

Velkommen til gruppetime i IN1000



20. oktober 2021
Jessie Yue Guan

Planen for i dag

- Datastrukturtegning
- Jobbe med oppgaver
- Løsningsforslag

Objekter inni objekter

- La oss si at vi har fått et oppdrag i å lage et program som fungerer som et katteregister
- Du har fått informasjon om navn, alder, farger og kjønn til alle kattene i en tekstfil
- Din oppgave er å sjekke om det ved et uhell har blitt ført duplikater i den lista
- Du skal også kunne vise fram informasjon om alle kattene på en oversiktlig måte
- Hvilken datastruktur passer best til dette problemet? Hvilke klasser og filer burde vi ha?

Oppgave 1

- Dere skal få filen katt.py som inneholder klassen Katt
- Den har allerede en konstruktør som initialiserer følgende instansvariabler
 - Navn
 - Alder
 - Farger
 - Kjønn
- Lett oppgave: Dere skal lage de manglende metodene str og eq

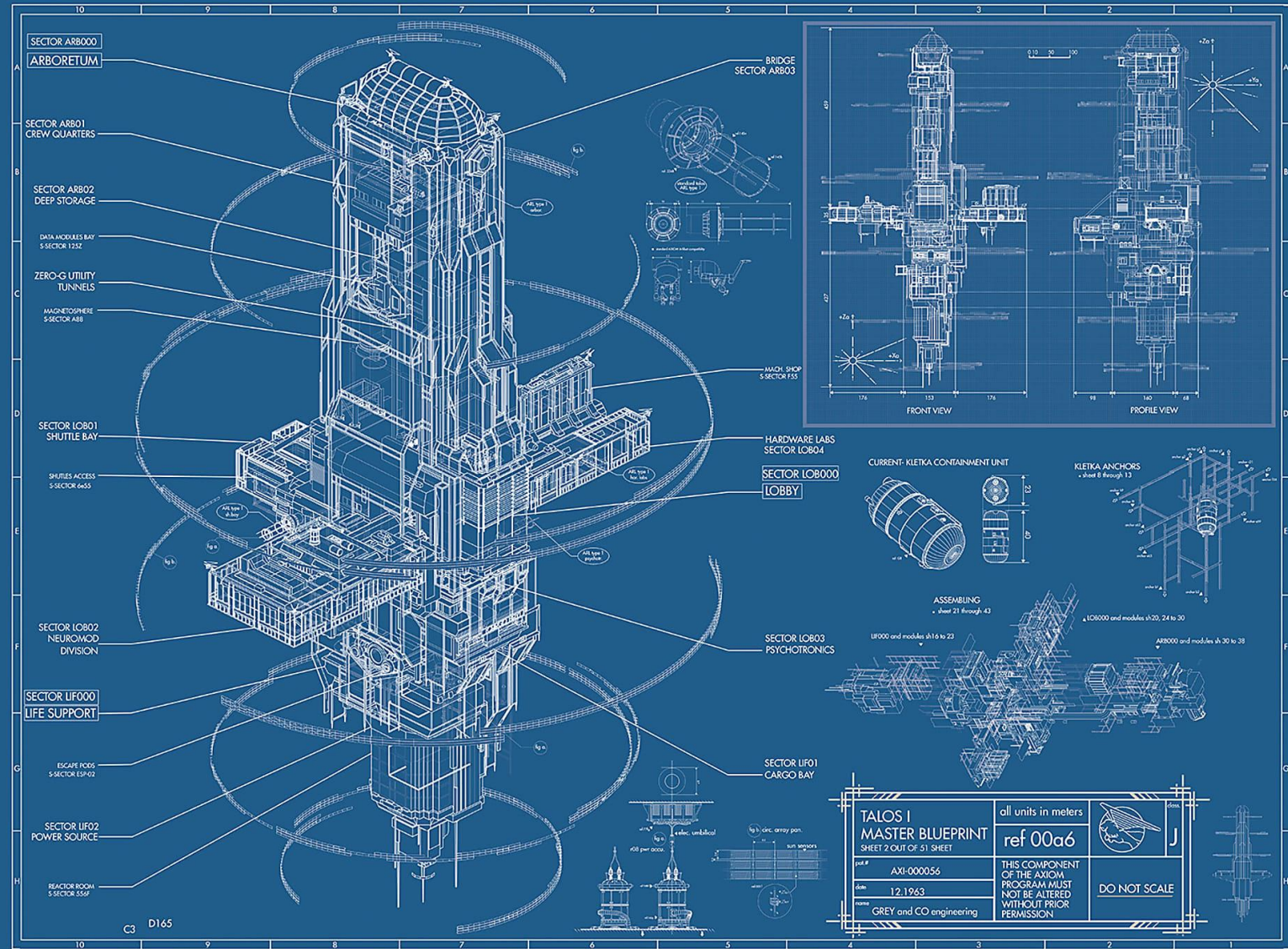
Oppgave 2

- Dere skal få filen katteregister.py som inneholder klassen Katteregister
- Den har allerede en konstruktør som initialiserer følgende instansvariabler
 - Katter
- Vanskelig oppgave: Dere skal lage de manglende metodene `les_fra_fil`, `finn_duplikater` og `vis_alle_katter`

Oppgave 3

- Dere skal få filen hovedprogram.py som skal lese inn fra fil, finne duplikater, og vise informasjon
- Den har allerede en prosedyre som er tom til å begynne med
 - Hovedprogram
- Lett oppgave: Dere skal fylle inn prosedyren, dvs. lage objekter og kjøre metoder

Datastruktur- tegning



Datastrukturtegning

- La oss si at vi har fått i oppgave å lage et helt nytt datasystem for helsevesenet
- Systemet skal bestå av flere programmer/filer/klasser som holder styr på blant annet:
 - Sykehus
 - Leger
 - Sykepleiere
 - Pasienter
 - Medikamenter
 - Resepter
 - Apoteker
- Hvor skal du i det hele tatt begynne? Hvordan skal du sørge for at de ulike komponentene hanger sammen, ikke overlapper, og ikke motsier hverandre?

Datastrukturtegning

- Før du begynner å kode store komplekse programmer kan det være lurt å lage en plan
- Hvilke instansvariabler skal de ulike klassene ha? Hvordan skal de initialiseres?
- Hvilke instansmetoder skal de ulike klassene ha? Hvordan skal grensesnittene se ut?
- Hvilke referanser skal vi ha mellom klassene? Hvordan skal de utveksle informasjon?
- En datastrukturtegning er på en måte et tankekart over programmet ditt
- Den lar deg planlegge, gir deg oversikt, og kan brukes som et utkast til programmet

Datastrukturtegning

- Variabler tegnes som firkanter
- Objekter tegnes som sirkler eller avrundede firkanter
- Referanser tegnes som piler
- Lister og ordbøker må tegnes som objekter
- Strenger, tall, og boolske verdier kan tegnes som variabler

Oppgave 4

- Lag en datastrukturtegning av oppgave 1, 2 & 3 

Løsningsforslag

- Datastrukturtegning av oppgave 1, 2 & 3 💡

Spørsmål?

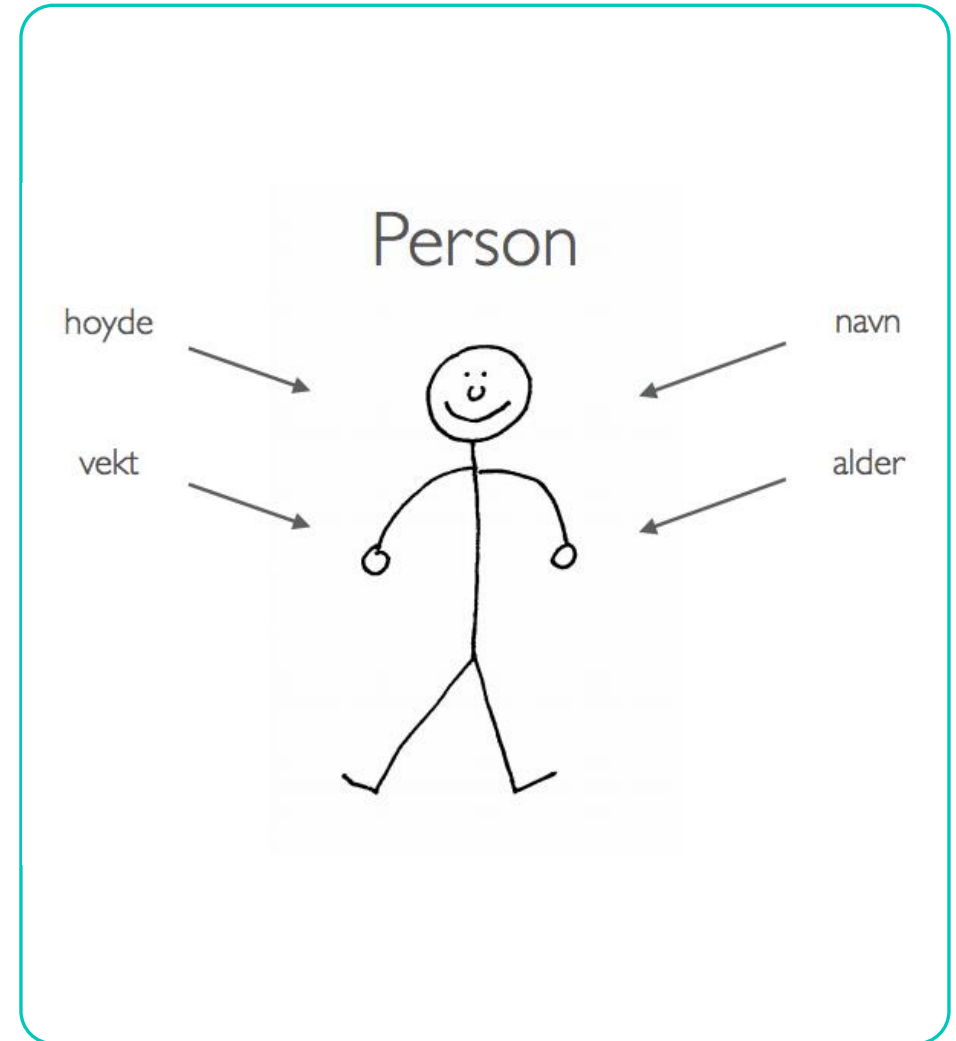
- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! 😊

Oppgave: Person



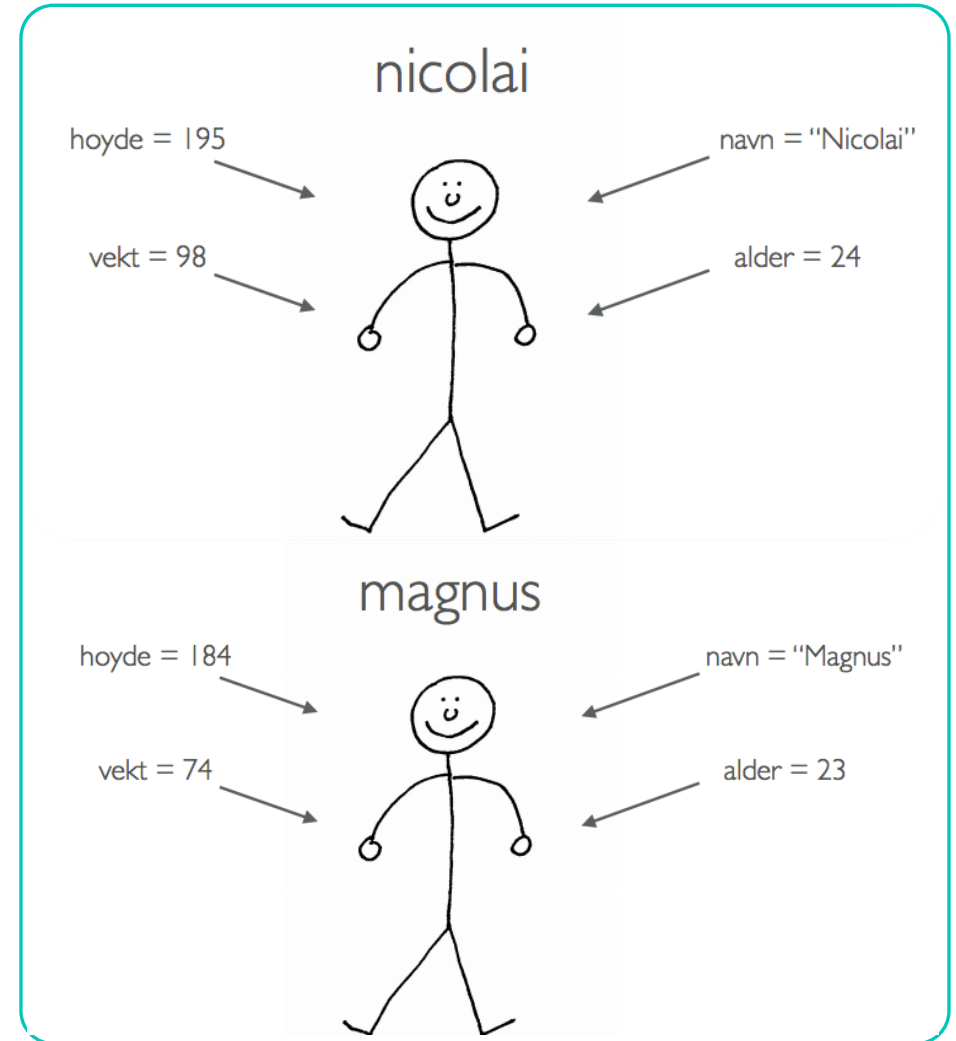
Person

- Her en modell av en person med de egenskapene vi er mest interessert i
- Alle egenskapene vi har lagt til personen er felles for alle personer
- Foreløpig er dette kun en generell mal og ikke en spesifikk person i seg selv



Person

- Når vi vil lage en instans av klassen Person følger vi bare malen
- Både «magnus» og «nikolai» er ulike objekter av samme klasse
- Vi er ikke bare opptatt av egenskaper, men også handlinger



Person

- Alle klasser har et grensesnitt eller et sett med metodeneavn, parametere, og returverdier...
- Vi vet kanskje ikke hva som ligger inni disse metodene eller hvordan de implementeres...
- Vi vet bare hvordan vi skal bruke dem



Oppgave 1

- Lag en klasse Person med en konstruktør som oppretter følgende instansvariabler...
 - navn – setter navnet på personen
 - alder – setter alderen på personen
 - vekt – setter vekten på personen
 - hoyde – setter høyden på personen

Oppgave 2

- Utvid klassen Person med følgende metoder...
 - oppgiNavn() – returnerer “Jeg heter <navn>”
 - erHoyereEnn(annen) – returner true hvis «self» er høyere enn «annen»
 - lagHilsen() – returnerer “Hei! Jeg heter <navn> og jeg er <alder> år gammel”
 - skrivUtInfo() – skriver ut infoen om personen

Oppgave 3

- Lag et program `person_test.py` hvor du...
 - Oppretter et person-objekt
 - Gir personen navnet "Emilie"
 - Skriver ut navnet til personen
 - Gir personen alderen 24
 - Skriver ut info om personen

Oppgave 4

- Utvid programmet `person_test.py` slik at du...
 - Oppretter fire person-objekter til
 - Bare finn på noen verdier, ikke bruk masse tid på dette, de trenger ikke å være realistiske
 - Lagre alle objektene i en liste
 - Bruk en løkke for å kalle på `lagHilsen()` på alle objektene
 - Lag en funksjon som tar inn en liste med personer og returnerer navnet til den høyeste personen

Løsningsforslag

- På tide med litt live-programmering! 

Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! 😊