











package

time

package

Chron

o

package

field

package

bas

e

class

Instan

t

class

Chronolog

y

class

Datetime

class

Duration

class

DateTime

Fiel

d

class

DateTime

FieldType

class

DateTime

Zone

class

ReadableI

nstant

class

FieldUtil

s

class

BaseDuration

class

BaseDateTime

class

AbstractInstan

t

class

IsoChronology

class

GIChronolog

y

class

GregorianChro

nology

class

JulianChronolog

y

Exercise

1.1

Exercise 1.1

For the map of the architecture, we decided to look at the imports that each class makes. We started at the package joda time, because from the documentation we have inferred, that here are the most classes defined, the user interacts with and it is in general the most important package.

We also looked at the package base, because there are many of the important backend classes of the library definded. Many connections inside the time package are made through classes in the package base. This is another reason we decided to include this package.

And we looked at the package chrono. There the calendar system is implemented, with the IsoChronology as default calendar but also other calendars and the possibility to plug in own calendar system. We found these features very important since it determines in what sizes we can talk about time, week, day or hour for example. So we decided to include this package as well.

The last package we decided to include in our analysis is the package field. This package is used for displaying the date and time information in different formats. For example: The DayOfYear Method shows the days passed at that year.

Many of the important User interaction is done via this package. Therefore, we decided to also include this package.

From the documentation, we also inferred which classes are the most important ones in the package joda.time. Otherwise, because there are so many classes in this package and some of them are significantly less important then others. Then we looked, which classes are importet from joda.time made to the other selected packages: base, chrono, field. And we also regarded the connections from these other connections back to joda.time. This way we could figure out the connections between important members within joda.time going via these other packages, most importantly base, for this purpose.

class

Datetim

e

now

()

joda.time

joda.time.base

joda.time.chron

o

class

Datetime

Datetime

(

)

class

Datetime

super(

)

class

BaseDateTime

BaseDateTime

(

)

class

DateTimeUtils

currentTimeMilis

(

)

class

MilllisProvider

getMillis

(

)

class

SystemMilisProvider

System.currentTimeMillis

(

)

class

ISOChronology

getInstance

()

class

DateTimeUtils

getDefault

()

Exercise

1.

2