IN1010 - Gruppe 11

Uke 4

Plan for dagen

- Overloading vs override
- Abstract
-

Overloading vs Overriding

- **Overloading**: skjer når flere metoder i samme klasse har samme navn, men ulike parametere **(kompileringstid)**
- **Overriding:** skjer når en subklasse gir en ny implementasjon av en metode fra superklassen **(kjøretid)**

```
class Dyr {
    void lagLyd() { System.out.println("Dyr lager lyd"); } // Overstyres
    void lagLyd(String s) { System.out.println(s); } // Overlastes
}

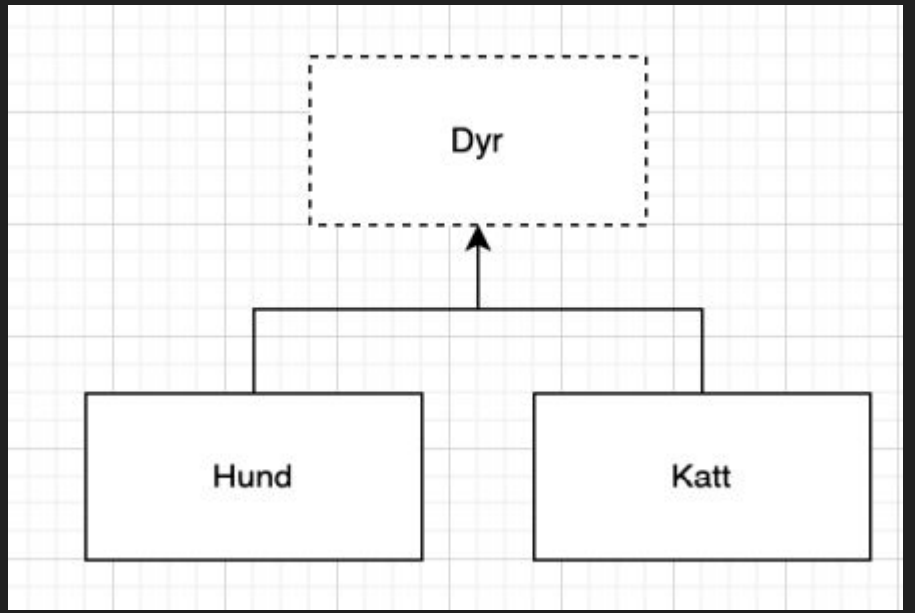
class Hund extends Dyr {
    @Override
    void lagLyd() { System.out.println("Voff"); } // Overstyring
}
```

Abstract

Vi snakket litt om abstract forrige uke, hva husker dere?

Abstract

- Abstrakt betyr at man ikke kan opprette objekt av klassen brukes som en en “blueprint” /”skall” for subklassene



Oppgave

```
class A {}
class B extends A {}
class C extends A {}
class D extends C {}

class TrueOrFalse {
        public static void main(String[] args) {
                A a = new A();
                B b = new B();
                C c = new C();
                D d = new D();
                A e = new D();
                A f = new B();
                System.out.println("a) " + (b instanceof A));
                System.out.println("b) " + (c instanceof D));
                System.out.println("c) " + (d instanceof A));
                System.out.println("d) " + (e instanceof C));
                System.out.println("e) " + (f instanceof C));
        }
}
```

Lab!