

DEV WORKSPACE

Powered by Ad Hoc Minds



DATETIME MIGLIORI CON JAVA 8

SPEAKER FEDERICO PAREZZAN

> 17 FEBBRAIO

> ORE 18:45

ENJOY THE SPACE_FOLLOW US: [f](#) [in](#) [e](#)

Federico Parezzan



Huawei Developer Expert



Oracle Certified Associate



Sviluppatori Flutter



[federico-parezzan](#)



[parez93](#)



[parezzan.com](#)



[thecmmbay.com](#)

Sono un essere umano, nulla che sia umano mi è estraneo - Terenzio

Agenda



A LOOK TO THE PAST

*How to manage dates
in early java versions*



JAVA 8+ API

*What's new with
version 8*



DATABASE HINTS

*How to map the new
API in your database*



PARSING AND FORMATTING HINTS

*How to parse and format
with the new API*



A LOOK TO THE PAST

How to manage dates in early java versions

DATE & CALENDAR

Lorem Ipsum is simply dummy text of the printing and typesetting industry.



POOR API DESIGN

I.e. months start with 1 and days with 0



DATE AS TIME INTERVAL

Time interval in millisecond since 1/1/1970 until object creation calculated with the jvm time zone



NO IMMUTABILITY

All methods edit the original object state instead of make an object copy



NO THREAD SAFE

Unsuitable for concurrent applications



FORMATTING CLASSES WORK ONLY WITH DATE

Utility classes for formatting are available only for Date and not for Calendar



JAVA 8+ API

What's new with version 8

JAVA 8+ API

Lorem Ipsum is simply dummy text of the printing and typesetting industry.

The new APIs are in the following packages: java.time, java.time.chrono, java.time.format, java.time.temporal, java.time.zone



API CLARITY

Developers keep in mind use cases when designed the apis



IMMUTABILITY

Methods return an object copy and don't edit the object



DATE AS VALUE

Date is a value and not a time interval



THREAD SAFE

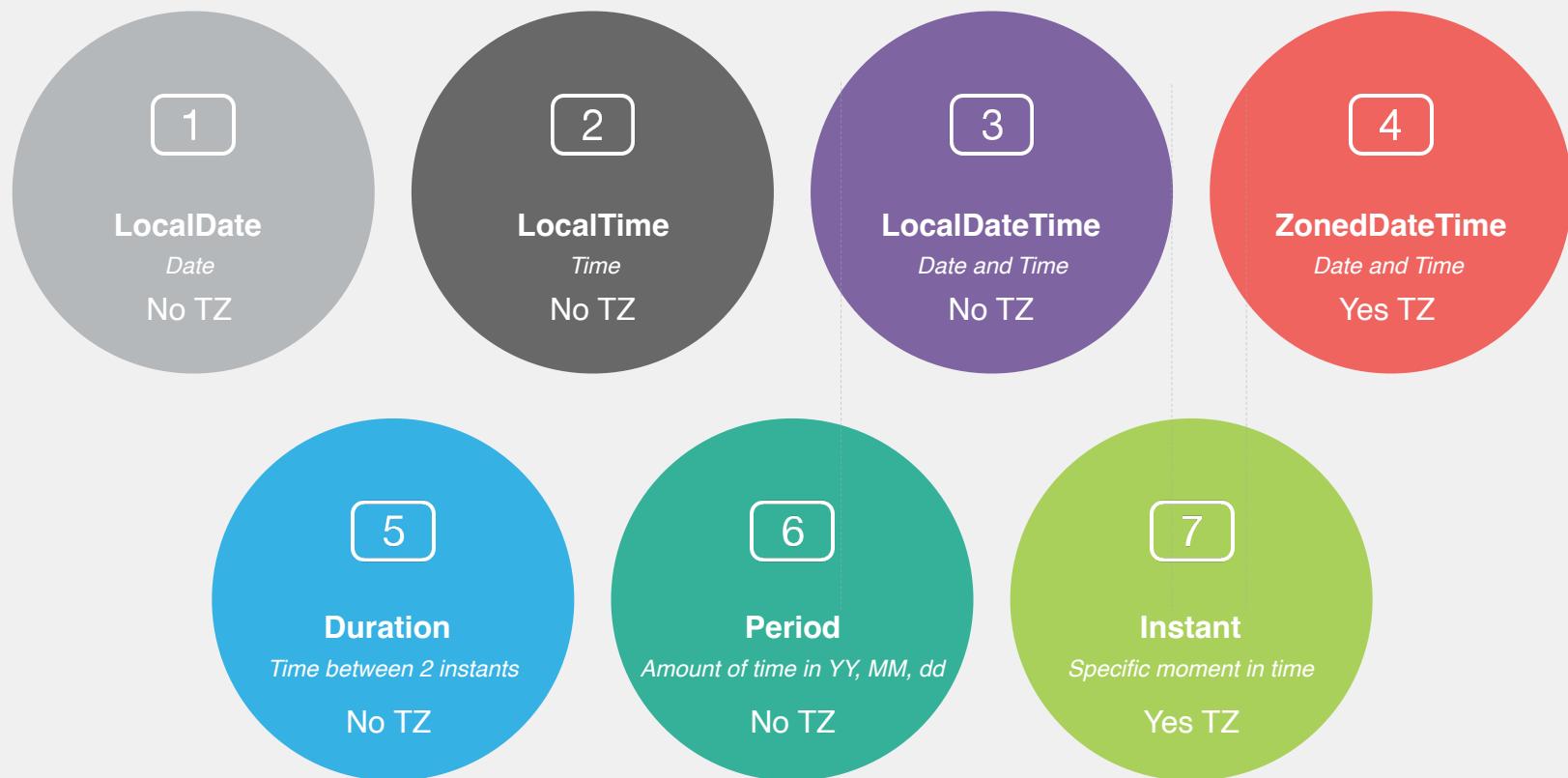
Date can be used in concurrent applications without side-effects



DEMO

API

Lorem Ipsum is simply dummy text of the printing and typesetting industry.





DEMO





DATABASE HINTS

How to map the new API in your database



DATABASE TYPES MAPPING

Lorem Ipsum is simply dummy text of the printing and typesetting industry.

SQL	JAVA 8+
DATE	LocalDate
TIME	LocalTime
TIMESTAMP	LocalDateTime
TIME WITH TIME ZONE	OffsetTime
TIMESTAMP WITHOUT TIME ZONE	OffsetDateTime



PARSING AND FORMATTING HINTS

How to parse and format with the new API

PARSING & FORMATTING

Lorem Ipsum is simply dummy text of the printing and typesetting industry.

```
● ● ●  
LocalDateTime dateTime = LocalDateTime  
        .of(2022, 9, 1, 13, 14);  
DateTimeFormatter formatter = DateTimeFormatter  
        .ofPattern("dd/MM/yyyy HH:mm:ss");  
String formattedDateTime = dateTime.format(formatter);  
System.out.println(formattedDateTime);
```

PARSING

PARSING & FORMATTING

Lorem Ipsum is simply dummy text of the printing and typesetting industry.

FORMATTING



```
LocalDateTime parsedDate = LocalDateTime  
    .parse(formattedDateTime, formatter);  
System.out.println(parsedDate);
```

Cheatsheet

Lorem Ipsum is simply dummy text of the printing and typesetting industry.

Java Date Time in Java 8	
java 1.0	java.util.Date;java.text.Format
java 1.2	java.util.Calendar
java 8	java.time.*
java 8 classes: LocalDate, LocalTime, LocalDateTime, Month (1-12), Period, Duration	

Creating Dates and Times	
of()	Overridden by LocalDate, LocalTime and LocalDateTime
LocalDate.of(y,M,d);	M: int or Month(1-12)
LocalTime.of(h,m,s,ns)	ns: nano second
LocalDateTime.of(y,M,d,h,m,s,ns)	LocalDateTime.of(d,t)
now()	Overridden by LocalDate, LocalTime and LocalDateTime
parse(s,f)	Overridden by LocalDate, LocalTime and LocalDateTime
LocalDateTime.parse(s,f)	f must be DateTimeFormatter
LocalDate d= new LocalDate() //not compiled	
LocalDate d= LocalDate.of(2020,10,27); //not compile	

Manipulating Dates and Times	
plusYYY(int)	
minusYYY(int)	
return new immutable, original not changed, like new String()	
chain of method on date and time	
getXXX(); //XXX= DayOfWeek,DayOfYear, Month, Year	
dt1.isBefore(dt2)	
dt1.isAfter(dt2)	
YYY = Years / Months /Days/Hours/Minutes/Seconds/Nanos;	
Java 8 date & time are immutable, like String objects.	

Period and Duration
java.time.Period; java.time.Duration;
4 ways to create a period
Period.ofXXX(int); //XXX=Years,Months,Days
Period.of(int years,int months, int days);
Period.between(d1,d2)
Period.parse("P1Y2M3W4D");//same as Period.of(1, 2, 25)
Modify period
plusYYY(int); //YYY = Days Weeks Months Years
minusYYY(int); //YYY = Days Weeks Months Years
Use period
dt.plus/minus(period);
d.plus/minus(period);
time.plus/minus(duration); //if input period, UnsupportedTemporalTypeException
Chain of Periods method does not make sense
Formatting
2 methods to create a formatter
DateTimeFormatter f=DateTimeFormatter.ofXXX(FormatStyle.SHORT MEDIUM LONG FULL);
DateTimeFormatter f=DateTimeFormatter.ofPattern("yyyy-MMMMd-dd hh:mm:ss"); //M->1, MM->01, MMM->Jan, MMMM->January
Standard formatter:
DateTimeFormatter.ISO_LOCAL_DATE/TIME/DATETIME
ISO_LOCAL_DATETIME: '2020-10-27T1:45:30'
3 ways to use formatter
f.format(date/time/datetime_obj);
date/time/datetime_obj.format(f);
LocalDate/Time/DateTime.parse(String s, DateTimeFormatter f);
XXX = LocalizedDate, LocalizedTime, LocalizedDateTime
date and time format legal for datatime object, datatime format not legal for date or time object



THANKS

ASK A QUESTION