UKE 4!

Gruppe 9<3
ragnhinw@uio.no

Oblig 1

❌ if (0 <= teller){...}

✅ if (teller >= 0){...}

```
Person(String navn){
        this.navn = navn;
}

Person(String n){
        navn = n;
}
```

Arv - repetisjon

- class Subklasse extends Superklasse
- Super()

```
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));
        }
}
```

```java
//Tilfelle 1
B ab = (B) a;

//Tilfelle 2
if( e instanceof B ){
    B eb = (B) e;
}

//Tilfelle 3
D ed = (D) e;

//Tilfelle 4
if( e instanceof D ){
    D ed = (D) e;
}

//Tilfelle 5
B fb = (B) f;

//Tilfelle 6
C ec = (C) e;
```

Arv VS Komposisjon

- Bolig – Leilighet
- Leilighet – Stue
- Tre – Bjørk
- Person – Barn
- Tre - Bark

POLYMORFI

Polymorfi og override

- Samme metodesignatur som superklassens metode
- Nøkkelord: @Override
- Mest spesifikke metode blir kalt under kjøretid

Override VS overload

Override

- Samme metodesignatur, men mer typespesifikt innhold

Overload

- Samme metodenavn, men ulike parametere

Parkeringshus

Dere skal nå lage et program som visualiserer et parkeringshus. Et parkeringshus har parkeringsplasser hvor både Motorsykler og Biler kan parkere. Motorsykler og Biler har en del til felles, bl.a. har de begge en sjåfør og et registreringsnummer. Mens en bil også skal kunne holde på alt fra 2 til 5 personer (spesifisert i det bilen lages), derav en av de må være sjåføren.

1. Tegn klassehierarkiet
2. Lag alle klassene med instansvariabler. Inntil videre skal de ikke ha noen metoder. Det kan være lurt at Motorsykkel og Bil arver av samme klasse - hvorfor tror du det?
3. Parkeringshus skal ha to metoder: en for å sette inn en bil eller motorsykkel og en for å fjerne en bil eller motorsykkel på en gitt plass. Vi vil at metoden skal returnere bilen/motorsykkelen som står på plassen, og returnere null hvis det hverken finnes bil/motorsykkel på plassen. Når dere skal sette inn en bil eller motorsykkel, skal det gjøres på en ønsket plass. Lag unntakshåndtering til parkeringshuset ditt, for å håndtere tilfeller der man prøver å sette inn når det er fullt, eller på en ugyldig plass. **Hint:** ArrayIndexOutOfBoundsException.
4. Lag en metode skrivInformasjon(), som skriver ut all informasjon om hvert av kjøretøyene. Husk at motorsykler og biler har noe felles informasjon, siden begge er kjøretøy, men de har også noe særegen tilleggsinformasjon. Når du itererer gjennom kjøretøyene, skal all informasjon om hvert enkelt kjøretøy skrives til terminalen. Du kan ikke bruke instanceof i denne oppgaven, og du skal unngå å repetere kode

TRIX & OBLIG