

# 2.2 Metoder i klasshierarkier

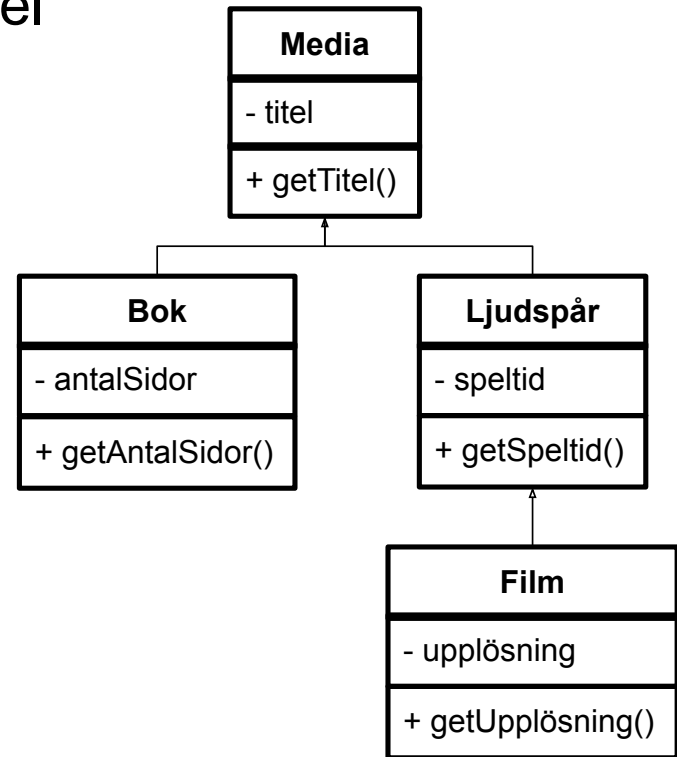
Holger Rosencrantz

## Ärvda metoder

```
class Fordon:
    def kör(self):
        print("Nu kör vi!")
class Bil(Fordon):
    def tuta(self):
        print("Tuuut!!")
b = Bil()
b.kör()      # anropar en Fordon-metod i ett Bil-objekt
b.tuta()     # anropar en Bil-metod
```

## Utökning av förra veckans exempel

- Privata attribut som sätts i respektive klass konstruktor
- Publika getter-metoder som ärvs



## Overriding (överskuggning)

```
class Superclass:
    def hej(self):
        print("Hej från Super!")
class Subclass(Superclass):
    def hej(self):
        print("Hej från Sub!")
sup = Superclass()
sup.hej()    # Hej från Super!
sub = Subclass()
sub.hej()    # Hej från Sub!
```

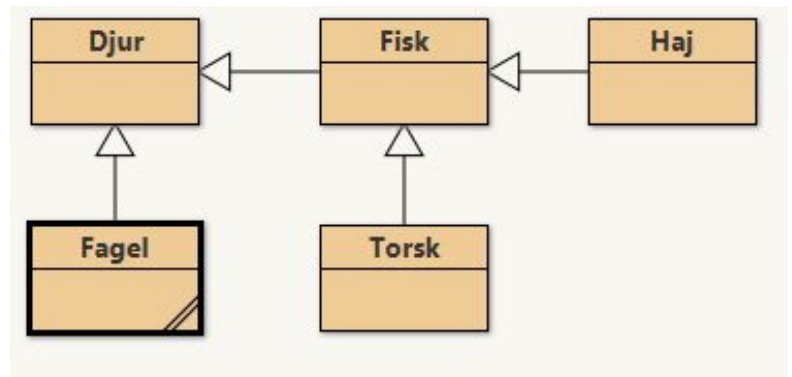
# Ändra signatur med overriding

```
class Superclass:
    def hej(self):
        print("Hej från Super!")
class Subclass(Superclass):
    def hej(self, x):
        print("Hej från Sub!")
        print("input:", x)
sup = Superclass()
sup.hej()    # Hej från Super!
sub = Subclass()
sub.hej(2)   # Hej från Sub! \\ input: 2
sub.hej()   # FEL!
```

## Förra veckan: Övningar introduktion till ärvning

Studera fler exempel från w3schools enligt länkar i lektionsanteckningarna.

1. Implementera en klass `Djur` med attributet `namn`. Skapa två subclasser till `Djur`: `Fagel` och `Fisk`. `Fagel` ska ha ett attribut `vingspann`. `Fisk` ska ha ett attribut `maxdjup`. Skapa två subclasser till `Fisk`: `Haj` och `Torsk`. `Haj` ska ha ett attribut `antalTänder`. `Torsk` ska ha ett attribut `hastighet`. Se bild (pil betyder ärvning):



2. Skriv en funktion `fånga(haj, torsk)` som returnerar `True` ifall (a) torskens hastighet är mindre än 30 och (b) hajens `maxdjup` är minst lika stort som torskens, annars `False`.

Övrigt: Kattis-utmaningar inför progolymp, glöm inte att pusha till GitHub

# Den här veckan: Övningar metoder i klasshierarkier

## Uppgifter:

1. Utöka den tidigare klasshierarkin så att `Djur` har publika metoder `at()` och `sov()`. `Fisk` ska ha en publik metod `simma()`. `Haj` ska ha en publik metod `at(djur)`. (Du får implementera metoderna som du vill, men se till att någon utskrift sker t.ex. "Fisken simmar" när du anropar metoden `simma()`.) Uppdatera även klassdiagrammet!
2. Lägg till klasserna `Cykel` och `Sportbil` i det inledande exemplet (med klasserna `Fordon` och `Bil`). I verkligheten är en cykel ett fordon och en sportbil en bil, så låt dem ärva på motsvarande sätt i din klasshierarki. Cykeln ska ha metoden `plinga` och sportbilen ska överskugga metoden `kör`. Du får själv bestämma vad som ska hända när man anropar metoderna, men enklast är att bara göra någon utskrift som i föregående övning. Rita ett klassdiagram.

Övrigt: Kattis, Progolymp