

GRAFISKE BRUKERGRENSESNITT

INF100

HØST 2026

Torstein Strømme

FLEKSIBLE TEGNINGER

Skriv en funksjon som tegner et fjes mellom (0, 0) og (400, 400)
Fyll hele lerretet

face.py

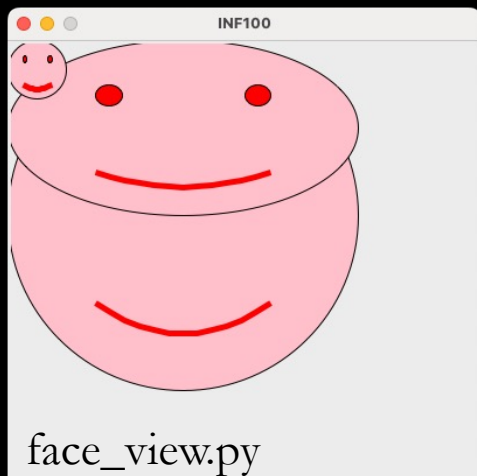
```
def draw_face_fixed(canvas):  
    ...
```

face_view.py

```
from face import draw_face_fixed  
from uib_inf100_graphics.simple import canvas, display  
  
draw_face_fixed(canvas)  
  
display(canvas)
```



Skriv en funksjon som tegner et fjes mellom (0, 0) og (width, height)



face.py

```
def draw_face_scaled(canvas, width, height):  
    ...
```

```
from face import draw_face_scaled  
from uib_inf100_graphics.simple import canvas, display  
  
draw_face_scaled(canvas, 300, 300)  
draw_face_scaled(canvas, 300, 150)  
draw_face_scaled(canvas, 50, 50)  
  
display(canvas)
```

Gang alle x-verdier med width/400
Gang alle y-verdier med height/400

Skriv en funksjon som tegner et fjes som begynner i (x_lft, y_top) og har den angitte bredde og høyde



face.py

```
def draw_face_shifted(canvas, x_lft, y_top, width, height):  
    ...
```

```
from face import draw_face_shifted  
from uib_inf100_graphics.simple import canvas, display  
  
draw_face_shifted(canvas, 50, 50, 300, 300)  
draw_face_shifted(canvas, 50, 50, 50, 50)  
draw_face_shifted(canvas, 300, 50, 50, 50)  
  
display(canvas)
```

Legg til x_lft til alle x-verdier
Legg til y_top til alle y-verdier

Skriv en funksjon som tegner et fjes sentrert rundt (x, y)
med radius r



face.py

```
def draw_face(canvas, x, y, radius):  
    ...
```

```
from face import draw_face_shifted  
from uib_inf100_graphics.simple import canvas, display  
  
draw_face(canvas, 200, 200, 100)  
draw_face(canvas, 200, 200, 30)  
draw_face(canvas, 50, 100, 50)  
  
display(canvas)
```

$x_{\text{lft}} = x - \text{radius}$
 $y_{\text{top}} = y - \text{radius}$
 $\text{width} = \text{height} = 2 * \text{radius}$
Så kall til `draw_face_shifted`



EVENT_APP



```
from uib_inf100_graphics.event_app import run_app

def app_started(app):
    app.count = 42

def key_pressed(app, event):
    app.count += 1

def redraw_all(app, canvas):
    cx = app.width / 2
    cy = app.height / 2
    canvas.create_text(cx, cy, text=f'{app.count}', font='Arial 60')

run_app(width=500, height=200)
```



```
from uib_inf100_graphics.event_app import run_app

def app_started(app):
    app.count = 42

def key_pressed(app, event):
    app.count += 1

def redraw_all(app, canvas):
    cx = app.width / 2
    cy = app.height / 2
    canvas.create_text(cx, cy, text=f'{app.count}', font='Arial 60')

run_app(width=500, height=200)
```



```
from uib_inf100_graphics.event_app import run_app

def app_started(app):
    app.count = 42

def key_pressed(app, event):
    app.count += 1

def redraw_all(app, canvas):
    cx = app.width / 2
    cy = app.height / 2
    canvas.create_text(cx, cy, text=f'{app.count}', font='Arial 60')
```

```
run_app(width=500, height=200)
```

← *for å starte programmet*



```
from uib_inf100_graphics.event_app import run_app
```

```
def app_started(app):  
    app.count = 42
```



kalles når programmet starter

```
def key_pressed(app, event):  
    app.count += 1
```

```
def redraw_all(app, canvas):  
    cx = app.width / 2  
    cy = app.height / 2  
    canvas.create_text(cx, cy, text=f'{app.count}', font='Arial 60')
```

```
run_app(width=500, height=200)
```



```
from uib_inf100_graphics.event_app import run_app
```

```
def app_started(app):  
    app.count = 42
```

```
def key_pressed(app, event):  
    app.count += 1
```

```
def redraw_all(app, canvas):  
    cx = app.width / 2  
    cy = app.height / 2  
    canvas.create_text(cx, cy, text=f'{app.count}', font='Arial 60')
```

```
run_app(width=500, height=200)
```

*tenk på **app** som en samling med variabler*



```
from uib_inf100_graphics.event_app import run_app
```

```
def app_started(app):  
    app.count = 42
```

```
def key_pressed(app, event):  
    app.count += 1
```

```
def redraw_all(app, canvas):  
    cx = app.width / 2  
    cy = app.height / 2  
    canvas.create_text(cx, cy, text=f'{app.count}', font='Arial 60')
```

```
run_app(width=500, height=200)
```

oppretter en ny variabel «count»



```
from uib_inf100_graphics.event_app import run_app
```

```
def app_started(app):  
    app.count = 42
```

```
def key_pressed(app, event):  
    app.count += 1
```

kalles (av uib_inf100_graphics) når lærretet skal tegnes



```
def redraw_all(app, canvas):  
    cx = app.width / 2  
    cy = app.height / 2  
    canvas.create_text(cx, cy, text=f'{app.count}', font='Arial 60')
```

```
run_app(width=500, height=200)
```



```
from uib_inf100_graphics.event_app import run_app
```

```
def app_started(app):  
    app.count = 42
```

```
def key_pressed(app, event):  
    app.count += 1
```

```
def redraw_all(app, canvas):  
    cx = app.width / 2  
    cy = app.height / 2  
    canvas.create_text(cx, cy, text=f'{app.count}', font='Arial 60')
```

```
run_app(width=500, height=200)
```

tegning gjøres akkurat som før





```
from uib_inf100_graphics.event_app import run_app
```

```
def app_started(app):  
    app.count = 42
```

```
def key_pressed(app, event):  
    app.count += 1
```

```
def redraw_all(app, canvas):  
    cx = app.width / 2  
    cy = app.height / 2  
    canvas.create_text(cx, cy, text=f'{app.count}', font='Arial 60')
```

```
run_app(width=500, height=200)
```

variabler for width og height finnes i app fra før



```
from uib_inf100_graphics.event_app import run_app
```

```
def app_started(app):  
    app.count = 42
```

```
def key_pressed(app, event):  
    app.count += 1
```

```
def redraw_all(app, canvas):  
    cx = app.width / 2  
    cy = app.height / 2  
    canvas.create_text(cx, cy, text=f'{app.count}', font='Arial 60')
```

```
run_app(width=500, height=200)
```

variabel vi nettopp opprettet i app_started





```
from uib_inf100_graphics.event_app import run_app
```

```
def app_started(app):  
    app.count = 42
```

```
def key_pressed(app, event):  
    app.count += 1
```

← kalles hver gang en tast trykkes

```
def redraw_all(app, canvas):  
    cx = app.width / 2  
    cy = app.height / 2  
    canvas.create_text(cx, cy, text=f'{app.count}', font='Arial 60')
```

```
run_app(width=500, height=200)
```

app

