



NTNU

Kunnskap for en bedre verden

Øvingsforelesning 8 - Oppgaver

TDT4100 Objektorientert programmering

Onsdag 26.02.2020



Oppgaver

I denne øvingsforelesningen skal vi jobbe med grensesnitt. Vi kommer til å lage kode for å representere en gård og dyrene der.



Pakker

Utdelt kode og oppgavene finnes i mappen foreksempel/src/of8.

`foreksempel/src/of8.kode`

Her finner dere utdelt kode og kan skrive deres egen kode.

`foreksempel/src/of8.underveis`

Her blir kode lagt inn rett etter felles gjennomgang.

`foreksempel/src/of8.testar`

Her ligger kode som kan brukes til å teste at en oppgave er løst riktig

`foreksempel/src/of8.lf`

Blir gjort tilgjengelig i etterkant og inneholder et løsningsforslag for oppgavene.



Oppgave 1

 7 min

Motivasjon

Vi kommer til å ha flere forskjellige typer dyr, og ønsker å ha et felles sett med metoder tilgjengelig for dyrene.

Oppgavetekst

Lag et grensesnitt **Animal** som har tre metoder. **getName** og **getAge** som fungerer som getter-metoder, og **makeSound** som gir ut en tekststreng som representerer at dyret lager en lyd.

Testing

Ikke relevant

Oppgave 2

 7 min

Motivasjon

Vi har nå et grensesnitt for dyr, men ingen dyr. Vi ønsker derfor å lage et sett med dyr.

Oppgavetekst

Lag to klasser som implementerer **Animal** grensesnittet. Vi kommer til å lage klassene **Dog** og **Chicken** ved gjennomgang, men det er fritt frem for å lage noe annet.

Testing

Ikke veldig relevant, men hvis du ønsker kan du lage en **main**-metode hvor du tester at metodene dine gjør som de skal.

Oppgave 3

 7 min

Motivasjon

Vi har nå et lite sett med dyr, og kan begynne å lage en representasjon for en gård.

Oppgavetekst

Lag en klasse **Farm** som tar vare på et sett med dyr. Klassen skal ha to metoder **addAnimal** og **getAnimals** som respektivt legger til et dyr og gir ut en liste over alle dyrene på gården.

Testing

main-metoden i filen *oppgave3-main.txt* kan brukes til å teste at metodene fungerer ordentlig. Den kan måtte endres hvis dyrene fra oppgave 2 ikke er implementert på samme måte som under felles gjennomgang.



Oppgave 4

 7 min

Motivasjon

Vi har nå en gård med et sett med dyr, men vi har ingen måte å sortere dyr på.

Oppgavetekst

Lag en klasse **AnimalComparator** som implementerer **Comparator** grensesnittet. Denne skal sammenligne dyr basert på alderen deres.

Testing

main-metoden i filen *oppgave4-main.txt* kan brukes til å teste at sorteringen fungerer. Da skal lydene dyrene lager skrives ut i motsatt rekkefølge av den i oppgave 3.

Oppgave 5

 5 min

Motivasjon

Vi kan nå bruke metoden **getAnimals** til å hente ut dyr på en gård. Dette er tungvindt hvis vi skal hente ut et enkelt dyr eller hvis vi senere vil implementere en iterator for **Farm**.

Oppgavetekst

Implementer metodene **getAnimal** og **numberOfAnimals** som henholdsvis gir ut dyret lagret på en oppgitt indeks og hvor mange dyr det er på gården.

Testing

main-metoden i filen *oppgave5-main.txt* skal ved riktig implementasjon skrive ut det samme resultatet som i oppgave 3.

Oppgave 6

 7 min

Motivasjon

Vi ønsker å kunne iterere over alle dyrene i en gård.

Oppgavetekst

Lag en klasse **FarmIterator** som implementerer **Iterator** grensesnittet.

Testing

main-metoden i filen *oppgave6-main.txt* skal ved riktig implementasjon skrive ut det samme resultatet som i forrige oppgave.



Oppgave 7



5 min

Motivasjon

Det er litt tungvindt å bruke en **iterator** for å iterere over et objekt av en gitt klasse.

Oppgavetekst

Implementer grensesnittet **Iterable** i **Farm** klassen.

Testing

main-metoden i filen *oppgave7-main.txt* skal ved riktig implementasjon skrive ut det samme resultatet som i forrige oppgave.

Oppgave 8

 7 min

Motivasjon

Det kan være litt tungvindt å måtte opprette et **AnimalComparator** objekt hver gang vi ønsker å sortere dyr.

Oppgavetekst

La **Animal** grensesnittet utvide **Comparable** grensenittet. Implementer nødvendige endringer i **Dog** og **Chicken** klassene.

Testing

main-metoden i filen *oppgave8-main.txt* skal ved riktig implementasjon skrive ut lyden dyrene lager i motsatt rekkefølge av i forrige oppgave.