



# Yrgo-2016 Designmönster

Introduktion till kursen



# Omfattning

Kursen är på 10 poäng och ges under veckorna 45-48 under tisdagar och torsdagar (tis 8/11 - torsd 1/12)

Varje tillfälle är schemalagt till 3,5 timmar.

Förmiddag 8.30-12.00 tisdagar, eftermiddag 13.00-16.30 torsdagar.

Sista veckan är det endast torsdag eftermiddag!

Totalt är det 7x3,5h i klassrum

# Kursplan

Designmönster har till uppgift att lösa problem som liknar varandra på samma sätt som man har löst dem tidigare. Det betyder att man kan ägna sig åt att lösa unika problem och använda kända metoder för kända problem. Detta innebär att man kan hålla kostnader nere för att det ger en kodstruktur som alla kan sätta sig in i. Vi tar under kursen upp några av de vanligaste designmönstren för att den studerande skall förstå principen bakom tankesättet. identifiera och beskriva ett tiotal av de vanligaste designmönstren, samt förklara när och hur de kan användas för att lösa vanliga designproblem

det objektorienterade programmeringsparadigmet

kategorier av designmönster: t.ex. skapande, strukturella och beteenderelaterade

vanliga designmönster ur olika kategorier i teori och praktik

implementation och tillämpning av olika designmönster i Java.

## Mål

Den studerande skall efter genomgången kurs kunna:

välja och motivera ett designmönster för ett givet designproblem anpassa designmönster för aktuella tillämpningar implementera behandlade designmönster i Java.

## Undervisning

Undervisningen genomförs i form av:

Lektioner, Övningar och Självstudier

## Kunskapskontroll

Kunskapskontroll sker genom prov och inlämningsuppgifter.

## Betygskriterier:

Godkänd

Den studerande kan använda och förstå designmönster med visst stöd

Den studerande kan skriva kod med hjälp av designmönster.

Den studerande arbetar tidseffektivt med designmönster.

## Väl godkänd

Utöver kriterierna för Godkänd ska följande kriterier uppfyllas.

Den studerande tillämpar med säkerhet designmönster i sin kod.

# Innehåll

- Vad är design patterns
- Vad är OO-principer
- Exempel på mönster
  - Builder
  - Singleton
  - Exception and Tiers (fault barrier)
  - (Abstract)Factory
  - Decorator
  - Strategy
  - Observer

# Undervisningsformer

- Ni förväntas läsa en hel del!
- Teori blandat med övningar
  - Det är ganska tidskrävande med realistiska och relevanta övningar!
  - Kan bli lite förenklade övningar

# Examination

Skriftligt prov på teorin - kunna beskriva några patterns:

- syfte (vilket problem de löser)
- effekt (vilka OO- och andra principer de stödjer)
- implementera (givet ett mönster, ska ni skriva om ett program till att använda ett mönster)

# Grovplanering

- Tis 2016-11-08 Vad är designpatterns (teori)
- Tor 2016-11-10 OO-principer (teori)
- Tis 2016-11-15 Our first patterns - Builder + Singleton (teori+övn)
- Tor 2016-11-17 Using exceptions to protect tiers/reduce coupling (teori+övn)
- Tis 2016-11-22 Factory/AbstractFactory (teori+övn)
- Tor 2016-11-24 Decorator (and composition) (teori+övn)
- Tor 2016-12-01 Strategy (and Observer) (teori+övn)
- Tenta: To be decided

# Recommenderad litteratur

- Head first design patterns:
  - [CDON](#) 362:-
  - [Pricerunner](#)
  - [Bokus](#) 362:-
  - [Adlibris](#) 362:-
- Design Patterns - Elements of reusable OOS
  - Köp en ny utgåva med Java-exempel! (lite torrare och äldre bok än ovanstående)
- Tutorial
  - <http://www.journaldev.com/1827/java-design-patterns-example-tutorial> 0:-
  - [https://sourcemaking.com/design\\_patterns](https://sourcemaking.com/design_patterns) 0:-