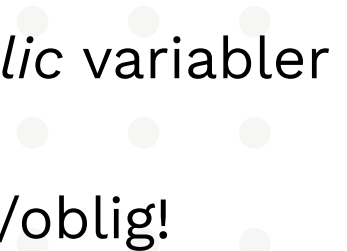




# IN1010 uke 3

Gruppe 7

# Agenda

- Noen vanlige feil
  - Arv og subklasser
    - Referansevariabler
    - Casting
    - *instanceOf*
    - *private/protected/public* variabler
    - Abstrakte klasser
  - Jobbe med oppgaver/trix/oblig!
- 

# Noen vanlige feil



```
class KattHovedprogram {
```

Run | Debug

```
public static void main(String[] args) {
```

```
    Katt[] mineKatter = new Katt[4];
```

```
    mineKatter[0] = new Katt("Bob");
```

```
    mineKatter[1] = new Katt("Tom");
```

```
    mineKatter[2] = new Katt("Pusur");
```

```
    for (Katt katt : mineKatter) {
```

```
        if(katt.harSpist() == false) {
```

```
            if(!katt.hentNavn().equals("Tom")) {
```

```
                katt.giMat();
```

```
            }
```

```
        }
```

```
Exception in thread "main" java.lang.NullPointerException: Cannot  
invoke "Katt.harSpist()" because "katt" is null  
at KattHovedprogram.main(KattHovedprogram.java:10)
```

# Finn feil i koden

- Hint:  
1 feil  
1 forbedring

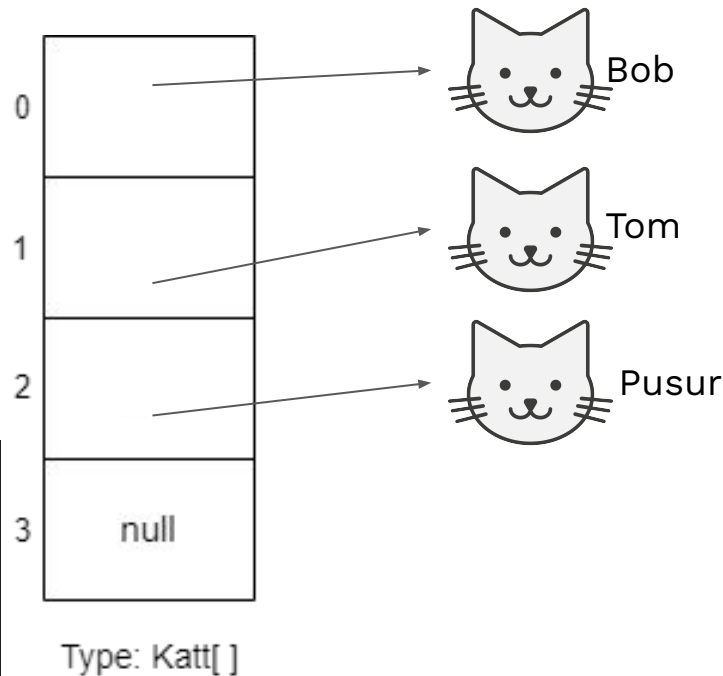
# NullPointerException

```
Katt[] mineKatter = new Katt[4];
```

```
mineKatter[0] = new Katt("Bob");  
mineKatter[1] = new Katt("Tom");  
mineKatter[2] = new Katt("Pusur");
```

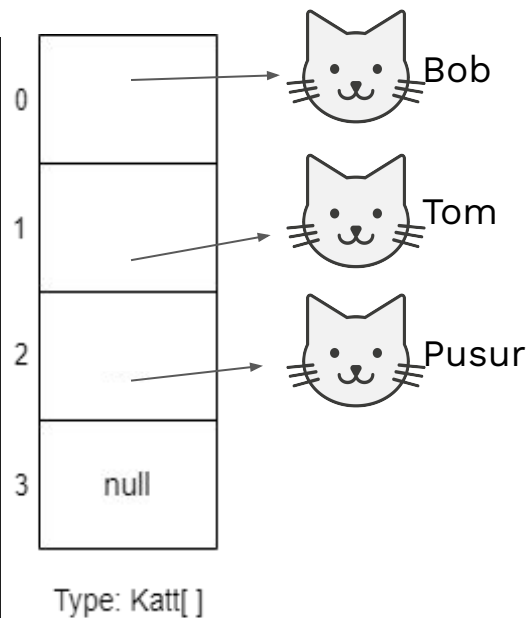
```
for (Katt katt : mineKatter) {  
    if(katt.harSpist() == false) {  
        if(!katt.hentNavn().equals("Tom")) {  
            katt.giMat();  
        }  
    }  
}
```

```
Exception in thread "main" java.lang.NullPointerException: Cannot  
invoke "Katt.harSpist()" because "katt" is null  
at KattHovedprogram.main(KattHovedprogram.java:10)
```



# Løsning

```
for (Katt katt : mineKatter) {  
    if (katt != null) {  
        if(katt.harSpist() == false) {  
            if(katt.hentNavn().equals("Tom")) {  
                katt.giMat();  
            }  
        }  
    }  
}
```



# Liten forbedring

```
for (Katt katt : mineKatter) {  
    if (katt != null) {  
        if(katt.harSpist() == false) {  
            if(katt.hentNavn().equals("Tom")) {  
                katt.giMat();  
            }  
        }  
    }  
}
```

# Liten forbedring

```
for (Katt katt : mineKatter) {  
    if (katt != null) {  
        if(!katt.harSpist()) {  
            if(katt.hentNavn().equals("Tom")) {  
                katt.giMat();  
            }  
        }  
    }  
}
```



# Arv og subklasser

1 minutt: skriv ned alt du  
husker om arv og subklasser!



# Eksempel: hunder

Object

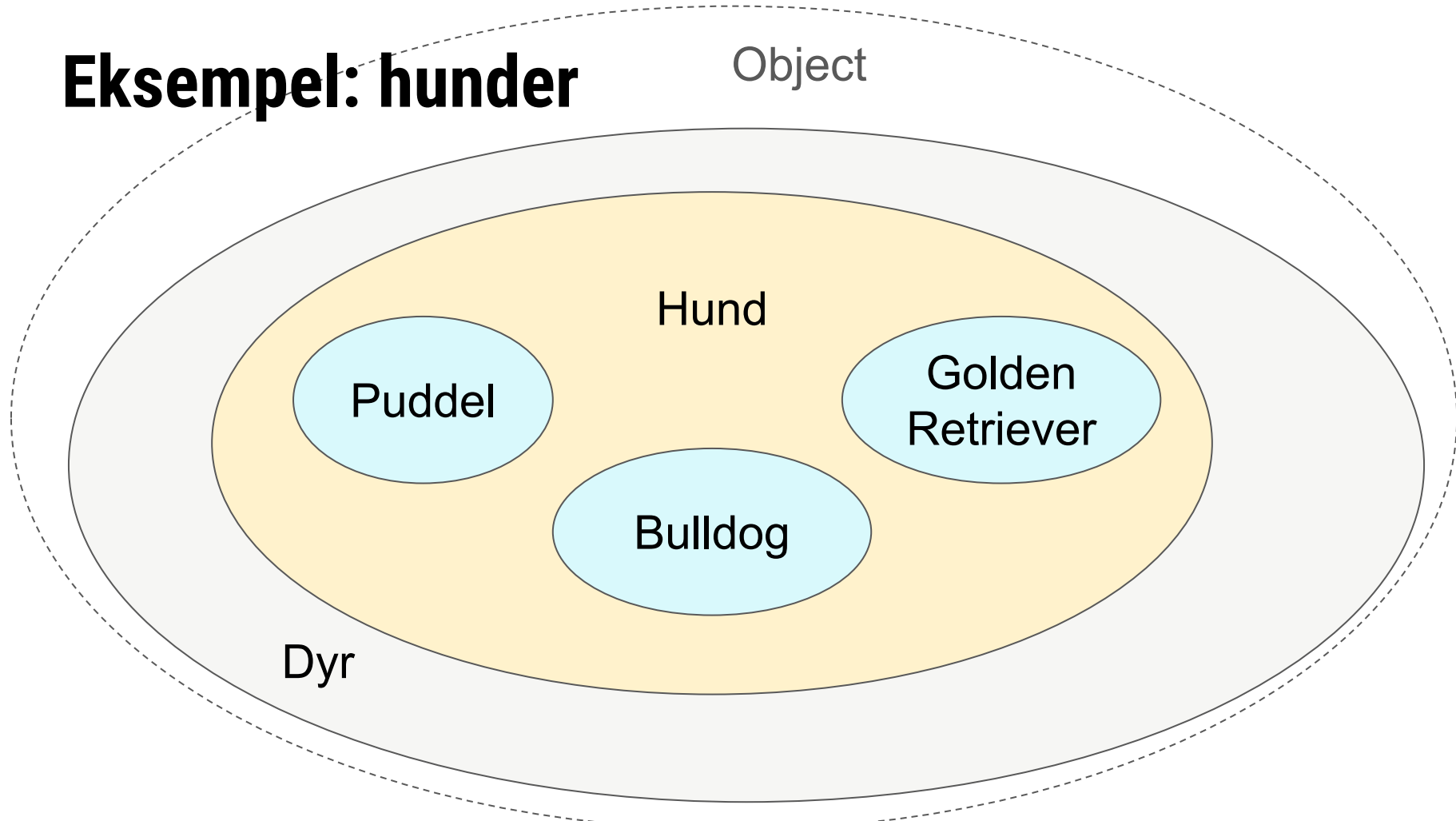
Hund

Puddel

Golden  
Retriever

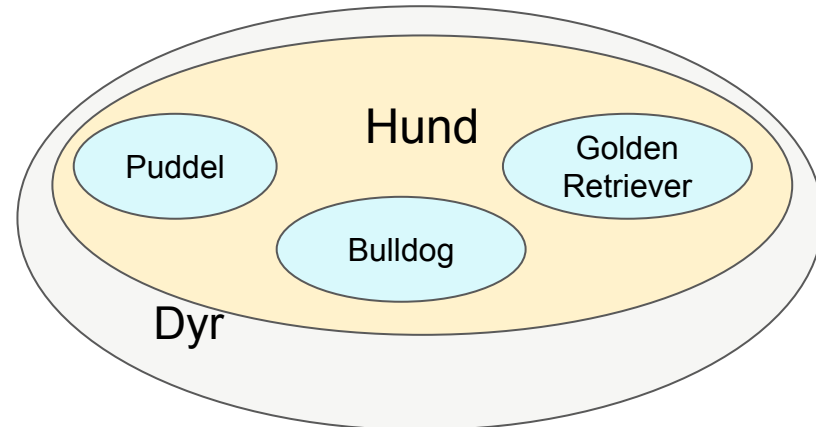
Bulldog

Dyr



# Eksempel: hunder

- Hvilke klasser arver fra hvilke?
- Hvordan og i hvilke klasser skal du skrive *extends*?



# Eksempel: hunder

*Svar:*

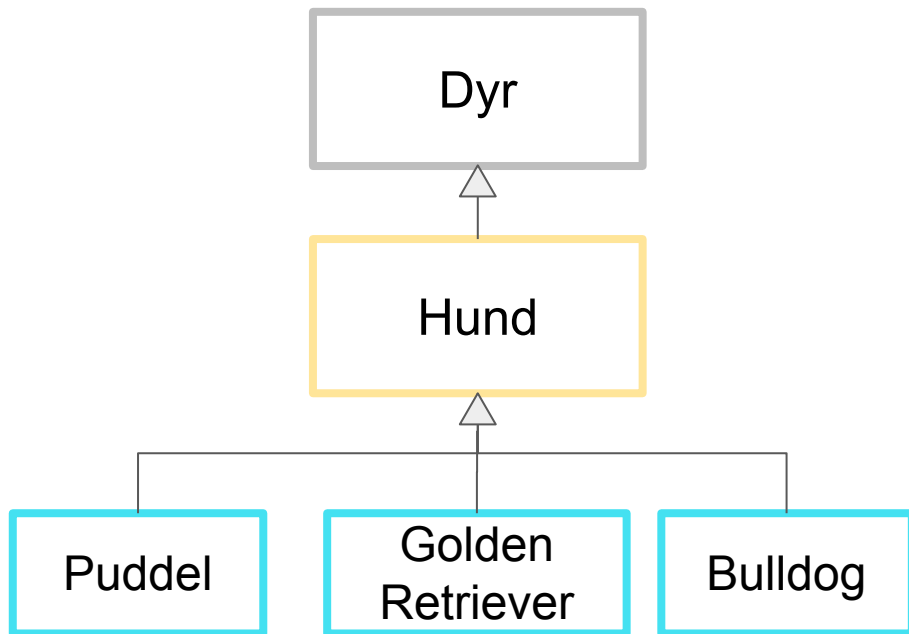
```
class Dyr
```

```
class Hund extends Dyr
```

```
class Puddel extends Hund
```

```
class Bulldog extends Hund
```

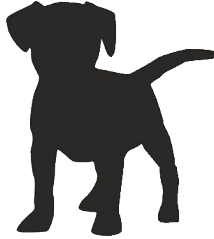
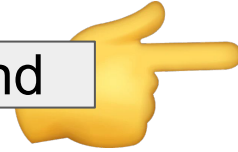
```
class GoldenRetriever extends Hund
```



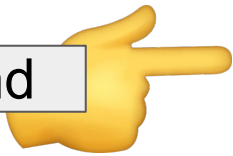
# Referansevariabler

```
Hund hund1 = new Hund();  
Hund hund2 = new GoldenRetriever();  
GoldenRetriever golden1 = new Hund();
```

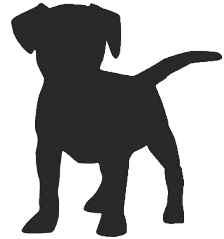
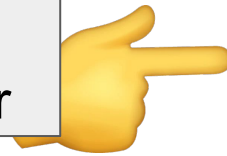
Hund



Hund



Golden  
Retriever



```
class Hund extends Dyr {  
    public void bjeff() {  
        System.out.println("Voff");  
    }  
}
```

```
class GoldenRetriever extends Hund {  
    public void vaerGlad() {  
        System.out.println("Jeg er såå glad!");  
    }  
}
```

*Hovedprogram:*

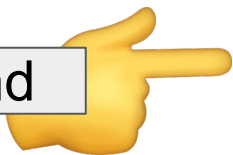
```
public static void main(String[] args) {  
    Hund minHund = new GoldenRetriever();  
    minHund.bjeff();  
    minHund.vaerGlad();  
}
```

Hva skjer når vi  
kjører  
hovedprogrammet?

⊗ The method vaerGlad() is undefined for the type Hund

```
public static void main(String[] args) {  
    Hund minHund = new GoldenRetriever();  
    minHund.bjeff();  
    minHund.vaerGlad();  
}
```

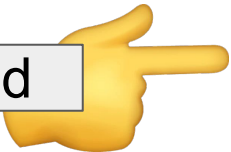
Hund



- Pekeren/variabelen “ser” bare at det er en Hund, ikke GoldenRetriever-egenskapene

# Casting

```
public static void main(String[] args) {  
    Hund minHund = new GoldenRetriever();  
    minHund.bjeff();  
  
    GoldenRetriever minRetriever = (GoldenRetriever) minHund;  
    minRetriever.bjeff();  
    minRetriever.vaerGlad();  
}
```

Hund 



Golden Retriever 





# InstanceOf

- Sjekker om et objekt er av, eller arver fra, en klasse

*Hva printes ut?*

```
Hund minHund = new GoldenRetriever();
Hund minHund2 = new Bulldog();
GoldenRetriever minRetriever = (GoldenRetriever) minHund;

System.out.println(minHund instanceof GoldenRetriever);      true
System.out.println(minHund2 instanceof GoldenRetriever);    false
System.out.println(minRetriever instanceof GoldenRetriever); true
System.out.println(minRetriever instanceof Hund);            true
```

# Protected

- private - bare synlig i klassen
- protected - synlig i klassen og alle subklasser
- public - synlig i alle klasser

```
class Hund extends Dyr {  
    protected String navn;  
    public void bjeff() {  
        System.out.println("Voff");  
    }  
}
```

# Abstrakte klasser

- Kan ikke lage objekt av dem
- Men kan arve fra dem

```
abstract class Dyr {  
    private boolean sulten = true;  
    public abstract void spis();  
}
```

# Jobb med oppgaver!

IN1010 Emnesiden → Grupper →  
Gruppe 7 → Uke3

