Uke 3 - Arv og subklasser del 1

Gruppe 5 - jonasbny@ifi.uio.no

Agenda



Arv

Superklasser og subklasser

casting / typekonvertering

Object

Abstrakte klasser

Polymorfi

Ukas shoutout

Oppgaver

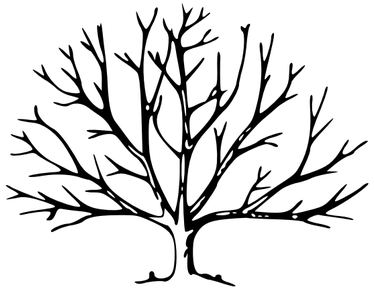
Arv

Et helt sentralt konsept innen objektorientering

En måte å opprette objekter som deler enkelte egenskaper

Kan gjøre at vi dupliserer mindre kode

Kan gi oss en mer intuitiv måte å modellere data på



Superklasser og subklasser

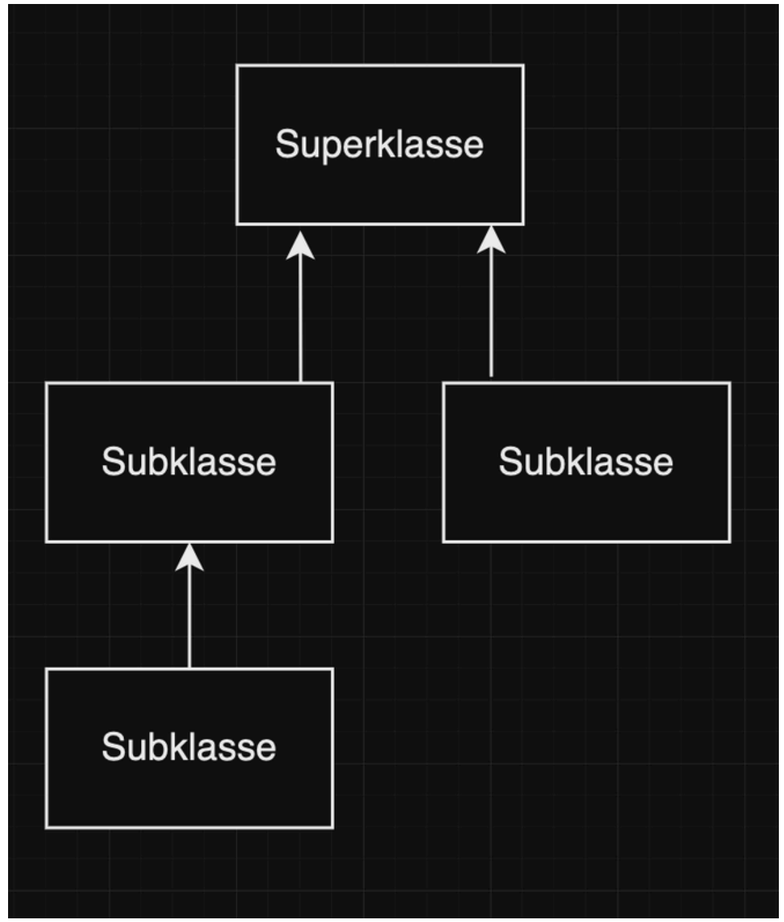
En superklasse er en klasses “forelder”

På samme måte er en subklasse en annen klasses “barn”

En superklasse defineres som en vanlig klasse

En subklasse defineres med nøkkelordet *extends*

Subklassen vil ha alle instansvariablene og metodene som superklassen har

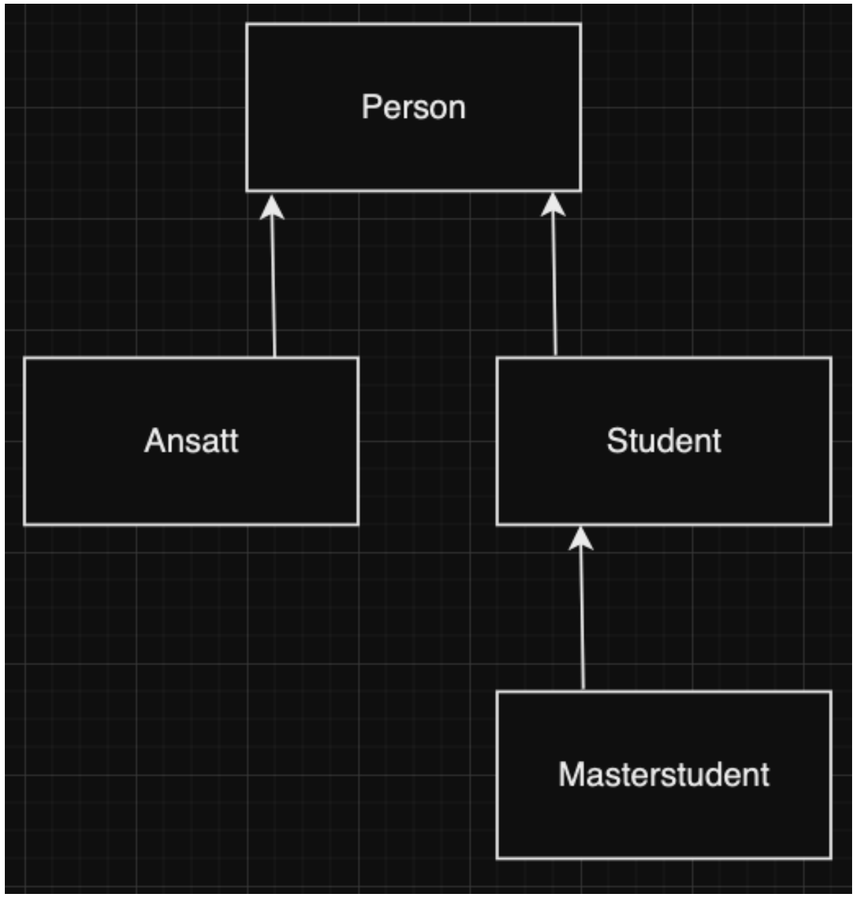


Klassehierarki

En annen type tegning enn datastrukturtegning

Viser klassenes “familietre”

Bokser med klassenavn, hvor klassene peker «opp» på klassene de arver fra



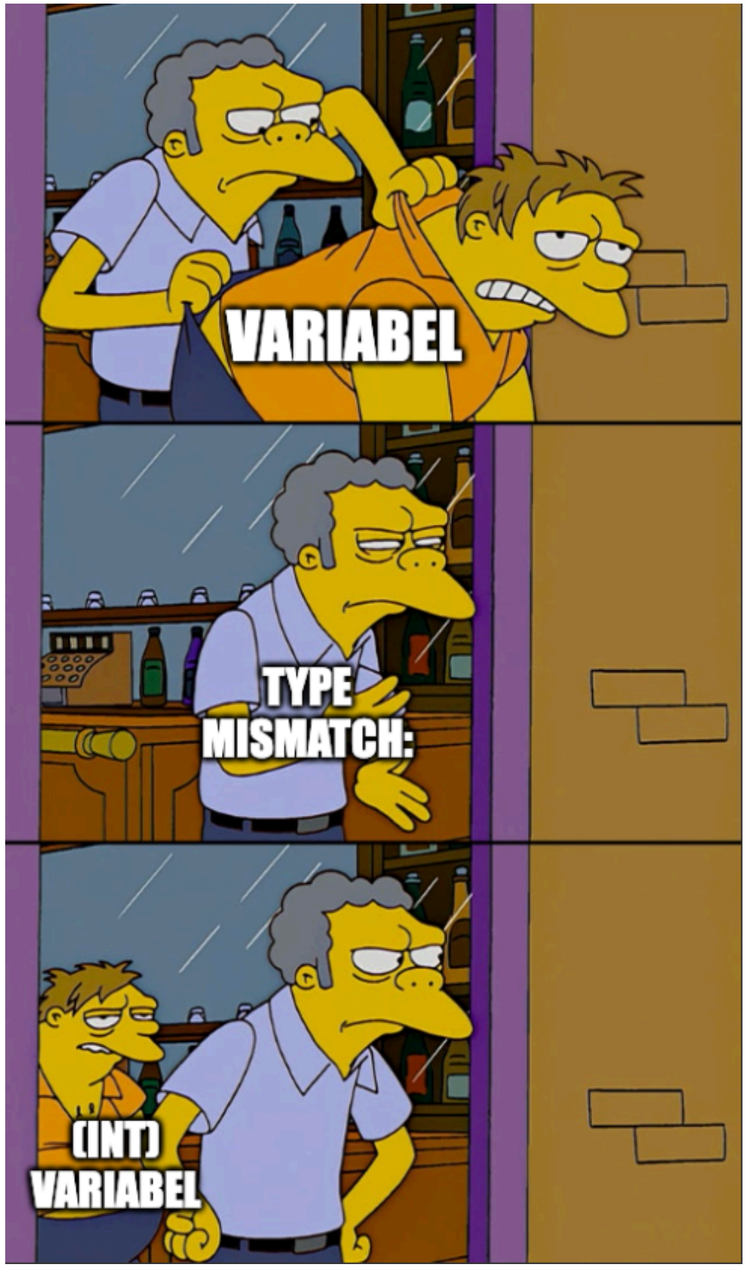
Casting / typekonvertering

Casting i Java brukes til å konvertere typen til et objekt eller en variabel til en annen type.

Noen ganger kan du ha behov for å konvertere en variabel fra en type til en annen fordi de ikke er direkte kompatible.

Konvertere en referanse av en underliggende klasse til en av dens superklasser for å få tilgang til spesifikke metoder eller egenskaper.

implisitt og eksplisitt, forskjellen



InstanceOf

Brukes for å sjekke om et objekt er av en gitt klasse

Undersøker egentlig om objektet inneholder “egenskapene” til det du tester mot



Oppgave

Blir dette å returnere true eller false?



```
return student instanceof Person;
```

Oppgave

Blir dette å returnere true eller false?



```
return person instanceof Student;
```

Oppgave

Hva er egentlig forskjellen her?



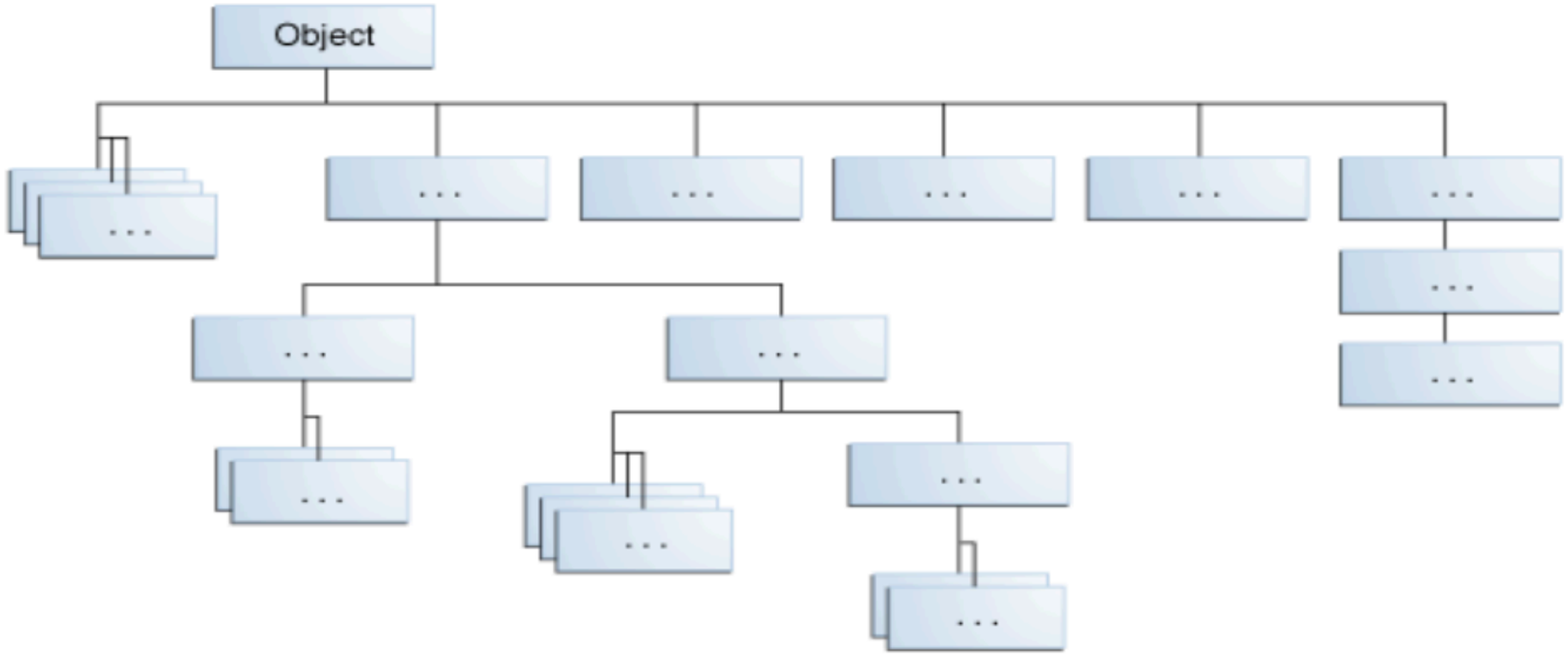
```
Person person1 = new Person();
Person person2 = new Student();
```

Object

Roten i Java sitt “familietre” er klassen Object

Object har et par innebygde metoder som alle (sub)klasser også kan bruke

- -toString()
- -equals()
- ...



Abstrakte klasser

En abstrakt klasse er en klasse man ikke kan opprette en instans av

Gir oss en slags oppskrift for hvordan subklassene skal se ut

Vi kan definere både instansvariabler og metoder i abstrakte klasser

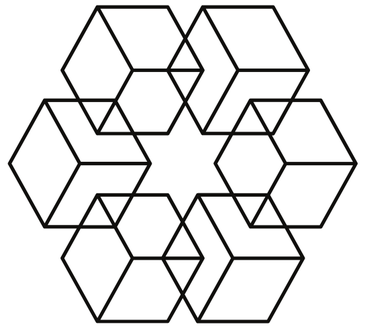


POLYMORFI

Handler om at metoder med samme signatur på tvers av superklasser og subklasser får ulik funksjonalitet

Redefinering av metoder med *@Override* i subklassene

Overriding og Overloading, forskjellen



Ukas shoutout

Alligator bites never heal



liveprogg

Oppgaver