

# Datum, OOP

Avancerad Javaprogrammering

Utbildare: Mikael Olsson

# LocalTime

- Representerar en tidpunkt
- Innehåller
  - Timmar
  - Minuter
  - Sekunder
- Skapas genom att anropa `now()` eller med `of(h, min, sek)` eller `parse(string)`:
  - `LocalTime t = LocalTime.now(); //tiden just nu`
  - `LocalTime t = LocalTime.of(13,15,0);`
  - `LocalTime t = LocalTime.parse("13:56:24");`
- Format vid utskrift bestäms av lokalitet

# LocalDate

- Representerar ett datum
- Innehåller
  - År
  - Månad
  - Dag
- Skapas upp genom att anropa `now()`, `of(år, månad, dag)` eller `parse(string)`:
  - `LocalDate t = LocalDate.now();`
  - `LocalDate t = LocalDate.of(2015, 12, 24);`
  - `LocalDate t = LocalDate.parse("2015-12-24");`
- Format vid utskrift bestäms av lokalitet

# LocalDateTime

- Representerar ett datum och ett klockslag
- Format vid utskrift bestäms av lokalitet
- `LocalDateTime today = LocalDateTime.now();`
- `today = LocalDateTime.of(LocalDate.now(),  
LocalTime.now());`
- `LocalDateTime specificDate = LocalDateTime.of(2014,  
Month.JANUARY, 1, 10, 10, 30);`

# Formatering av datum

- Klassen `DateTimeFormatter`
- Metoden `ofPattern` tar ett mönster
- För att skriva ut en tidpunkt `t` på formatet `HH:mm:ss`
  - `t.format(DateTimeFormatter.ofPattern("HH:mm:ss")) ;`
- För att skriva ut ett datum `d` på formatet `dd/mm YYYY`
  - `d.format(DateTimeFormatter.ofPattern("dd/mm YYYY")) ;`

# TimeZone

- Representerar en tidszon
- Ger din aktuella tidszon:
  - `TimeZone.getDefault()`
- Du kan sätta tidszon med
  - `TimeZone.setDefault(tidszon) ;`

# Övningsuppgift 9

- Skriv ett program som tar namnet på en (förutbestämd) stad och skriver ut vad klockan är i den staden just nu på formatet HH:mm:ss. Implementera detta för åtminstone Toronto, Stockholm och Shanghai (fler städer om du vill). Använd klassen TimeZone. Googla för att hitta namnen på tidszonerna eller deras id:n.
- Programmet ska funka både då användaren skriver stadsnamnet med stora, små, eller blandade bokstäver. Det ska funka även om användaren lägger in blankslag efter stadens namn.
- Om en användare skriver in en stad som ditt program inte kan visa tid för, visa upp ett lämpligt felmeddelande.

# Instant

- Representerar en tidpunkt
- Antal nanosekunder från 1 januari 1970
- För att få nuvarande tid:
  - `Instant inst = Instant.now();`
- Ersätter den gamla klassen Date som fungerade på liknande sätt, men visade antalet millisekunder från 1 jan 1970.



# Duration

- Representerar ett tidsintervall
- Har metoder som omvandlar intervall-längden mellan olika tidsenheter

Demo instant och duration

# Övningsuppgift 10

- Förr när man fortfarande spelade in film på videoband var det ibland svårt att veta om ett kommande tv-program skulle få plats på bandet.
- Skriv ett program som hjälper oss räkna ut om ett band kommer att räcka eller inte.
- Programmet ska fråga efter videobandets längd (i minuter), hittills använd tid av bandet, den tidpunkt då tv-programmet börjar och den tidpunkt då det slutar (anges som klockslag, tt:mm).
- Använd klasserna `LocalTime` och `Duration` för att räkna ut tiden. Jobba testdrivet och skriv testerna innan du skriver koden. Om användaren skriver in något konstigt, visa ett felmeddelande.