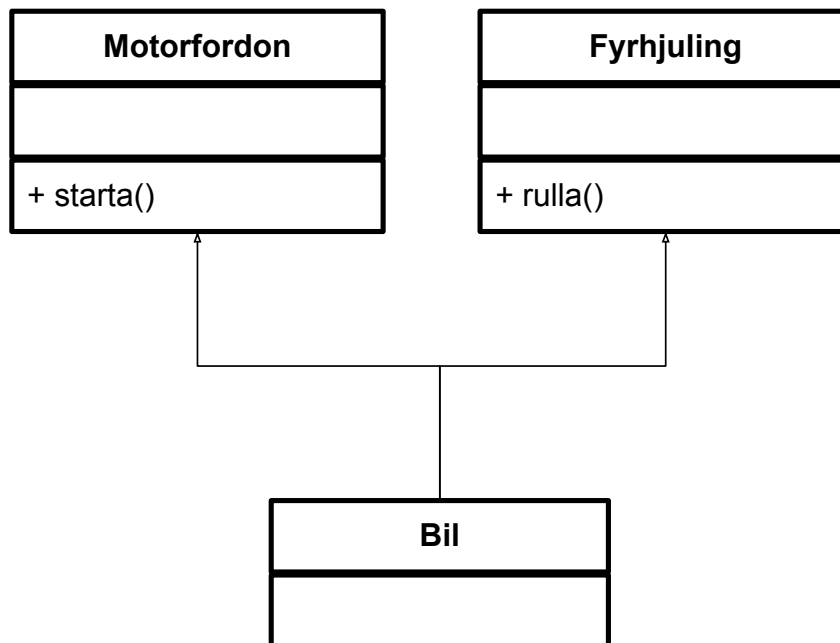


## 2.3 Flerfaldig ärvning

Holger Rosencrantz

Bil ärver från både Motorfordon och Fyrhjuling



# Syntax

```
class Motorfordon:
    def starta(self):
        print("Vroom!")
class Fyrhjuling:
    def rulla(self):
        print("Rullar!")
class Bil(Motorfordon, Fyrhjuling):
    pass
b = Bil()
b.starta() # Vroom!
b.rulla() # Rullar!
print(isinstance(b, Fyrhjuling) and isinstance(b, Motorfordon)) # True
```

## Konstruktörer och attribut vid flerfaldig ärvning

```
class Motorfordon:
    def __init__(self, max_speed):
        self.max_speed = max_speed
class Fyrhjuling:
    def __init__(self, wheel_radius):
        self.wheel_radius = wheel_radius
class Bil(Motorfordon, Fyrhjuling):
    def __init__(self, max_speed, wheel_radius):
        Motorfordon.__init__(self, max_speed)
        Fyrhjuling.__init__(self, wheel_radius)
b = Bil(120, 15)
print(b.max_speed) # 120
print(b.wheel_radius) # 15
```

# Förra 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

# Den här veckan: Övningar flerfaldig ärvning

## Uppgifter:

1. En häst är ett däggdjur och även ett landdjur. En val är ett däggdjur och ett havsdjur. En fisk är ett havsdjur, men inte ett däggdjur. En ödla är ett landdjur, men inte ett däggdjur. Hur ser arvshierarkin ut? Vilka attribut och metoder kan tänkas finnas i varje klass (minst ett attribut och en metod vardera i varje klass). Rita ett diagram och implementera hierarkin i Python!
2. Utforska och konstruera övriga tänkbara exempel efter behov och intresse:
  - a. Ett fantasy-spel där världen befolkas av människor, orcher och hobbitar, som var för sig kan vara krigare, trollkarlar eller tjuvar.
  - b. Ett grafiskt gränssnitt med rektanglar, cirklar och andra geometriska figurer, som var för sig kan vara klickbara, dragbara eller svara på annan användarinteraktion.

Övrigt: Kattis, Progolymp, JavaScript-projekt