Icon

Description automatically generated**Pygame Zero Cheat Sheet**

**Ein Fenster erstellen**

Mit WIDTH und HEIGHT kann die Fenstergrösse definiert werden:

WIDTH = 600

HEIGHT = 600

**Die update Funktion**

Die update Funktion ist das Herzstück des Spiels. Sie wird immer wieder aufgerufen, solange das Spiel läuft.

In diese Funktion wird alles eingebaut, das laufend passieren oder überprüft werden soll.

Icon

Description automatically generated

**Eine Spielfigur erstellen**

Figuren oder Gegenstände mit einer Interaktion heissen bei Pygame Zero „Actor“. Das Bild für die Spielfigur muss im „images“ Ordner abgelegt sein.

Eine Figur wird so erstellt:

figur = Actor('bunny')

Wobei „bunny“ hier der Name des Bildes ist, das für die Spielfigur verwendet wird.

Danach muss noch die Anfangsposition der Figur angegeben werden (x und y im Koordinatensystem):

figur.x = 300

figur.y = 300

In der draw Funktion wird die Figur schliesslich dargestellt:

def draw():

figur.draw()

**Die Figur bewegen mit Hilfe der Pfeiltasten**

Tasten können mit keyboard[keys.NAME] angesprochen werden.

Mit einem if-statement kann überprüft werden, ob die Taste gedrückt wurde. Dies sollte in der upadte Funktion gemacht werden:

def update():

if keyboard[keys.RIGHT]:

# Bewegt den Hasen nach rechts

figur.x = figur.x + 3

if keyboard[keys.LEFT]:

# Bewegt den Hasen nach links

figur.x = figur.x - 3

if keyboard[keys.UP]:

# Bewegt den Hasen nach oben

figur.y = figur.y - 3

if keyboard[keys.DOWN]:

# Bewegt den Hasen nach unten

figur.y = figur.y + 3

**Die draw Funktion**

In der draw Funktion werden Dinge im Fenster „gezeichnet“ wie z.B. Spielfiguren oder Hintergründe.

def draw():

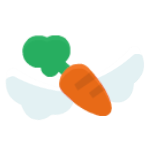
screen.clear()

screen.fill((240, 211, 242))

figur.draw()

Hier wird der screen (Bildschirm) zunächst geleert und danach der Hintergrund und die Figur dargestellt.

Der Hintergrund wird hier mit einer Farbe gefüllt (RGB).

****

****

**RGB Farben**

**Mausklicks**

Pygame Zero besitzt eine eigene Funktion für Mausklicks. Die **on\_mouse\_down** Funktion wird immer dann ausgeführt, wenn die linke Maustaste geklickt wird. Zum Bespiel kann man so per Klick das Bild der Spielfigur ändern:

def on\_mouse\_down(pos):

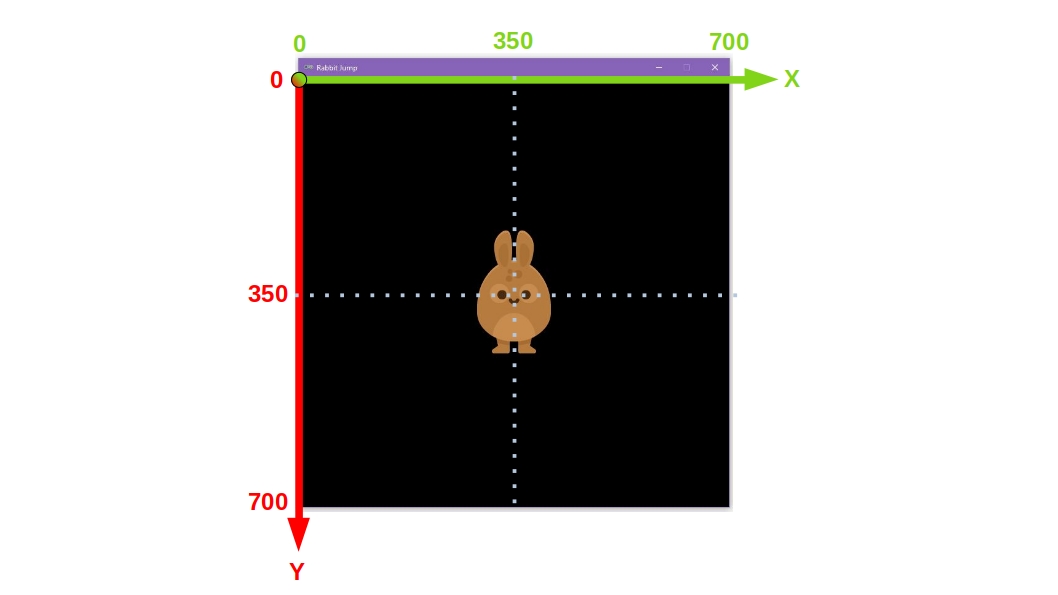
if figur.collidepoint(pos):

figur.image = 'bunny\_hurt'

A picture containing clipart, vector graphics

Description automatically generated**Icon

Description automatically generated**Icon

Description automatically generated

**Code Beispiel**

Ein komplettes Bespiel mit dem Code von diesem Cheat Sheet findest du auf:

<https://www.dropbox.com/home/Workshop_Game_Development/Beispiel%20Cheat%20Sheet>

**Das Koordinatensystem**

Alles was sich im Fenster befindet hat eine Gewisse Position im Koordinatensystem. Wir verwenden x für die horizontale und y für die vertikale Position. Angefangen wird bei 0 ganz oben links in der Ecke.

**Zeitereignisse**

Falls ein Ereignis nach einer bestimmten Zeit ausgelöst werden soll, kann die **clock.schedule\_unique** Funktion nützlich sein. Dabei kann man eine Funktion angeben, die man ausführen möchte und die Zeit bis dahin in Sekunden.

Im Folgenden Bespiel soll das Bild der Spielfigur nach 1 Sekunde zurück zum normalen Bild gewechselt werden nachdem die Maus geklickt wurde:

def on\_mouse\_down(pos):

if figur.collidepoint(pos):

figur.image = 'bunny\_hurt'

clock.schedule\_unique(set\_normal, 1.0)

def set\_normal():

figur.image = 'bunny'