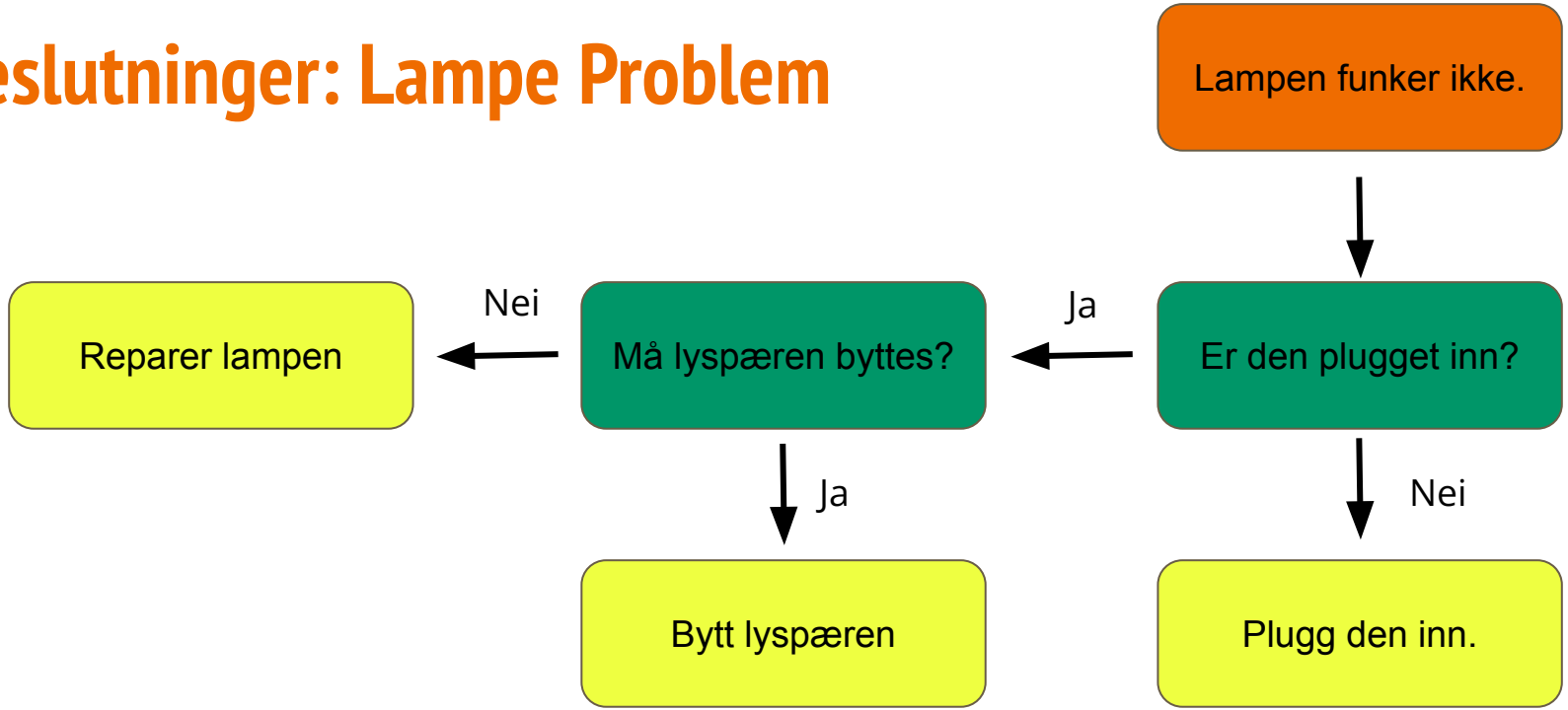
Beslutninger: Lampe Problem



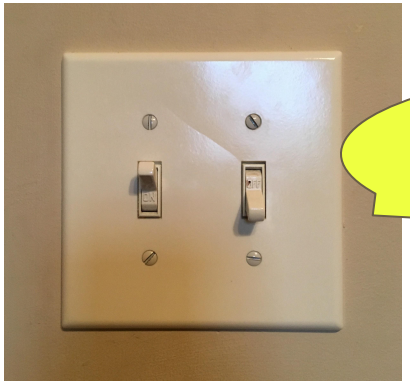
Seminar 2

Datatyper, Uttrykk, og Prosedyrer

Introduksjon

- Discourse
  - https://in1000-discourse.uio.no/

- Python Style Guide
  - https://peps.python.org/pep-0008/



Boolske uttrykk → lysbryter
enten **av** eller **på**

Evaluering av Uttrykk

Boolske uttrykk

Evalueres til **boolske verdier**: `True` eller `False`

Konnektivene: *`and`, `or`, `not`*

not

gjør et uttrykk til det motsatte av hva det er

and

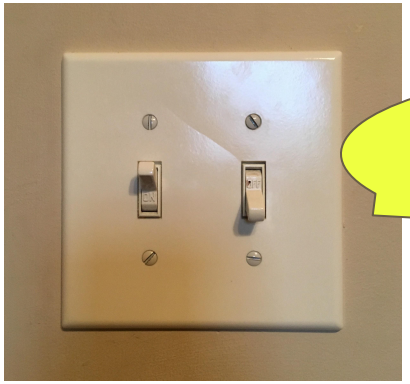
- begge uttrykkene må være sanne

`a and b` → True dersom både a er sann OG b er sann.

or

- bare ett av uttrykkene må være sant

`a or b` → True dersom enten a er True ELLER b er True



Boolske uttrykk → lysbryter
enten **av** eller **på**

Evaluering av Uttrykk

Boolske uttrykk

Evalueres til **boolske verdier**:
`True` eller `False`

Sammenligning

a == b	a er lik b
a != b	a er ikke lik b
a > b	a er større enn b
a >= b	a er større eller lik b
a < b	a er mindre enn b
a <= b	a er mindre eller lik b

Evaluering av Uttrykk

Hva evalueres de ulike uttrykkene til?

1. `True and True`
2. `True and False`
3. `True or True`
4. `True or False`
5. `False or False`
6. `not True`
7. `not False`

litt mer utfordrende:

8. `True and not True`
9. `True and not False`
10. `not True or not False`

1. True
2. False
3. True
4. True
5. False
6. False
7. True
8. False
9. True
10. True

Evaluering av Uttrykk

Hva evalueres de ulike uttrykkene til?

1. `6 > 6`
2. `6 >= 6`
3. `6 < 6`
4. `6 <= 6`
5. `6 == 6`
6. `6 != 6`

1. False
2. True
3. False
4. True
5. True
6. False

Datatyper

- Tekst (String)
- Heltall (Integer)
- Flyttall (Float)
- Boolsk Verdi (True/False)

NB: kan skrive strenger med både “” og ‘’

Hva er datatypen til *x*?

1. *x* = 3.14

   x er et **flyttall**.

2. *x* = 9

   x er et **heltall**.

3. *x* = “9“

   x er en **streng**.

4. *x* = True

   x er en **boolsk verdi**

Datatyper

- Tekst (String)
- Heltall (Integer)
- Flyttall (Float)
- Boolsk Verdi (True/False)

Typekonvertering: *casting*

- Gjør en verdi av en datatype om til en annen datatype

`int(x)` → *x* blir et heltall (integer)

`str(x)` → *x* blir en streng (string)

`float(x)` → *x* blir et flyttall (float)

NB: pass på at *x* kan være datatypen du prøve å konvertere den til!

Feil og Feilmeldinger

Tre hovedtyper feil:

1. syntaksfeil
2. logiske feil
3. exceptions

NB: Noen feil gir ikke feilmeldinger (f.eks. logiske feil!)

```
fornavn = 'Victoria'

if fornavn == 'Victoria'
    print('Ja! Det er deg!')
```

```
(base) vlhandfo@vlhandfo-XPS-13-9305:~/in1000-h22/uke-2$ python syntaksfeil.py
  File "/home/vlhandfo/in1000-h22/uke-2/syntaksfeil.py", line 4
    if fornavn == 'Victoria'
                           ^
SyntaxError: invalid syntax
(base) vlhandfo@vlhandfo-XPS-13                    h22/uke-2$
```

Type

Hvor på linjen feilen skjer

Kodelinjen

Finn 3 feil i dette programmet

```
tall = int(input("Skriv inn et tall: "))

if tall < 5:
   print(Tallet er større enn 5)
else:
   print(Tallet er ikke større enn 5)
```

1. Det inne `print()` skal stå med “”, ellers så gir det **syntaksfeil**.
2. `tall < 5` evalueres til `True` dersom tallet som skrives inn er mindre enn 5. Dette er en **logisk feil**.
3. Konverterer til int - tillater ikke annet enn heltall. Det kan gi til en **runtime error** dersom inputen ikke kan konverteres til et heltall (f.eks. bokstav).

Prosedyrer

Unngå å repetere kode!

- ASCII-katt eksempel
- Sjekk ut https://tinyurl.com/2p83zudz