

# Täglich grüßt das Murmeltier

Aber um wieviel Uhr?

Jan Müller

## 2 Zeitkonzepte

### Datum & Uhrzeit

- Tupel (Jahr, Monat, Tag, Stunde, ...)
- 15.6.2020 18:14:08
- Vormittags, abends, Mitternacht, ...



`LocalDate(Time)`

### Zeitpunkt

- Global, überall auf der Welt gleich
- „jetzt“, „gepostet vor 10 Minuten“
- Epoch Millis, Unix Timestamp

`Instant, System.currentTimeMillis()`



- Zeitzonen
- Winter- und Sommerzeit
- UTC, GMT, CET, CEST, UTC+4, Europe/Berlin, ...



# Empfehlungen

Soll ich jetzt LocalDateTime, ZonedDateTime, Instant oder Date verwenden???

- Date bitte nicht
- Ansonsten: Kommt drauf an was ich machen will
- (Gewagte) These: Außer in fachlichen Sonderfällen braucht man
  - im Server eher Zeitpunkte (Instants oder einfach Unix Timestamps)
  - im Client die Umrechnung in ein Datum und eine Uhrzeit zur Anzeige
- Wenn ich im Code UTC als Zeitsystem verwende: Ist das wirklich ein fachlich motiviertes Zeitsystem? Vermutlich verwende ich besser einfach ein Instant
- Wenn ich bewusst Zeitsysteme verwende: Achtung bei CET, CEST usw., das sind keine Zeitzonen und werden nicht einheitlich verwendet!



## Wo lag Phileas Foggs Bug?

Er zählte die Tage anhand von

```
LocalDateTime.atZone(ZoneId.systemDefault())
```

Allerdings änderte sich `ZoneId.systemDefault()` ständig während seiner Reise. Richtig wäre:

```
LocalDateTime.atZone(ZoneId.of("Europe/London"))
```

Oder einfach berechnen:

```
Duration.between(start, Instant.now()).toDays()
```

