

Joda Date and Time API

- Before Java 8, date/time API has multiple design problems
 - ✓ java.util.Date and SimpleDateFormat classes aren't thread safe
 - ✓ date class doesn't represent actual date instead it specifies an instant in time, with millisecond precision
 - ✓ years start from 1900
 - ✓ months start at index 0
 - ✓ Suppose if you want to represent a date of 31 Dec 2021 then you have to create the Date Object as follows.
 - **Date mydate = new Date(121, 12, 31);**
- To Avoid the Problems with Date, Calendar was introduced but Calendar also has similar problems and design flaws
- Finally Oracle decided to provide high quality date and time support in the native Java API. As a result, Java 8 integrates many of the Joda Time features in the java.time package
- APIs from java.time package
 - DayOfWeek (enum)
 - Month (enum)
 - Year
 - YearMonth
 - MonthDay
 - LocalDate
 - LocalTime
 - LocalDateTime
 - Instant
 - Duration
 - Period
 - Etc
- APIs from java.time.format package
 - TextStyle (enum)
- APIs from java.time.temporal package
 - TemporalAdjuster (Functional interface)
 - TemporalAdjusters (class)



Demo1.java

```
package com.jlcindia.demos;

import java.time.DayOfWeek;
import java.time.format.TextStyle;
import java.util.Locale;
/*
 * @Author : Srinivas Dande
 * @Company: Java Learning Center
 */

public class Demo1 {

    public static void main(String[] args) {

        DayOfWeek monday = DayOfWeek.MONDAY;
        System.out.println("Day " + monday);
        System.out.println("Day " + monday.getValue());

        Locale loc = Locale.getDefault();
        System.out.println("Full " + monday.getDisplayName(TextStyle.FULL, loc));
        System.out.println("Short " + monday.getDisplayName(TextStyle.SHORT, loc));
        System.out.println("Narrow " + monday.getDisplayName(TextStyle.NARROW, loc));
        System.out.println("-----");

        DayOfWeek sunday = DayOfWeek.SUNDAY;
        System.out.println("compareTo " + sunday.compareTo(monday));
        System.out.println("compareTo " + monday.compareTo(sunday));
        System.out.println("compareTo " + sunday.compareTo(sunday));

        System.out.println("-----");
        System.out.println("Day " + monday);

        System.out.println("-2 " + monday.minus(2));
        System.out.println("+2 " + monday.plus(2));
    }
}
```



Demo2.java

```
package com.jlcindia.demos;

import java.time.Month;
import java.time.format.TextStyle;
import java.util.Locale;
/*
 * @Author : Srinivas Dande
 * @Company: Java Learning Center
 */

public class Demo2 {
    public static void main(String[] args) {
        Month june = Month.JUNE;
        System.out.println("Month :"+june);
        System.out.println("Value :"+june.getValue());
        System.out.println("Max :"+june.maxLength());
        System.out.println("Min :"+june.minLength());

        Locale loc = Locale.getDefault();
        System.out.println("Full :"+june.getDisplayName(TextStyle.FULL, loc));
        System.out.println("Short :"+june.getDisplayName(TextStyle.SHORT, loc));
        System.out.println("Narrow :"+june.getDisplayName(TextStyle.NARROW, loc));
        System.out.println("-----");

        System.out.println("-2 :"+june.minus(2));
        System.out.println("+2 :"+june.plus(2));
        System.out.println("N of Days:"+june.length(false));
        System.out.println("N of Days (Leap Y):"+june.length(true));
        System.out.println("-----");

        System.out.println("\nDay of Year :"+Month.JANUARY.firstDayOfYear(false));
        System.out.println("Day of Year :"+Month.JANUARY.firstDayOfYear(true));
        System.out.println("\nDay of Year :"+Month.MARCH.firstDayOfYear(false));
        System.out.println("Day of Year :"+Month.MARCH.firstDayOfYear(true));
        System.out.println("\nMon of Quater :"+Month.JANUARY.firstMonthOfQuarter());
        System.out.println("Mon of Quater :"+Month.MAY.firstMonthOfQuarter());
        System.out.println("Mon of Quater :"+Month.SEPTEMBER.firstMonthOfQuarter());
        System.out.println("Mon of Quater :"+Month.DECEMBER.firstMonthOfQuarter());
    }
}
```



Demo3.java

```
package com.jlcindia.demos;

import java.time.LocalDate;
/*
 * @Author : Srinivas Dande
 * @Company: Java Learning Center
 */

public class Demo3 {
    public static void main(String[] args) {

        LocalDate date = LocalDate.of(2016, 2, 5);
        System.out.println("YEAR :"+date.getYear());
        System.out.println("MONTH :"+date.getMonth());
        System.out.println("MONTH VALUE :"+date.getMonthValue());
        System.out.println("DATE :"+date.getDayOfMonth());
        System.out.println("DAYOF WEEK :"+date.getDayOfWeek());
        System.out.println("DAYOF WEEK VALUE :"+date.getDayOfWeek().getValue());
        System.out.println("DAY OF YEAR :"+date.getDayOfYear());
        System.out.println("Month Len :"+date.lengthOfMonth());
        System.out.println("Year Len :"+date.lengthOfYear());

        LocalDate date1 = LocalDate.now();
        System.out.println("\nYEAR :"+date1.getYear());
        System.out.println("MONTH :"+date1.getMonth());
        System.out.println("MONTH VALUE :"+date1.getMonthValue());
        System.out.println("DATE :"+date1.getDayOfMonth());
        System.out.println("DAYOF WEEK :"+date1.getDayOfWeek());
        System.out.println("DAYOF WEEK VALUE :"+date1.getDayOfWeek().getValue());
        System.out.println("DAY OF YEAR :"+date1.getDayOfYear());
        System.out.println("Month Len :"+date1.lengthOfMonth());
        System.out.println("Year Len :"+date1.lengthOfYear());
    }
}
```



Demo4.java

```
package com.jlcindia.demos;

import java.time.DayOfWeek;
import java.time.LocalDate;
import java.time.Month;
import java.time.ZoneId;
import java.time.temporal.TemporalAdjusters;
/*
 * @Author : Srinivas Dande
 * @Company: Java Learning Center
 */
public class Demo4 {
    public static void main(String[] args) {

        LocalDate date = LocalDate.now();
        System.out.println(date);
        LocalDate nextWedDate = date.with(TemporalAdjusters.next(DayOfWeek.WEDNESDAY));
        System.out.println("nextWed Date :" + nextWedDate);
        LocalDate lastWedDate =
        date.with(TemporalAdjusters.lastInMonth(DayOfWeek.WEDNESDAY));
        System.out.println("lastWedDate:" + lastWedDate);
        System.out.println("-2 Year :" + date.minusYears(2));
        System.out.println("-2 Mon :" + date.minusMonths(4));
        System.out.println("-2 Wk :" + date.minusWeeks(2));
        System.out.println("-2 Day :" + date.minusDays(2));
        System.out.println("-----");
        System.out.println(date);
        System.out.println("+2 Year :" + date.plusYears(2));
        System.out.println("+2 Mon :" + date.plusMonths(2));
        System.out.println("+2 Wk :" + date.plusWeeks(2));
        System.out.println("+2 Day :" + date.plusDays(2));
        System.out.println(date);
        System.out.println("-----");
        System.out.println(LocalDate.MIN);
        System.out.println(LocalDate.MAX);
        System.out.println("-----");
        LocalDate currentDateInLosAngeles = LocalDate.now(ZoneId.of("America/Los_Angeles"));
        LocalDate currentDateInLocalSystem = LocalDate.now(ZoneId.systemDefault());
        System.out.println(currentDateInLosAngeles);
        System.out.println(currentDateInLocalSystem);
    }
}
```



Demo5.java

```
package com.jlcindia.demos;

import java.time.LocalDateTime;
import java.time.ZoneId;
/*
 * @Author : Srinivas Dande
 * @Company: Java Learning Center
 */
public class Demo5 {
    public static void main(String[] args) {

        LocalDateTime time = LocalDateTime.of(15, 23, 45);
        System.out.println(time);
        System.out.println("Hour :"+time.getHour());
        System.out.println("Min :"+time.getMinute());
        System.out.println("Sec :"+time.getSecond());
        System.out.println("Nano Sec :"+time.getNano());
        System.out.println("-----");

        System.out.println("-2 Hour :"+time.minusHours(2));
        System.out.println("-2 Min :"+time.minusMinutes(2));
        System.out.println("-2 Sec :"+time.minusSeconds(2));
        System.out.println("+2 Hour :"+time.plusHours(2));
        System.out.println("+2 Min :"+time.plusMinutes(2));
        System.out.println("+2 Sec :"+time.plusSeconds(2));
        System.out.println("-----");

        System.out.println(time);
        System.out.println("with Hour-12 :"+time.withHour(12));
        System.out.println("with Min-34 :"+time.withMinute(34));
        System.out.println("with Sec-56 :"+time.withSecond(56));
        System.out.println("-----");

        System.out.println("MIN "+LocalTime.MIN);
        System.out.println("MID "+LocalTime.MIDNIGHT);
        System.out.println("NOON "+LocalTime.NOON);
        System.out.println("MAX "+LocalTime.MAX);
        System.out.println("now "+LocalTime.now());
        System.out.println("-----");
    }
}
```



```
LocalTime currentTimeInLosAngeles = LocalTime.now(ZoneId.of("America/Los_Angeles"));
LocalTime currentTimeInLocalSystem = LocalTime.now(ZoneId.systemDefault());

System.out.println(currentTimeInLosAngeles);
System.out.println(currentTimeInLocalSystem);

}
}
```

Demo6.java

```
package com.jlcindia.demos;

import java.time.LocalDateTime;
import java.time.ZoneId;
/*
 * @Author : Srinivas Dande
 * @Company: Java Learning Center
 */

public class Demo6 {
    public static void main(String[] args) {
        LocalDateTime dateTime = LocalDateTime.now();
        System.out.println(dateTime);
        System.out.println("Year :"+dateTime.getYear());
        System.out.println("Month :"+dateTime.getMonth());
        System.out.println("Mon Value :"+dateTime.getMonthValue());
        System.out.println("Date :"+dateTime.getDayOfMonth());
        System.out.println("Hour :"+dateTime.getHour());
        System.out.println("Min :"+dateTime.getMinute());
        System.out.println("Sec :"+dateTime.getSecond());
        System.out.println("Nano Sec :"+dateTime.getNano());
        System.out.println("-----");
        System.out.println("-2 Hour :"+dateTime.minusHours(2));
        System.out.println("-2 Min :"+dateTime.minusMinutes(2));
        System.out.println("-2 Sec :"+dateTime.minusSeconds(2));
        System.out.println("+2 Hour :"+dateTime.plusHours(2));
        System.out.println("+2 Min :"+dateTime.plusMinutes(2));
        System.out.println("+2 Sec :"+dateTime.plusSeconds(2));
        System.out.println(dateTime);
        System.out.println("-----");
    }
}
```



```
System.out.println("with Hour-12 :"+dateTime.withHour(12));
System.out.println("with Year-2019 :"+dateTime.withYear(2019));
LocalDateTime currentTimeInLosAngeles =
LocalDateTime.now(ZoneId.of("America/Los_Angeles"));
LocalDateTime currentTimeInLocalSystem = LocalDateTime.now(ZoneId.systemDefault());
System.out.println(currentTimeInLosAngeles);
System.out.println(currentTimeInLocalSystem);

}
}
```

Demo7.java

```
package com.jlcindia.demos;

import java.time.Instant;
import java.time.LocalDateTime;
import java.time.ZoneId;
import java.time.temporal.ChronoUnit;
import java.util.Date;
/*
 * @Author : Srinivas Dande
 * @Company: Java Learning Center
 */
public class Demo7 {
    public static void main(String[] args) {
        System.out.println(Instant.MIN);
        System.out.println(Instant.MAX);
        System.out.println(Instant.EPOCH);

        Instant instant = Instant.now();
        System.out.println(instant);
        System.out.println(instant.minus(2, ChronoUnit.MILLIS));
        System.out.println(instant.minusMillis(2));
        System.out.println(instant.getEpochSecond());

        long secondsFromEpoch = Instant.ofEpochSecond(0L).until(Instant.now(),
ChronoUnit.SECONDS);
        System.out.println(secondsFromEpoch);
        long millisFromEpoch = Instant.ofEpochSecond(0L).until(Instant.now(), ChronoUnit.MILLIS);
        System.out.println(millisFromEpoch);
    }
}
```




```
Date dt = new Date();
System.out.println(dt.getTime());
LocalDateTime dateTime = LocalDateTime.ofInstant(instant, ZoneId.systemDefault());
LocalDateTime dateTimeUTC = LocalDateTime.ofInstant(instant, ZoneId.of("UTC"));
System.out.println(instant);
System.out.println(dateTimeUTC);
System.out.println(dateTime);
}
}
```

Demo8.java

```
package com.jlcindia.demos;

import java.time.Duration;
import java.time.Instant;
import java.time.temporal.ChronoUnit;
/*
 * @Author : Srinivas Dande
 * @Company: Java Learning Center
 */
public class Demo8 {
    public static void main(String[] args) {

        Instant t1 = Instant.now();
        Instant t2 = Instant.parse("2017-03-05T10:15:30.00Z");

        Duration d1 = Duration.between(t1, t2);
        System.out.println(d1.toDays());
        System.out.println(d1.toHours());
        System.out.println(d1.toMinutes());

        long value = ChronoUnit.DAYS.between(t1, t2);
        System.out.println(value);

        Duration gap = Duration.ofSeconds(10);
        Instant t3 = t1.plus(gap);

        System.out.println(t1);
        System.out.println(t3);
        long value1 = ChronoUnit.SECONDS.between(t1, t3);
        System.out.println(value1);
    }
}
```



Demo9.java

```
package com.jlcindia.demos;

import java.time.Duration;
import java.time.LocalDate;
import java.time.LocalDateTime;
import java.time.Month;
import java.time.Period;
import java.time.temporal.ChronoUnit;
/*
 * @Author : Srinivas Dande
 * @Company: Java Learning Center
 */
public class Demo9 {
    public static void main(String[] args) {

        LocalDateTime today = LocalDateTime.now();
        LocalDateTime birthday = LocalDateTime.of(1978, Month.JUNE, 16, 9, 00, 00);
        Duration gap = Duration.between(birthday, today);
        System.out.println(gap.toDays());

        LocalDate today1 = LocalDate.now();
        LocalDate birthday1 = LocalDate.of(1978, Month.JUNE, 16);

        Period p1 = Period.between(birthday1, today1);
        long days1 = ChronoUnit.DAYS.between(birthday, today);

        System.out.println("You are " + p1.getYears() + " years, " + p1.getMonths() + " months, and " +
            p1.getDays() + " days old. (" + days1 + " days total)");

    }
}
```