
public static String getDatum(int iJahr, int iMonat, int iTag)
{
    return (A0301KalenderKlasse_v2.getDatum(iJahr, iMonat, iTag, 0));
}

public static String getAktuellesDatum()
{
    return (A0301KalenderKlasse_v2.getAktuellesDatum(0));
}

public static String getAktuellesDatum(int iFormat)
{
    LocalDate ldHeute = LocalDate.now();
    return A0301KalenderKlasse_v2.getDatum(ldHeute.getYear(),
        ldHeute.getMonthValue(),
        ldHeute.getDayOfMonth(), iFormat);
}

public static String getDatum(LocalDate ldDatum)
{
    return A0301KalenderKlasse_v2.getDatum(ldDatum.getYear(),
        ldDatum.getMonthValue(),
        ldDatum.getDayOfMonth(), 0);
}

public static String getAktWochentag()
{
    LocalDate ldHeute = LocalDate.now();
    return ldHeute.format(DateTimeFormatter.ofPattern("EEEE"));
}

public static void druckeAktKalenderMonat()
{
    LocalDate ldJetzt = LocalDate.now();
    A0301KalenderKlasse_v2.druckeKalenderMonat(ldJetzt.getYear(),
        ldJetzt.getMonthValue());
}

public static String getAktuelleUhrzeit()
{
    LocalTime ltJetzt = LocalTime.now();
    String sAktuell = ltJetzt.format(DateTimeFormatter.ofPattern("HH:mm:ss"));
    return (sAktuell);
}

public static int getAktStunde()
{
    LocalTime ltJetzt = LocalTime.now();
    return ltJetzt.getHour();
}

public static int getAktMinute()
{
    LocalTime ltJetzt = LocalTime.now();
    return ltJetzt.getMinute();
}

public static int getAktSekunde()
{
    LocalTime ltJetzt = LocalTime.now();
    return ltJetzt.getSecond();
}
```

```
public static void druckeKalenderMonat(int iJahr, int iMonat)
{
    LocalDate ldTag = LocalDate.of(iJahr, iMonat, 1);
    DayOfWeek dwTag;
    int iAnz;

    dwTag = ldTag.getDayOfWeek();

    System.out.println(A0301KalenderKlasse_v2.getDatum(iJahr, iMonat, 1, 7));
    System.out.println("Mo Di Mi Do Fr Sa So");

    for (iAnz = 1; iAnz < dwTag.getValue(); iAnz++)
    {
        System.out.print("  ");
    }

    for (iAnz = 1; iAnz <= ldTag.lengthOfMonth(); iAnz++)
    {
        System.out.printf("%02d ", iAnz);
        if ((iAnz + dwTag.getValue() - 1) % 7 == 0)
            System.out.println();
    }
    System.out.println();
}
```