```
import java.time.Instant;
import java.time.LocalDateTime;
import java.time.ZoneId;
import java.time.ZonedDateTime;
import java.util.Date;
import com.ibm.icu.text.DateFormat;
import com.ibm.icu.text.SimpleDateFormat;
import com.ibm.icu.util.Calendar;
import com.ibm.icu.util.ULocale;

public class DateTimeUtils {
   public static final ULocale PERSIAN_LOCALE = new ULocale("fa_IR@calendar=persian");
   public static final ULocale PERSIAN_EN_LOCALE = new ULocale("en@calendar=persian");
   public static final ZoneId IRAN_ZONE_ID = ZoneId.of("Asia/Tehran");

   public static Calendar fromDateToPersianCalendar(Date date) {
      Calendar persianCalendar = Calendar.getInstance(PERSIAN_LOCALE);
      persianCalendar.clear();
      persianCalendar.setTime(date);
      return persianCalendar;
   }

   /**
    * @param date
    * @param field example: Calendar.YEAR, Calendar.MONTH, Calendar.DAY_OF_MONTH, etc
    */
   public static int fromDateToPersianCalendarField(Date date, int field) {
      return fromDateToPersianCalendar(date).get(field);
   }

   public static String fromDateToPersianString(Date date) {
      DateFormat df = DateFormat.getDateInstance(DateFormat.FULL, PERSIAN_LOCALE);
      return df.format(date);
   }

   public static String fromDateToPersianString(Date date, String pattern) {
      return new SimpleDateFormat(pattern, PERSIAN_LOCALE).format(date);
   }

   public static String fromDateToPersianString(Date date, String pattern, ULocale locale) {
      return new SimpleDateFormat(pattern, locale).format(date);
   }

   /**
    * @param month is zero based. (e.g. Farvardin = 0, Ordibehesht = 1, etc.)
    */
   public static Date fromPersianDateToDate(int year, int month, int day, int hour, int minutes, int seconds) {
      return new Date(fromPersianDate(year, month, day, hour, minutes, seconds));
   }

   /**
    * @param month is zero based. (e.g. Farvardin = 0, Ordibehesht = 1, etc.)
    */
   public static String fromPersianDateToPersianString(int year, int month, int day, int hour, int minutes, int
seconds) {
      return fromDateToPersianString(fromPersianDateToDate(year, month, day, hour, minutes, seconds));
```

```
    }

    /**
     * @param month is zero based. (e.g. Farvardin = 0, Ordibehesht = 1, etc.)
     */
    public static LocalDateTime fromPersianDateToLocalDateTime(int year, int month, int day, int hour, int minutes,
int seconds) {
        return fromPersianDateToZonedDateTime(year, month, day, hour, minutes, seconds).toLocalDateTime();
    }

    /**
     * @param month is zero based. (e.g. Farvardin = 0, Ordibehesht = 1, etc.)
     */
    public static ZonedDateTime fromPersianDateToZonedDateTime(int year, int month, int day, int hour, int
minutes, int seconds) {
        return toZonedDateTime(fromPersianDate(year, month, day, hour, minutes, seconds));
    }

    /**
     * @param month is zero based. (e.g. Farvardin = 0, Ordibehesht = 1, etc.)
     */
    public static long fromPersianDate(int year, int month, int day, int hour, int minutes, int seconds) {
        Calendar persianCalendar = Calendar.getInstance(PERSIAN_LOCALE);
        persianCalendar.clear();
        persianCalendar.set(year, month, day, hour, minutes, seconds);
        return persianCalendar.getTimeInMillis();
    }

    public static ZonedDateTime toZonedDateTime(Long epochMilli) {
        if(epochMilli == null) return null;
        return Instant.ofEpochMilli(epochMilli).atZone(IRAN_ZONE_ID);
    }
}
```

Usage:

```
import java.time.LocalDateTime;
import java.time.ZonedDateTime;
import java.util.Date;
import com.ibm.icu.util.Calendar;

public class DateTimeUtilsTest {
    public static void main(String[] args) {
        System.out.println("Java 7 and before:");
        Date date = new Date(1467262800000L);
        System.out.println("Converting Gregorian date to Persian:");
        Calendar persianCalendar = DateTimeUtils.fromDateToPersianCalendar(date);
        System.out.println(persianCalendar.get(Calendar.YEAR));
        System.out.println(persianCalendar.get(Calendar.MONTH));
        System.out.println(persianCalendar.get(Calendar.DAY_OF_MONTH));
        System.out.println(DateTimeUtils.fromDateToPersianString(persianCalendar.getTime()));

        System.out.println("\nAdding 1 month and 5 days:");
        persianCalendar.add(Calendar.MONTH, 1); // add a month
        persianCalendar.add(Calendar.DAY_OF_MONTH, 5); // add 5 days
        System.out.println(persianCalendar.get(Calendar.YEAR));
```

```
        System.out.println(persianCalendar.get(Calendar.MONTH));
        System.out.println(persianCalendar.get(Calendar.DAY_OF_MONTH));
        System.out.println(DateTimeUtils.fromDateToPersianString(persianCalendar.getTime()));

        System.out.println("\nConverting Persian date to Gregorian:");
        Date gregorianDate = DateTimeUtils.fromPersianDateToDate(1395, 3, 10, 9, 30, 0);
        System.out.println(gregorianDate);
        System.out.println(DateTimeUtils.fromDateToPersianString(gregorianDate)); // to Persian string
        System.out.println(DateTimeUtils.fromDateToPersianString(gregorianDate, "dd/MM/yy - H:mm:dd")); // to
Persian string with custom format
        System.out.println(DateTimeUtils.fromDateToPersianString(gregorianDate, "dd/MM/yy - H:mm:dd" ,
DateTimeUtils.PERSIAN_EN_LOCALE)); // to Persian string with custom format and Latin characters

        System.out.println("\n"+"Java 8 onward:");
        ZonedDateTime gregorianZonedDateTime = DateTimeUtils.fromPersianDateToZonedDateTime(1395, 3, 10,
9, 30, 0);
        System.out.println(gregorianZonedDateTime);
        LocalDateTime gregorianLocalDateTime = DateTimeUtils.fromPersianDateToLocalDateTime(1395, 3, 10, 9,
30, 0);
        System.out.println(gregorianLocalDateTime);
    }
}
```

Output:

```
Java 7 and before:
Converting Gregorian date to Persian:
1395
3
10
تیر ۱۰, پنجشنبه ۱۳۹۵

Adding 1 month and 5 days:
1395
4
15
مرداد ۱۵, جمعه ۱۳۹۵

Converting Persian date to Gregorian:
Thu Jun 30 09:30:00 IRDT 2016
تیر ۱۰, پنجشنبه ۱۳۹۵
۹۵/۰۴/۱۰ - ۹:۳۰:۱۰
10/04/95 - 9:30:10

Java 8 onward:
2016-06-30T09:30+04:30[Asia/Tehran]
2016-06-30T09:30
```