

Erhvervsakademi MidtVest

Programming Plan

1. semester 2018

Hans Iversen
24-08-2018

Indholdsfortegnelse

Programmering - Overblik	2
Lekturer:.....	2
Diverse artikler, Links, mm.:	2
Læringsmål fra Studieordningen	3
Viden (programmering)	3
Færdigheder (programmering).....	3
Kompetencer (programmering)	3
Plan:.....	4
Uge 1 (35): Java kap. 1-2.....	4
Læringsmål:	4
Uge 2 (36): Java kap. 3-4, Clean Code kap. 1	4
Læringsmål:	4
Uge 3 (37): Java kap. 5-6, Clean Code kap. 2	4
Læringsmål:	4
Uge 4 (38): Java kap. 7-9, Clean Code kap. 3.....	4
Læringsmål:	4
Uge 5 (39): Java kap. 9, Tutorial Collections, Clean Code kap. 4-5.....	4
Læringsmål:	5
Uge 6 (40): Java 16-17 (kap. 16 skimmes), Clean Code kap. 6-7	5
Læringsmål:	5
Uge 7 (41): Unit-test. JUnit, Clean Code kap. 8-10	5
Læringsmål:	5
Uge 8 (43): Opsamling	5
Læringsmål:	5
Uge 9 (44): Java kap. 10.....	5
Læringsmål:	5
Uge 10 (45): Java kap. 11 (skimmes)-12	6
Læringsmål:	6
Uge 11 (46): Java 13 (skimmes)-14.....	6
Læringsmål:	6

Programmering - Overblik

Lecture:

[Studieordning, boglister, mm.](#)

Herbert Schildt / Java – A Beginner's Guide – Seventh Edition / McGraw-Hill / 978-1259589317

Robert C. Martin / Clean Code / Prentice Hall/ 978-0132350884

[Java Tutorial](#)

[Java API](#)

WiKipedia

[Test-driven development](#)

[Bits and Bytes](#)

[Junit](#)

Diverse artikler, Links, mm.:

<http://www.information.dk/501493>

<http://www.information.dk/516588>

Læringsmål fra Studieordningen

1. semester – påbegyndes før efterårsferien (HI)

1. semester – påbegyndes efter efterårsferien (FKJ)

2. semester – påbegyndes (eller med hovedvægt) efter jul

Viden (programmering)

Den studerende har viden om

1. specifikation af abstrakte datatyper
2. kriterier for programkvalitet
3. abstraktionsmekanismer i moderne programmeringssprog

Færdigheder (programmering)

Den studerende kan

1. specificere og konstruere algoritmer
2. anvende programmeringssproget til realisering af algoritmer, design mønstre, abstrakte datatyper, datastrukturer, designmodeller og brugergrænseflader
3. anvende et moderne integreret udviklingsværktøj, herunder versionsstyringssystem
4. realisere modeller i et databasesystem og konstruere programmer, der benytter en databasegrænseflade
5. designe og konstruere programmer som samarbejdende processer/tråde
6. udvikle applikationer baseret på en lagdelt softwarearkitektur
7. anvende softwarekomponenter/biblioteker
8. udfærdige dokumentation i forhold til gældende de-facto standarder i professionen
9. anvende moderne teknikker og værktøjer til afvikling af test og kvalitetssikring
10. vurdere kvalitative og kvantitative egenskaber ved algoritmer og datastrukturer

Kompetencer (programmering)

Den studerende kan

1. indgå som en professionel programmør i udviklings- og vedligeholdelsesprojekter
2. tilegne sig nye færdigheder inden for programmeringssprog, udviklingsværktøjer, programmeringsteknikker og programdesign

Plan:

Uge 1 (35): Java kap. 1-2

- Præsentation af faget og plan for semestret. Introduktion til opgaver og arbejdsform.
- Computer-opsætning, JDK og Eclipse-installation ifølge HowTo-dokumentet.
- Java kap. 1-2 eksempler og øvelser.

Læringsmål:

- I gang med brugen af Eclipse (grundlæggende)
- Fornemmelse for Java sprogets syntaks (grundlæggende)
- F3, K2

Uge 2 (36): Java kap. 3-4, Clean Code kap. 1

- Opsamling uge 1.
- Java kap. 3-4 eksempler og øvelser.

Læringsmål:

- Fortrolig med Java sprogets syntaks (grundlæggende)
- Kan anvende Javas simple typer (grundlæggende)
- Kan anvende Java control statements (if, switch, for, while, do-while)
- Kan konstruere Klasser og metoder og arbejde med objekter (grundlæggende)
- Kan forstå begrebet Object-Oriented Programming
- V1, V2, V3, F1, F2

Uge 3 (37): Java kap. 5-6, Clean Code kap. 2

- Opsamling uge 2.
- Java kap. 5-6 eksempler og øvelser (Kun én-dimensionelle arrays).
- Opgave: Container

Læringsmål:

- Kan anvende én-dimensionelle Arrays
- Kan anvende enhanced for-loops og ?-operator
- Kan anvende access-modifiers (public, private, ...)
- Kan konstruere og anvende metoder med rekursion.

Uge 4 (38): Java kap. 7-9, Clean Code kap. 3

- Opsamling uge 3.
- Java kap. 7-9 eksempler og øvelser, selv-tests.
- Opgave: Lommeregner: Funktionalitet? Struktur? Grænseflader?

Læringsmål:

- Kan anvende Arv
- Kan oprette og placere klasser i Pakker
- Kan anvende Interfaces
- Kan konstruere og anvende Exceptions
- F7

Uge 5 (39): Java kap. 9, Tutorial Collections, Clean Code kap. 4-5

(Introduction, Collection, List, Set, Map, ArrayList, HashSet, HashMap)

- Opsamling uge 4.
- Java kap. 9 eksempler, opgaver og selv-tests.
- Collection, List, Set, (relation til opgave?), ArrayList, HashSet, HaspMap.
- Opgave: Lommeregner

Læringsmål:

- Kan forstå og anvende Collections
- Kan anvende List, Set, Map
- Kan anvende ArrayList, HashSet, HashMap

Uge 6 (40): Java 16-17 (kap. 16 skimmes), Clean Code kap. 6-7

- Opsamling uge 5.
- Java kap. 16 overblik
- Java Kap. 17 eksempler, øvelser og selv-tests.
- Opgave: Lommeregner, Specielt fokus på brugergrænseflader.

Læringsmål:

- Kan udarbejde simple brugerflader med anvendelse af JavaFX
- F6

Uge 7 (41): Unit-test. JUnit, Clean Code kap. 8-10

- Opsamling uge 6
- TDD (Test Driven Development), JUnit
- Opgave: Lommeregner
- Evaluering, Videre forløb.

Læringsmål:

- Forstår begrebet Test Driven Development
- Kan udarbejde simple tests med JUnit
- F9

Uge 8 (43): Opsamling

- Opsamling, repetition
- Opgave: Lommeregner
- Evaluering, Videre forløb.

Læringsmål:

- Har overblik over egen status i forhold til tidligere læringsmål

Uge 9 (44): Java kap. 10

- Opsamling uge 8.
- Java kap. 10 eksempler, øvelser og selv-tests
- Opgave: Lommeregner(GUI)

Læringsmål:

- Kan anvende Java I/O og Streams

Uge 10 (45): Java kap. 11 (skimmes)-12

- Opsamling uge 9.
- Java kap. 11 overblik
- Java Kap. 12 eksempler, øvelser og selv-tests, Enumerations
- Opgave: Banksystem

Læringsmål:

- Kan anvende Enumerations
- Kan anvende Java wrapper-klasser

Uge 11 (46): Java 13 (skimmes)-14

- Opsamling uge 10.
- Java kap. 13 overblik
- Java kap. 14 eksempler, øvelser og selv-tests, Lambdas i opgaven?
- Opgave: Banksystem

Læringsmål:

- Forstår begrebet Generics
- Forstår begrebet Functional Interfaces
- Kan anvende Java 8 Lambdas i simple sammenhænge