Pragmatisk programmering

Mer om arv og subklasser - Interface

Tema for uken

- Interface
  - Nøkkelord: `implements`
- Andre ønsker/behov?
  - Behov for å bruke mer tid på forrige ukes pensum?
  - (dette tar forrang!)

Interface

- Et interface (*norsk: grensesnitt*) er en samling egenskaper (rolle) som ikke naturlig hører hjemme i et arvehierarki

- Har ingen variabler
- Kun metodesignaturer – implementasjon skjer inne i klassen
- Multippel arv: Man kan bare arve fra én superklasse, men kan implementere flere interfacer!

```java
interface Dyr {
    boolean harEier();
    void lagLyd();
}

class Ulv implements Dyr {

    @Override
    public boolean harEier() {
        return false;
    }

    @Override
    public void lagLyd() {
        System.out.println("Awoooo");
    }
}
```

```java
class Hund implements Dyr {
    String eier;

    public Hund(String eier) {
        this.eier = eier;
    }

    @Override
    public boolean harEier() {
        return !eier.isEmpty();
    }

    @Override
    public void lagLyd() {
        System.out.println("Voff-voff");
    }
}
```

- Du kan implementere flere interfacer

```
class Lastebil extends Bil implements Miljøvennlig, Skattbar {}
```

- Du kan implementere flere interfacer

```
class Lastebil extends Bil implements Miljøvennlig, Skattbar {}
```

- Interfacer kan også arve!

```
interface Dyr {
    void lagLyd();
}

interface Husdyr extends Dyr {
    int hentPris();
}
```

- Du kan implementere flere interfacer

```
class Lastebil extends Bil implements Miljøvennlig, Skattbar {}
```

- Interfacer kan også arve!

```
interface Dyr {
    void lagLyd();
}

interface Husdyr extends Dyr {
    int hentPris();
}
```

- Interfacer kan brukes som typenavn

```
interface Skattbar {}

class Bil implements Skattbar {}

class Ost implements Skattbar {}

Skattbar[] s = {new Bil(), new Ost()};
```

Pensum så langt …

- Overgang fra Python til Java
  - Ny syntaks: krøllparanteser, semikolon osv.
  - If-sjekker, for-/while-løkker, brukerinput, fillesing osv.
- Klasser og subklasser (`extends`)
- Abstrakte klasser (`abstract`)
- Polymorfi (`@Override`)
- Interface (`implements`)

Før midtveis …

- Beholdere
- Generiske beholdere
- Designmønstre

- Behov frem til midtveiseksamen?