

Galaktisk Galskab i PyGame Zero – Workshop 1

Du kender måske Galaktisk Galskab som et Scratch-program fra Intro-workshoppen. I denne workshop vil vi lave en version af spillet i Python med spil-biblioteket PyGame Zero.

Installér udviklingsmiljø og PyGame Zero

- Installér *Thonny Python IDE* fra <https://thonny.org/>
- Start *Thonny* og vælg *Tools* → *Manage Packages...* Skriv *pgzero* i *PyPI* søgefeltet for at finde PyGame Zero. Klik på *pgzero* og vælg *Install*
- Prøv PyGame Zero med følgende program

```
import pgzrun

radius = 0

def draw():
    screen.clear()
    screen.draw.circle((300, 300), radius, 'orange')

def update():
    global radius
    radius = (radius + 1) % 100

pgzrun.go()
```

Funktionerne `draw()` og `update()` er *callback*-funktioner, som bliver kaldt af PyGame Zero, når skærmen skal opdates. PyGame Zero kalder `draw()`, hver gang der skal tegnes, og `update()`, når spillets tilstand skal opdateres.

- Objektet `screen` repræsenterer spil-skærmen. Prøv at google ”pgzero screen” for at finde flere `draw` kommandoer, der kan tegne på skærmen.

Tegn stjerne-baggrund, der bevæger sig

- Prøv PyGame Zero med følgende program.
- Før programmet kan køre, skal grafikfilen `stjerner.png` være gemt i en underfolder, der hedder `images`.
- Du hente og gemme `images` fra <https://github.com/stee0640/pgzero2025>

```
import pgzrun

WIDTH = 600
HEIGHT = 600

bg_y = -600

def draw():
    screen.blit('stjerner.png', (0, bg_y))
```

```
def update():
    global bg_y
    bg_y += 1
    if bg_y > 0:
        bg_y = -1200

pgzrun.go()
```

Programmet flytter baggrundsbilledet nedad for at skabe en illusion af at man flyver på stjernehimlen. Funktionen `blit` er lynhurtig til at kopiere billeder ind på skærmen.

- Prøv at finde et lille billede, fx en planet, og brug `blit` til at få det ind på skærmen oven på stjernehimlen. `(0, 0)` er det øverste venstre hjørne på skærmen. Nederste højre hjørne er `(WIDTH, HEIGHT)`
- Prøv at bevæge billedet nedad med en hurtigere hastighed end stjernerne. Det giver en *parallax*-effekt, som er en 3D effekt, du måske kender fra andre 2D spil.

Sådan laves uendeligt scroll af baggrundsbillede

- Filen `stjerner.png` består af tre sammensatte skærbilleder. Nr. 1 og 3 er helt ens.
- Startværdien `-600` for `bg_y` betyder, at nr. 1 er gemt over den synlige skærm. Man ser altså nr. 2 på skærmen.
- Nr. 1 kan scrolles ned indtil den fylder hele skærmen. Derefter springer programmet til det identiske nr. 3. Det kan ikke ses, fordi nr. 1 og nr. 3 er ens og det hele gentages uendeligt.
-
- Prøv at se på `stjerner.png` i et billedvisnings-program. Man kan godt indsætte flere billeder, bare første og sidste billede er ens og har blød overgang til deres nabo.
- Hvis du er god til grafik, så prøv at lave din egen bane i et grafikprogram som *gimp*.

Indsæt en Actor på skærmen

I PyGame Zero bruger man `Actor()`, til at lave de objekter, man vil animere og eventuelt tjekke for om rammer hinanden. Det er det samme som en *sprite*, du måske kender fra Scratch. Herunder laver vi et rumskib, som du kan sætte ind i toppen af det tidligere program.

```
rumskib = Actor('rumskib1.png', (300, 550))
```

For at `Actor`-objektet skal blive tegnet i hvert billede skal man kalde dets `draw`-metode fra funktionen `draw()` i PyGame Zero-programmet. Så indsæt dette i `draw()`:

```
rumskib.draw()
```

Hvis rumskibet skal bevæge sig rundt, skal vi ændre på `Actor`-objektets koordinator. Indsæt f.eks. dette i `update()`:

```
if keyboard.right:
    rumskib.x += 1
if keyboard.left:
    rumskib.x -= 1
```

- Kan du få rumskibet til at bevæge sig op og ned også?