

# Velkommen til gruppetime i IN1000



1. september 2021  
Jessie Yue Guan

# Planen for i dag

- Litt repetisjon
- If-setninger
- Feilmeldinger
- Aritmetiske operasjoner
- Relasjonelle operasjoner
- Logiske operasjoner
- Prosedyrer
- Kodeflyt

# Variabler



# Variabler – Oppretting og endring

- Vi oppretter en variabel ved å velge et navn, skrive et likhetstegn, og velge en verdi
  - For eksempel: **tall = 2**
- Vi kan senere endre denne verdien ved å skrive det samme, men velge en ny verdi
  - For eksempel: **tall = 4**

# Variabler – Oppretting og endring

- `x = 4`
- `y = 7`
- `x = y`
- `y = x`
- `print(x, y)`
- Vi oppretter en variabel med navn `x` og verdien `4`
- Vi oppretter en variabel med navn `y` og verdien `7`
- Vi setter `x` til å være lik `y`, slik at den endrer verdi fra `4` til `7`, `x` er nå lik `7` (!!!)
- Vi setter `y` til å være lik `x`, slik at den endrer verdi fra `7` til `7`, `y` er nå lik `7`
- Vi printer `x` og `y` og får: `7 7`

# Datatyper – Sjekking og endring

- Python er et svakt typet programmeringsspråk
- Det vil si at den er veldig fleksibel når det kommer til datatyper
- Noe som betyr at i de fleste tilfeller klarer den å sjekke og endre datatype automatisk
- Men det finnes unntak og da må programmereren sjekke og endre datatype manuelt

# Dat typer – Sjekking og endring

- Du kan sjekke datatypen til en variabel ved hjelp av `type()`
- Du kan endre datatypen til en variabel ved hjelp av `str()`, `int()`, `float()`, og `eval()`

- For eksempel:
  - `type(mitt_tall)`
  - `str(-12.3684)`
  - `int(249.583)`
  - `float(100)`
  - `eval("""`

# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! 😊



# Input



# Input

- For eksempel: **hoyde = input()**
- Funker, men vanskelig for brukeren å skjønne hva de skal gjøre

- 
- For eksempel: **print("Hvor høy er du i cm?")**
  - **hoyde = input()**
  - Bedre, nå har vi gode instruksjoner til brukeren

- 
- For eksempel: **hoyde = input("Hvor høy er du i cm?")**
  - Perfekt, nå har vi lettleselig kode også

Output



# Output/Print

- For eksempel: **print("Hei! Velkommen!")**
  - Funker, men ikke veldig fleksibelt
- 
- For eksempel: **print("Hei!")**
  - **print(navn)**
  - **print("Velkommen!")**
  - Bedre, nå har vi gode tilbakemeldinger til brukeren
- 
- For eksempel: **print("Hei", navn, "og velkommen!")**
  - Perfekt, nå har vi lettleselig kode også

# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! 😊

# Betingelser og beslutninger



# If, elif, og else

- `antall_katter = 3`
  - `if` `antall_katter < 1`:
    - `print("Liker du ikke katter?")`
  - `elif` `antall_katter < 3`:
    - `print("Du er glad i katter :)")`
  - `elif` `antall_katter < 5`:
    - `print("Du er veldig glad i katter!")`
  - `else`:
    - `print("Du har kanskje litt vel mange katter?")`
- Hva blir printet ut?
  - **Du er veldig glad i katter!**
  - Hvorfor?
    - Fordi `3 < 3` er ikke sant
    - Altså 3 er ikke mindre enn 3
      - Men `3 <= 3` er sant
      - Altså 3 er mindre enn **eller lik** 3

# If, elif, og else

- `alder = 10`
- `if alder < 18:`
  - `print("Du er myndig!")`
- `print("Du er ikke myndig!")`

- 
- `alder = 10`
  - `if alder < 18:`
    - `print("Du er myndig!")`
  - `else:`
    - `print("Du er ikke myndig!")`

- Hva er forskjellen mellom disse to programmene?
- I det øverste programmet får brukeren både beskjed om at de er myndig og at de ikke er det.
- Dette er forvirrende for brukeren som ikke nødvendigvis kan programmere. Regnes de som myndige eller ikke???
- Når en setning evalueres til sann og innholdet blir utført vil Python automatisk hoppe over resten av de tilhørende setningene



# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! 😊

# Feilmeldinger



# Hvordan lese feilmeldinger?

- **navn = Geir**

- NameError: name 'Geir' is not defined

- **if 2>1:**

- **print("ja")**

- IndentationError: expected an indented block

- **print("ja"**

- SyntaxError: invalid syntax

- **alder = int(input("Hvor gammel er du?"))**

- ValueError: invalid literal for int() with base 10: "

# Oppgave med feilmeldinger

- `tall = input("Skriv inn et tall: ")`
- `if svar < 5:`
- `print("Tallet er større enn 5")`
- `else`
- `print(Tallet er ikke større enn 5)`

# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! 😊

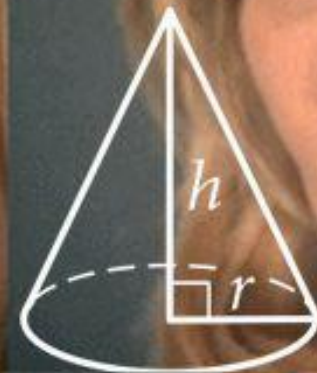
# Operasjoner



$$A = \pi r^2$$

$$C = 2\pi r$$

$$V = \frac{1}{3} \pi r^2 h$$



$$V = \pi r^2 h$$

	30°	45°	60°
sin	$\frac{1}{2}$	$\frac{\sqrt{2}}{2}$	$\frac{\sqrt{3}}{2}$
cos	$\frac{\sqrt{3}}{2}$	$\frac{\sqrt{2}}{2}$	$\frac{1}{2}$
tan	$\frac{\sqrt{3}}{3}$	1	$\sqrt{3}$



$$\int \sin x dx = -\cos x + C$$

$$\int \frac{dx}{\cos^2 x} = \operatorname{tg} x + C$$

$$\int \operatorname{tg} x dx = -\ln|\cos x| + C$$

$$\int \frac{dx}{\sin x} = \ln\left|\operatorname{tg} \frac{x}{2}\right| + C$$

$$\int \frac{dx}{a^2 + x^2} = \frac{1}{a} \operatorname{arctg} \frac{x}{a} + C$$

$$\int \frac{dx}{x} = \ln|x| + C$$



$$ax^2 + bx + c = 0$$

$$a\left(x^2 + \frac{b}{a}x + \frac{c}{a}\right) = 0$$

$$x^2 + 2\frac{b}{2a}x + \left(\frac{b}{2a}\right)^2 - \left(\frac{b}{2a}\right)^2 + \frac{c}{a} = 0$$

$$\left(x + \frac{b}{2a}\right)^2 - \frac{b^2 - 4ac}{4a^2} = 0$$

# Konkatenering

- Man kan også bruke + for å sette sammen to strenger etter hverandre

strengverdi	+	strengverdi
strengverdi	+	strengvariabel
strengvariabel	+	strengverdi
strengvariabel	+	strengvariabel

"Hei "	+	"Ola"
"Hei"	+	" Ola"

- Det vil ikke føre til at det blir et mellomrom mellom strengene
- Så dette må du huske å legge til selv hvis du vil ha det
- Enten på slutten av venstrestringen eller på begynnelsen av høyrestrengen
- NB!!! Kan ikke brukes mellom strenger og andre datatyper (int, float, boolean)
- Sjekk datatypen ved hjelp av type() eller konverter til en streng ved hjelp av str()

# Aritmetiske operasjoner

- Brukes på integers eller floats
- Resulterer i en integer eller en float
- Består av følgende:
  - Addisjon (+)
  - Subtraksjon (-)
  - Multiplikasjon (\*)
  - Divisjon (/)
  - Potens (\*\*)
  - Heltallsdivisjon (//)



# Aritmetiske operasjoner - rekkefølge

- Parentesene løses først
  - For eksempel:  $4 * (7 + 3)$  blir **40**
- Multiplikasjon og divisjon løses etter det
  - For eksempel:  $6 + 2 * 5$  blir **16**
- Addisjon og subtraksjon løses etter det
  - For eksempel:  $20 - 10 / 5$  blir **18**

# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! 😊

# Logiske operasjoner

- Brukes på booleans
- Resulterer i en boolean
- Består av følgende:
  - negasjon (**not**) som betyr "ikke"
  - konjunksjon (**and**) som betyr "både og"
  - disjunksjon (**or**) som betyr "enten eller"

# Logiske operasjoner - not

- La oss si at vi har en boolean variabel som heter **min\_bool**
- Hvis vi setter **True** som verdien til **min\_bool** betyr det at **not min\_bool** vil bli **False**
- Hvis vi setter **False** som verdien til **min\_bool** betyr det at **not min\_bool** vil bli **True**

<b>min_bool</b>	<b>not min_bool</b>
True	False
False	True

# Logiske operasjoner - and

- La oss si at vi har en boolean variabel kalt **x** og en boolean variabel kalt **y**
- Hvis *både* **x** og **y** har verdien **True**, så vil verdien av **x and y** også bli **True**
- I alle andre tilfeller vil **x and y** bli **False**

x	y	x and y
True	True	True
False	True	False
True	False	False
False	False	False

# Logiske operasjoner - or

- La oss si at vi har en boolean variabel kalt **x** og en boolean variabel kalt **y**
- Hvis *både* **x** og **y** har verdien **False**, så vil verdien av **x or y** også bli **False**
- I alle andre tilfeller vil **x or y** bli **True**

x	y	x or y
True	True	True
False	True	True
True	False	True
False	False	False

# Logiske operasjoner - rekkefølge

- Du skal alltid løse **not** først
  - For eksempel: **not True or True** blir altså **False**
- Deretter skal du løse **and**
  - For eksempel: **True and False or True** blir altså **False**
- Til slutt skal du løse **or**
  - For eksempel: **not False or False** blir altså **True**

# Oppgaver med logiske operasjoner

not False

not True

True and True

True and False

False and True

False and False

True or True

True or False

False or True

False or False

True

False

True

False

False

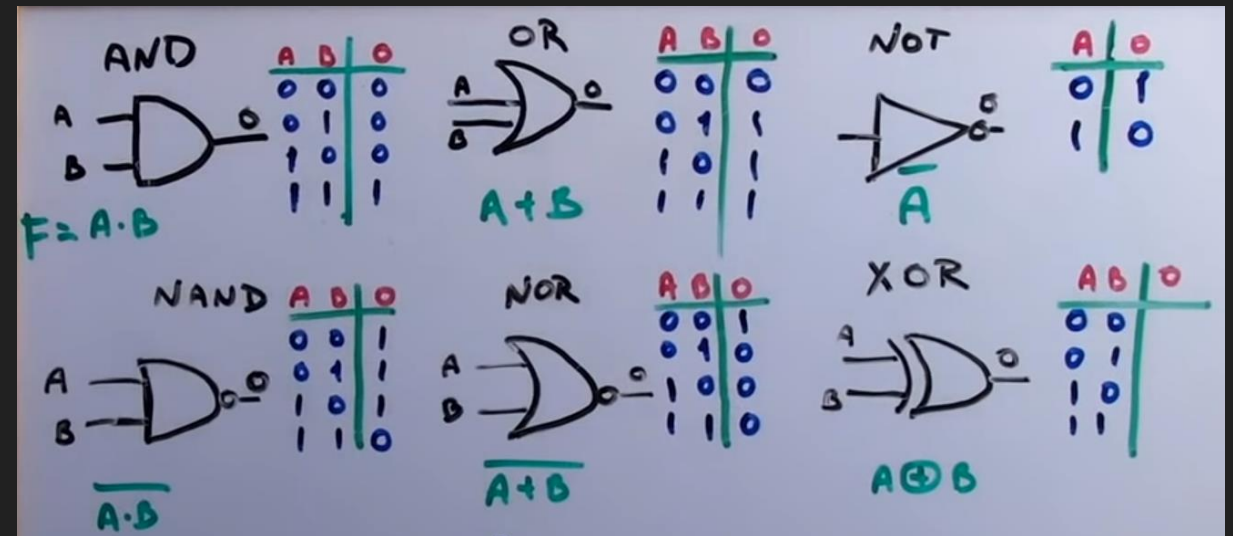
False

True

True

True

False





# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! 😊

# Relasjonelle operasjoner

- Brukes på integers (heltall) og floats (desimaltall)
- Resulterer i en boolean
- Består av følgende:
  - Er lik (==)
  - Er ikke lik (!=)
  - Mindre enn (<)
  - Mindre enn eller lik (<=)
  - Større enn (>)
  - Større enn eller lik (>=)

# Relasjonelle operasjoner - rekkefølge

- Rekkefølgen går fra venstre til høyre
- $x < y < z$  skal tolkes som  $x < y$  and  $y < z$

$x < y$	$y < z$	$(x < y) \text{ and } (y < z)$
True	True	True
False	True	False
True	False	False
False	False	False

# Oppgaver med relasjonelle operasjoner

$0 == 0$

True

$0 == 1$

False

$1 != 0$

True

$1 != 1$

False

$2 < 3$

True

$3 < 2$

False

$5 > 4$

True

$4 > 5$

False

$6 \leq 7$

True

$7 \leq 6$

False

$9 \geq 8$

True

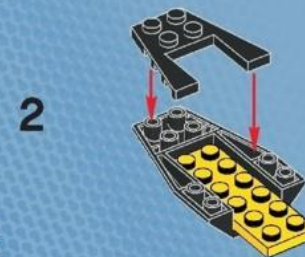
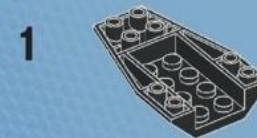
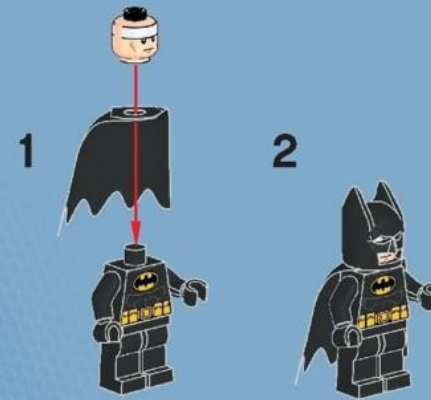
$8 \geq 9$

False

# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! 😊

# Prosedyrer



# Prosedyrer

- En variabel er en måte å lagre en verdi for senere bruk på
- En prosedyre er en måte å lagre en kodeblokk for senere bruk på
- En kodeblokk består av kodelinjer som har noe med hverandre å gjøre
- For eksempel, kodelinjer som regner ut volumet til en sylinder hører sammen
- Prosedyrer lar oss kjøre den samme kodeblokken flere ganger uten å måtte skrive alt på nytt

# Prosedyrer

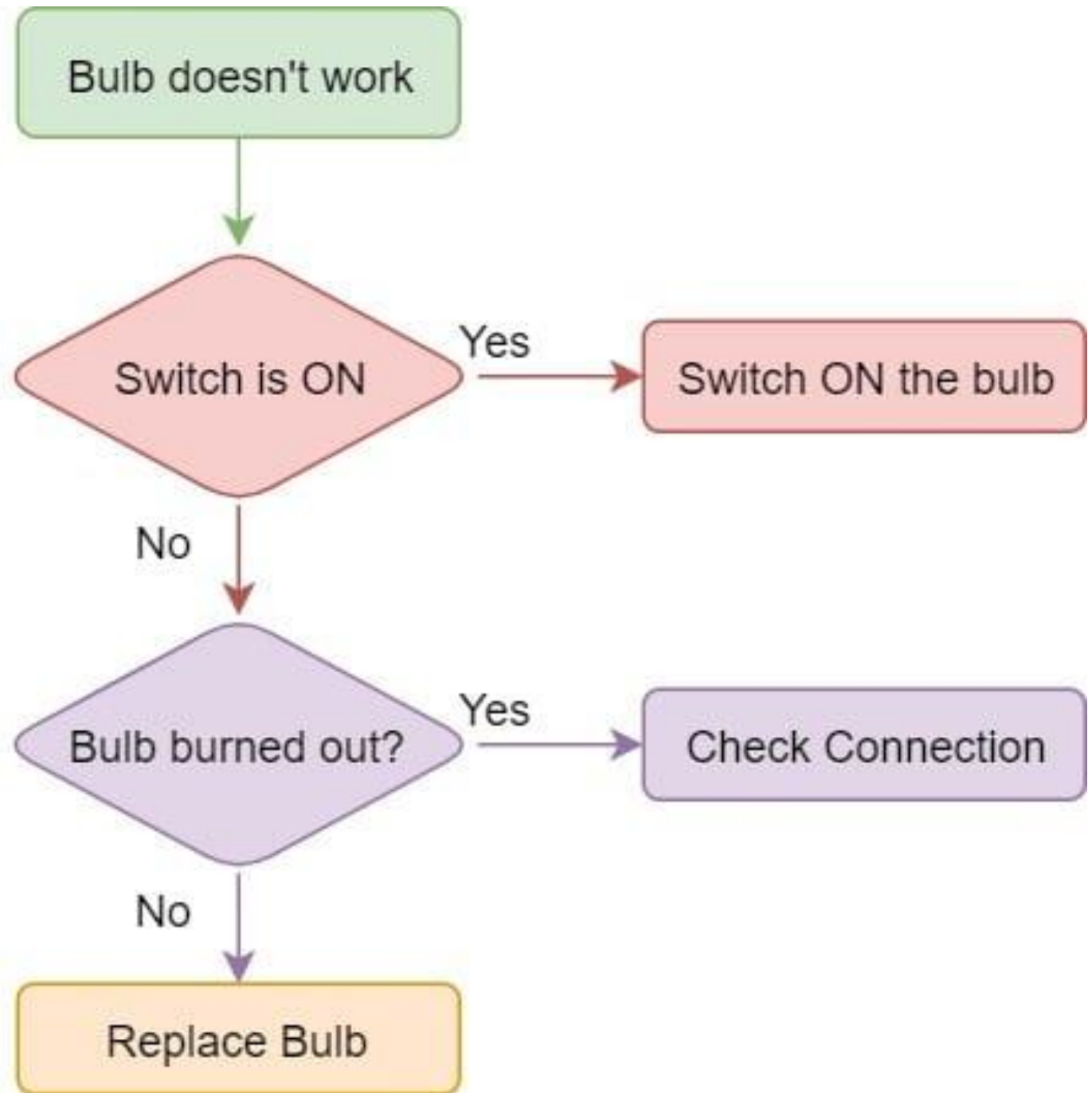
- `print("Velkommen til bursdagsprogrammet!")`
  - `def registrer_bursdag():`
    - `navn = input("Hva heter du? ")`
    - `bursdag = input("Når har du bursdag? ")`
    - `print("Bursdagen til", navn, "som er", bursdag, "er registrert!")`
  - `registrer_bursdag()`
  - `registrer_bursdag()`
  - `registrer_bursdag()`
- ← Definere prosedyren
- ← Fylle opp prosedyren
- ← Kjøre prosedyren



# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! 😊

# Kodeflyt



# Kodeflyt

- `def kvadrer() :`
- `tall = float(input("Skriv inn et tall: "))`
- `print("Produkt: ", tall*tall)`
  
- `def velkommen() :`
- `navn = input("Hva heter du? ")`
- `print("Velkommen til IN1000", navn)`
  
- `def siHallo() :`
- `print("Hallo!")`
  
- `siHallo()`
- `velkommen()`
- `print("Klar for å kvadrere litt?")`
- `kvadrer()`

- `> Hallo!`
- `> Hva heter du? Kristine`
- `> Velkommen til IN1000 Kristine`
- `> Klar for å kvadrere litt?`
- `> Skriv inn et tall: 2`
- `> Produkt: 4.0`

# Kodeflyt

```
14. def kvadrer() :
15.     tall = float(input("Skriv inn et tall: "))
17.     print("Produkt: ", tall*tall)

6.  def velkommen() :
7.     navn = input("Hva heter du? ")
9.     print("Velkommen til IN1000", navn)

2.  def siHallo() :
3.     print("Hallo!")

1.  siHallo()
5.  velkommen()
11. print("Klar for å kvadrere litt?")
13. kvadrer()
```

```
4. > Hallo!
8. > Hva heter du? Kristine
10. > Velkommen til IN1000 Kristine
12. > Klar for å kvadrere litt?
16. > Skriv inn et tall: 2
18. > Produkt: 4.0
```

# Spørsmål?

- Ikke vær redd for å spørre, det finnes ingen dumme spørsmål! 😊