

Java-Success.com

Industrial strength Java/JEE Career Companion for those who want to go places

[Home](#)
[Java FAQs](#)
[600+ Java Q&As](#)
[Career](#)
[Tutorials](#)
[Member](#)
[Why?](#)
[Can u Debug?](#)
[Java 8 ready?](#)
[Top X](#)
[Productivity Tools](#)
[Judging Experience?](#)

[Home](#) › [Interview](#) › [Core Java Interview Q&A](#) › [Data types](#) › 04: Understanding TimeZones with examples in Java 8

04: Understanding TimeZones with examples in Java 8

Posted on [November 7, 2015](#) by [Arulkumaran Kumaraswamipillai](#)



Tweet



What is UTC? which stands for **Coordinated Universal Time**. The diagram below shows the standard time offsets of different countries. Some places observe daylight saving time (**DST**) during their respective summer periods. For example,

Los Angeles (i.e LA) UTC – 8.0 hours (without DST) & UTC – 7.0 hours (with DST)

Berlin UTC + 3.0 hours (without DST) & UTC + 2.0 hours (with DST).

[9 tips to earn more](#) | [What can u do to go places?](#) | **945+** members. [LinkedIn Group](#). [Reviews](#)

600+ Full Stack Java/JEE Interview Q&As ♥Free ♦FAQs

[open all](#) | [close all](#)

[Ice Breaker Interview](#)

- 01: ♦ 15 Ice breake
- 02: ♥♦ 8 real life ex
- 03: ♦10+ Know you
- 04: Can you think c
- 05: ♥ What job inte
- 06: ► Tell me abou
- 07: ♥ 20+ Pre inter

[Core Java Interview C](#)

[Java Overview \(4\)](#)

01: ♦ ♥ 17 Java c



Lets take a reference datetime as **2015-05-15 9:00am** and represent it with timezone for LA and Berlin. Note that 15th of May is with **DST** in both LA and Berlin. Here is the basic code with time zones.

```

1
2 package test;
3
4 import java.time.LocalDateTime;
5 import java.time.ZoneId;
6 import java.time.ZonedDateTime;
7
8 public class Converting {
9
10     public static void main(String[] args) {
11
12         // 2015-05-15 9:00am
13         LocalDateTime utcDateTime = LocalDateTime.of(2015, 5, 15, 9, 0, 0);
14
15         ZoneId zoneLA = ZoneId.of("America/Los_Angeles");
16         ZoneId zoneBerlin = ZoneId.of("Europe/Berlin");
17
18         ZonedDateTime _9AmInLA = ZonedDateTime.of(utcDateTime, zoneLA);
19         ZonedDateTime _9AmInBerlin = ZonedDateTime.of(utcDateTime, zoneBerlin);
20
21         //UTC-7.00 hours: _9AmInLA = 2015-05-15T02:00:00-07:00
22         System.out.println("_9AmInLA = " + _9AmInLA);
23
24         //UTC+2.00 hours: _9AmInBerlin = 2015-05-15T11:00:00+02:00
25         System.out.println("_9AmInBerlin = " + _9AmInBerlin);
26
27         //if 9am in LA then UTC (9 am + 7 hours)
28         System.out.println("if 9am in LA then UTC is " + _9AmInLA.toInstant().toEpochSecond());
29
30         //if 9am in Berlin then UTC (9 am - 2 hours)
31         System.out.println("if 9am in Berlin then UTC is " + _9AmInBerlin.toInstant().toEpochSecond());
32
33         boolean laDaylightSavings = zoneLA.getRules().isDaylightSavings(_9AmInLA);
34         boolean berlinDaylightSavings = zoneBerlin.getRules().isDaylightSavings(_9AmInBerlin);
35
36         System.out.println("laDaylightSavings=" + laDaylightSavings);
37         System.out.println("berlinDaylightSavings=" + berlinDaylightSavings);
38     }

```

- 02: ♥♦ Java Con
- 03: ♦ 9 Core Jav
- 04: ♦ Top 10 mos
- ☐ Data types (6)
 - 01: Java data ty
 - 02: ♥♦ 10 Java S
 - 03: ♦ ♥ Java aut
 - 04: Understandir
 - 05: Java primitiv
 - Working with Da
- ☐ constructors-methc
 - Java initializers,
- ☐ Reserved Key Wor
 - ♥♦ 6 Java Modifi
 - Java identifiers
- ☐ Classes (3)
 - ♦ Java abstract c
 - ♦ Java class loa
 - ♦ Java classes a
- ☐ Objects (8)
 - ▶ Beginner Jav
 - ♥♦ HashMap & H
 - ♦ 5 Java Object i
 - ♦ Java enum inte
 - ♦ Java immutabl
 - ♥♦ Object equals
 - Java serialization
 - Mocks, stubs, dc
- ☐ OOP (10)
 - ♥ Design princip
 - ♦ 30+ FAQ Java
 - ♦ Why favor cor
 - 08: ♦ Write code
 - Explain abstracti
 - How to create a
 - Top 5 OOPs tips
 - Top 6 tips to go
 - Understanding C
 - What are good r
- ☐ GC (2)
 - ♦ Java Garbage

```
39 }
40
```

Output:

```
_9AmlnLA = 2015-05-15T09:00-07:00[America/Los_Angeles]
_9AmlnBerlin = 2015-05-15T09:00+02:00[Europe/Berlin]
if 9am in LA then UTC (9 am + 7 hours) = 2015-05-
15T16:00:00Z
if 9am in Berlin then UTC (9 am – 2 hours) = 2015-05-
15T07:00:00Z
laDaylightSavings=true
berlinDaylightSavings=true
```

Flying from LA to Berlin in 12 Hours

Extending the above example, let us assume that you are flying from LA to Berlin.

Flying Time: 12 Hours

Total Time Difference (with DST): 9 hours (i.e. 7 + 2), derived from LA is UTC-7 and Berlin is UTC+2.

Departure Time in LA: 2015-05-15 9:00

Arrival Time as per LA: 2015-05-15 21:00 (i.e. 2015-05-15 9:00 + 12 HOURS flying time)

Departure Time as per Berlin: 2015-05-15 18:00 (i.e. 2015-05-15 9:00 + 9 HOURS Total Time difference)

Arrival Time in Berlin: 2015-05-16 06:00 (i.e. 2015-05-15 18:00 + 12 HOURS flying time)

In Short, you depart LA at LA time **2015-05-15 9:00** and arrive in Berlin at Berlin time **2015-05-16 06:00**, which is 6.00 am next day. A total of 21 hours (i.e. 12 HOURS flying time + 9 HOURS zone offset).

Here is the Java 8 code:

03: Java GC tun

Generics (5)

♥ Java Generics

♥ Overloaded m

♦ 12 Java Gener

♦ 7 rules to reme

3 scenarios to ge

FP (8)

01: ♦ 19 Java 8 I

02: ♦ Java 8 Stre

03: ♦ Functional

04: ♥♦ Top 6 tips

05: ♥ 7 Java FP

Fibonacci numb

Java 8 String str

Java 8: What is c

IO (7)

♥ Reading a text

♦ 15 Java old I/C

06: ♥ Java 8 way

Processing large

Processing large

Read a text file f

Reloading config

Multithreading (12)

01: ♥♦ 15 Beginn

02: ♥♦ 10+ Java

03: ♦ More Java

04: ♦ 6 popular J

05: ♦ How a thre

06: ♦ 10+ Atomic

07: 5 Basic multi

08: ♦ ThreadLoc

09: Java FutureT

10: ♦ ExecutorSe

Java ExecutorSe

Producer and Co

Algorithms (5)

♦ Splitting input t

♦ Tree traversal :

♥ ♦ Java coding

```

1
2 package test;
3
4 import java.time.LocalDateTime;
5 import java.time.ZoneId;
6 import java.time.ZonedDateTime;
7
8 public class ZoneBasics {
9
10     public static void main(String[] args) {
11
12         // 2015-05-15 9:00am
13         LocalDateTime utcDateTime = LocalDateTime.now();
14
15         ZoneId zoneLA = ZoneId.of("America/Los_Angeles");
16         ZoneId zoneBerlin = ZoneId.of("Europe/Berlin");
17
18         ZonedDateTime _9AmInLA = ZonedDateTime.of(
19             utcDateTime.toLocalDate(),
20             utcDateTime.toLocalTime(),
21             zoneLA,
22             ZoneOffset.UTC);
23
24         //LA to berlin flying time 12 hours.
25         ZonedDateTime arrivalTimeAsPerLA = _9AmInLA
26             .plusHours(12);
27
28         //arrivalTimeAsPerLA = 2015-05-15T21:00-07:00[America/Los_Angeles]
29         System.out.println("arrivalTimeAsPerLA = " + arrivalTimeAsPerLA);
30
31         ZonedDateTime arrivalTimeAtBerlin = arrivalTimeAsPerLA
32             .withZoneSameInstant(zoneBerlin);
33         System.out.println("arrivalTimeAtBerlin = " + arrivalTimeAtBerlin);
34     }
35 }

```

Output:

```

arrivalTimeAsPerLA = 2015-05-15T21:00-07:00[America/Los_Angeles]
arrivalTimeAtBerlin = 2015-05-16T06:00+02:00[Europe/Berlin]

```

Popular Posts

◆ 11 Spring boot interview questions & answers

856 views

◆ Q11-Q23: Top 50+ Core on Java OOP Interview Questions & Answers

825 views

18 Java scenarios based interview Questions and Answers

447 views

Searching algorithm
Swapping, partitioning
Annotations (2)
8 Java Annotations
More Java annotations
Collection and Data Structures
Find the first non-repeating character in a string
Java Collections
Java Iterable & Iterator
HashMap & HashSet
Sorting objects
02: Java 8 Streams
04: Understanding TimeZones with examples in Java 8
4 Java Collections
If Java did not have
Java 8: Different
Part-3: Java Tree
Sorting a Map by
When to use which
Differences Between
Java Iterable & Iterator
Multithreading
Why do Proxy,
Core Java Modifications
Differences between
Java Collections
Event Driven Programming
Event Driven Programming
Event Driven Programming
Exceptions (2)
Java exception
Top 5 Core Java
Java 7 (2)
Java 7 fork and join
Java 7: Top 8 new
Java 8 (24)
01: 19 Java 8 Interview
02: Java 8 Streams
03: Functional
04: Top 6 tips
04: Convert Lists

001A: ♦ 7+ Java integration styles & patterns interview questions & answers

400 views

♦ 7 Java debugging interview questions & answers

311 views

♦ 10 ERD (Entity-Relationship Diagrams) Interview Questions and Answers

301 views

01b: ♦ 13 Spring basics Q8 – Q13 interview questions & answers

292 views

01: ♦ 15 Ice breaker questions asked 90% of the time in Java job interviews with hints

286 views

♦ Q24-Q36: Top 50+ Core on Java classes, interfaces and generics interview questions & answers

263 views

8 Git Source control system interview questions & answers

215 views

Bio

Latest Posts



Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via Amazon.com in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.



About Arulkumaran Kumaraswamipillai

Mechanical Eng to freelance Java developer in 3 yrs. Contracting since 2003, and attended 150+ Java job

04: Understanding

05: ♥ 7 Java FP

05: ♦ Finding the

06: ♥ Java 8 way

07: ♦ Java 8 API

08: ♦ Write code

10: ♦ ExecutorSe

Fibonacci numbe

Java 8 String str

Java 8 using the

Java 8: 7 useful

Java 8: Different

Java 8: Does "O

Java 8: What is c

Learning to write

Non-trivial Java &

Top 6 Java 8 fea

Top 8 Java 8 fea

Understanding J

JVM (6)

♦ Java Garbage

01: jvisualvm to s

02: jvisualvm to c

05: Java primitiv

06: ♦ 10+ Atomic

5 JMX and MBes

Reactive Program

07: Reactive Pro

10: ♦ ExecutorSe

3. Multi-Threadir

Swing & AWT (2)

5 Swing & AWT i

Q6 – Q11 Swing

JEE Interview Q&A (3

JEE Overview (2)

♦ 8 Java EE (aka

Java EE interview

Web basics (8)

01: ♦ 12 Web ba

02: HTTP basics

03: Servlet inter

interviews, and often got 4 - 7 job offers to choose from. It pays to prepare. So, published Java interview Q&A books via [Amazon.com](https://www.amazon.com) in 2005, and sold 35,000+ copies. Books are outdated and replaced with this subscription based site.

◀ Convert Java Date/Time to String & String back to Date/Time

Converting String to Amount and Amount to String ▶

Posted in Data types, Java 8

04: JSP overview

05: Web patterns

06: ♦ MVC0, MV

07: When to use

08: Web.xml inte

WebService (11)

01: ♥♦ 40+ Java

02: ♦ 6 Java RE

03: ♥ JAX-RS hc

04: 5 JAXB inter

05: RESTful We

06: RESTful Wel

07: HATEOAS R

08: REST constr

09: 11 SOAP We

10: SOAP Web S

11: ♥ JAX-WS hc

JPA (2)

10: Spring, Java

8 JPA interview c

JTA (1)

JTA interview Q&

JDBC (4)

♦ 12 FAQ JDBC

JDBC Overview

NamedParamete

Spring, JavaCon

JMS (5)

♦ 16 FAQ JMS ir

Configuring JMS

JMS versus AMQ

Spring JMS with

Spring JMS with

JMX (3)

5 JMX and MBe

Event Driven Pr

Yammer metrics

JNDI and LDAP (1)

JNDI and LDAP

Pressed for time? Jav

Job Interview Ice B

	01: ♦ 15 Ice brea
	02: ♥♦ 8 real life
	03: ♦10+ Know y
☐	FAQ Core Java Jot
	♥♦ Q1-Q10: Top
	♦ Q11-Q23: Top
	♦ Q24-Q36: Top
	♦ Q37-Q42: Top
	♦ Q43-Q54: Top
	01: ♥♦ 15 Beginr
	02: ♥♦ 10+ Java
☐	FAQ JEE Job Inter
	♦ 12 FAQ JDBC
	♦ 16 FAQ JMS ir
	♦ 8 Java EE (aka
	♦ Q01-Q28: Top
	♦ Q29-Q53: Top
	01: ♦ 12 Web ba
	06: ♦ MVC0, MV
	JavaScript mista
	JavaScript Vs Ja
	JNDI and LDAP
	JSF interview Q&
	JSON interview (
☐	FAQ Java Web Ser
	01: ♥♦ 40+ Java
	02: ♦ 6 Java RES
	05: RESTFul We
	06: RESTful Wel
	09: 11 SOAP We
☐	Java Application Ar
	001A: ♦ 7+ Java
	001B: ♦ Java arc
	04: ♦ How to go
☐	Hibernate Job Inter
	01: ♥♦ 15+ Hiber
	01b: ♦ 15+ Hiber
	06: Hibernate Fil
	8 JPA interview c
☐	Spring Job Intervie
	♦ 11 Spring boot

	01: ♥♦ 13 Spring
	01b: ♦ 13 Spring
	04 ♦ 17 Spring b
	05: ♦ 9 Spring B
☐	Java Key Area Ess
	♦ Design pattern
	♥ Top 10 causes
	♥♦ 01: 30+ Writir
	♦ 12 Java desigr
	♦ 18 Agile Develo
	♦ 5 Ways to debu
	♦ 9 Java Transac
	♦ Monitoring/Pro
	02: ♥♦ 13 Tips to
	15 Security key :
	4 FAQ Performa
	4 JEE Design Pa
	5 Java Concurr
	6 Scaling your Ja
	8 Java memory i
☐	OOP & FP Essenti
	♦ 30+ FAQ Java
	01: ♦ 19 Java 8 I
	04: ♥♦ Top 6 tips
☐	Code Quality Job I
	♦ Ensuring code
	♦ 5 Java unit tes
☐	SQL, XML, UML, JSC
☐	ERD (1)
	♦ 10 ERD (Entity
☐	NoSQL (2)
	♦ 9 Java Transac
	3. Understanding
☐	Regex (2)
	♥♦ Regular Expr
	Regular Express
☐	SQL (7)
	♦ 15 Database d
	♦ 14+ SQL interv
	♦ 9 SQL scenari
	Auditing databas

	Deleting records
	SQL Subquery in
	Transaction man
☐	UML (1)
	◆ 12 UML intervi
☐	JSON (2)
	JSON interview (
	JSON, Jackson,
☐	XML (2)
	XML basics inter
	XML Processing
☐	XSD (2)
	11 FAQ XSD inte
	XSD reuse inter
☐	YAML (2)
	YAML with Java
	YAML with Sprin
☐	Hadoop & BigData Int
	♥ 01: Q1 – Q6 Had
	02: Q7 – Q15 Hadc
	03: Q16 – Q25 Hac
	04: Q27 – Q36 Apa
	05: Q37 – Q50 Apa
	05: Q37-Q41 – Dat
	06: Q51 – Q61 HB
	07: Q62 – Q70 HDI
☐	Java Architecture Inte
	♥♦ 01: 30+ Writing
	001A: ♦ 7+ Java int
	001B: ♦ Java archi
	01: ♥♦ 40+ Java W
	02: ♥♦ 13 Tips to w
	03: ♦ What should l
	04: ♦ How to go ab
	05: ETL architectur
	1. Asynchronous pi
	2. Asynchronous pi
☐	Scala Interview Q&As
	01: ♥ Q1 – Q6 Scal
	02: Q6 – Q12 Scal
	03: Q13 – Q18 Sca

04: Q19 – Q26 Sca
05: Q27 – Q32 Sca
06: Q33 – Q40 Sca
07: Q41 – Q48 Sca
08: Q49 – Q58 Sca
09: Q59 – Q65 Hig
10: Q66 – Q70 Pat
11: Q71 – Q77 – S
12: Q78 – Q80 Rec
Spring, Hibernate, & I
Spring (18)
Spring boot (4)
♦ 11 Spring bc
01: Simple Sp
02: Simple Sp
03: Spring box
Spring IO (1)
Spring IO tuto
Spring JavaConl
10: Spring, Ja
Spring, JavaC
Spring, JavaC
Spring, JavaC
01: ♥♦ 13 Spring
01b: ♦ 13 Spring
02: ► Spring DI
03: ♥♦ Spring DI
04 ♦ 17 Spring b
05: ♦ 9 Spring B
06: ♥ Debugging
07: Debugging S
Spring loading p
Hibernate (13)
01: ♥♦ 15+ Hiber
01b: ♦ 15+ Hiber
02: Understandir
03: Identifying ar
04: Identifying ar
05: Debugging H
06: Hibernate Fil
07: Hibernate mi

	08: Hibernate au
	09: Hibernate en
	10: Spring, Java
	11: Hibernate de
	12: Hibernate cu
	AngularJS (2)
	♥ 8 AngularJS in
	More Angular JS
	Git & SVN (6)
	♥ Git & Maven fc
	♥ Merging Vs rel
	♥ Understanding
	6 more Git interv
	8 Git Source cor
	Setting up Cygw
	JMeter (2)
	♥ JMeter for test
	♦ JMeter perform
	JSF (2)
	JSF interview Q&
	More JSF intervi
	Maven (3)
	♥ Git & Maven fc
	12 Maven intervi
	7 More Maven ir
	Testing & Profiling/Sa
	Automation Testing
	♥ Selenium and
	Code Coverage (2)
	Jacoco for unit te
	Maven and Cobr
	Code Quality (2)
	♥ 30+ Java Code
	♦ Ensuring code
	jvisualvm profiling (
	01: jvisualvm to :
	02: jvisualvm to :
	03: jvisualvm to :
	Performance Testir
	♥ JMeter for test
	♦ JMeter perform

Unit Testing Q&A (2)

BDD Testing (4)

Java BDD (Be

jBehave and E

jBehave and j

jBehave with t

Data Access Uni

♥ Unit Testing

Part #3: JPA H

Unit Test Hibe

Unit Test Hibe

JUnit Mockito Sp

JUnit Mockito

Spring Con

Unit Testing

Part 1: Unit te

Part 2: Mockit

Part 3: Mockit

Part 4: Mockit

Part 5: Mockit

Testing Spring T.

Integration Un

Unit testing Sp

♦ 5 Java unit tes

JUnit with Hamc

Spring Boot in ui

Other Interview Q&A 1

Finance Domain In

12+ FX or Forex

15 Banking & fin

FIX Interview Q&A

20+ FIX basics in

Finding your way

Groovy Interview Q

Groovy Coding C

Cash balance

Sum grades C

♥ Q1 – Q5 Groov

♦ 20 Groovy clos

♦ 9 Groovy meta

Groovy method c

- Q6 – Q10 Groov
- JavaScript Interview
- JavaScript Top I
- ♥ Q1 – Q10 J
- ♦ Q11 – Q20
- ♦ Q21 – Q30
- ♦ Q31 – Q37
- JavaScript mis
- JavaScript Vs Ja
- JavaScript Vs
- Unix Interview Q&A
- ♥ 14 Unix intervi
- ♥ Hidden Unix, C
- sed and awk to v
- Shell script inter
- Unix history com
- Unix remoting in
- Unix Sed comm
- Free Java Interview
- Java Integration
- Java Beginner I
- 02: Spring DIP, I
- 06: Tell me abou

As a Java Architect

[Java architecture & design concepts interview Q&As with diagrams | What should be a typical Java EE architecture?](#)

Senior Java developers must have a good handle on

[open all](#) | [close all](#)

- [Best Practice \(6\)](#)
- [Coding \(26\)](#)
- [Concurrency \(6\)](#)
- [Design Concepts \(7\)](#)
- [Design Patterns \(11\)](#)
- [Exception Handling \(3\)](#)
- [Java Debugging \(21\)](#)
- [Judging Experience In](#)
- [Low Latency \(7\)](#)
- [Memory Management](#)
- [Performance \(13\)](#)
- [QoS \(8\)](#)
- [Scalability \(4\)](#)
- [SDLC \(6\)](#)
- [Security \(13\)](#)
- [Transaction Managen](#)

80+ step by step Java Tutorials

[open all](#) | [close all](#)

- [Setting up Tutorial \(6\)](#)
- [Tutorial - Diagnosis \(2\)](#)
- [Akka Tutorial \(9\)](#)
- [Core Java Tutorials \(2\)](#)
- [Hadoop & Spark Tuto](#)
- [JEE Tutorials \(19\)](#)
- [Scala Tutorials \(1\)](#)
- [Spring & Hibernate Ti](#)
- [Tools Tutorials \(19\)](#)
- [Other Tutorials \(45\)](#)

Preparing for Java written & coding tests

open all | close all

- ✚ ♦ Complete the given
- ✚ Can you write code? |
- ✚ Converting from A to I
- ✚ Designing your classe
- ✚ Java Data Structures
- ✚ Passing the unit tests
- ✚ What is wrong with th
- ✚ Writing Code Home A
- ✚ Written Test Core Jav
- ✚ Written Test JEE (1)

How good are your...to go places?

open all | close all

- ✚ Career Making Know-
- ✚ Job Hunting & Resum

Empowers you to open more doors, and fast-track

Technical Know Hows

☀ [Java generics in no time](#) ☀ [Top 6 tips to transforming your thinking from OOP to FP](#) ☀ [How does a HashMap internally work? What is a hashing function?](#)
 ☀ [10+ Java String class interview Q&As](#) ☀ [Java auto un/boxing benefits & caveats](#) ☀ [Top 11 slacknesses that can come back and bite you as an experienced Java developer or architect](#)

Non-Technical Know Hows

☀ [6 Aspects that can motivate you to fast-track your career & go places](#) ☀ [Are you reinventing yourself as a Java developer?](#) ☀ [8 tips to safeguard your Java career against offshoring](#) ☀ [My top 5 career mistakes](#)

Prepare to succeed

☀ [Turn readers of your Java CV go from “Blah blah” to “Wow”?](#) ☀ [How to prepare for Java job interviews?](#) ☀ [16 Technical Key Areas](#) ☀ [How to choose from multiple Java job offers?](#)

Select Category ▼

© Disclaimer

The contents in this Java-Success are copy righted. The author has the right to correct or enhance the current content without any prior notice.

These are general advice only, and one needs to take his/her own circumstances into consideration. The author will not be held liable for any damages caused or alleged to be caused either directly or indirectly by these materials and resources. Any trademarked names or labels used in this blog remain the property of their respective trademark owners. No guarantees are made regarding the accuracy or usefulness of content, though I do make an effort to be accurate. Links to external sites do not imply endorsement of the linked-to sites.