

Datumsformattierung mit DateFormat

Dr. Andreas Schroeder



Dieses Video behandelt

- Datumsobjekte und Locales
- Datum formatieren und einlesen mit DateFormat
- Datum formatieren und einlesen mit SimpleDateFormat



- `long System.currentTimeMillis()`
 - Liefert Anzahl der Millisekunden seit 1. Januar 1970
- `java.util.Date`
 - Stellt ein Zeitpunkt dar
 - Bietet Millisekunden-Genauigkeit
 - Keine Berechnungen möglich – dafür `Calendar` verwenden



- Datumsformatierung durch Locales
 - Trennsymbole
 - Reihenfolge von Tag/Monat/Jahr
- Ausführlichkeit durch DateFormat Styles
 - SHORT – kurz und knapp
 - MEDIUM – mittlerer Umfang
 - LONG – sehr ausführlich
 - FULL – vollständig





Schritte für Formatierung und Einlesen

- DateFormat erzeugen mit
 - getTimeInstance – nur Uhrzeit verarbeiten
 - getDateInstance – nur Datum verarbeiten
 - getDateTimelInstance – Datum und Uhrzeit verarbeiten

Dabei den gewünschten Stil und Locale festlegen

- Date zu String konvertieren mit `format(date)`
- String zu Date konvertieren mit `parse(string)`
 - `parse` wirft `ParseException`, falls das Einlesen misslingt



Schritte für Formatierung und Einlesen

- SimpleDateFormat erzeugen mit pattern

```
new SimpleDateFormat("yyyy-MM-dd'T'HH:mm:ss.SSSXXX");
```

- Date zu String konvertieren mit `format(date)`
- String zu Date konvertieren mit `parse(string)`
 - `parse` wirft `ParseException`, falls das Einlesen misslingt



Was in diesem Video gezeigt wurde

- Datumsobjekte und Locales
- Datum formatieren und einlesen mit DateFormat
- Datum formatieren und einlesen mit SimpleDateFormat