

1 Zustand 1: Paddle

```
import pygame
import random

fensterbreite = 1000
fensterhoehe = 600

paddlebreite = 200
paddlehoehe = 20

ballradius = 20

leben = 3
punkte = 0

pygame.init()
pygame.display.set_caption("My first game: Solo pong")
clockobject = pygame.time.Clock()

fensterdimensionen = (fensterbreite, fensterhoehe)
fenster = pygame.display.set_mode(fensterdimensionen)

paddledimensionen = (paddlebreite, paddlehoehe)
paddlestartposition = ((fensterbreite - paddlebreite) // 2, fensterhoehe - paddlehoehe)
paddle = pygame.Rect(paddlestartposition, paddledimensionen)
pygame.draw.rect(surface=fenster, color = "green", rect = paddle)

paddle_bewegung = [0, 0]

while True: # game loop
    for ereignis in pygame.event.get(): # Schleife über alle passierten
        # events (Tastatur, Maus, etc.)
        if ereignis.type == pygame.QUIT: # Fensterschliess-Symbol angeklickt
            exit()
        elif ereignis.type == pygame.KEYDOWN: # Taste gedrückt
            if ereignis.key == pygame.K_q: # Taste "q"
                exit()
            if ereignis.key == pygame.K_LEFT: # Pfeiltaste nach links
                paddle_bewegung[0] = paddle_bewegung[0] - 1
            if ereignis.key == pygame.K_RIGHT:
                paddle_bewegung[0] = paddle_bewegung[0] + 1
        elif ereignis.type == pygame.KEYUP: # Taste losgelassen
            if ereignis.key == pygame.K_LEFT:
                paddle_bewegung[0] = paddle_bewegung[0] + 1
            if ereignis.key == pygame.K_RIGHT:
                paddle_bewegung[0] = paddle_bewegung[0] - 1

    fenster.fill("black")

    paddle = paddle.move(paddle_bewegung)
    # neuespielerposition = paddle.move(paddle_bewegung)
    # if 0 <= neuespielerposition.left and neuespielerposition.right <= fensterbreite:
    #     paddle = neuespielerposition
    pygame.draw.rect(surface=fenster, color="green", rect=paddle)

    clockobject.tick(400) # maximal 400 frames pro Sekunde (framerate)
    # je höher die Zahl, desto schneller das Spiel
    pygame.display.flip() # update des Fensterinhalts
```

2 Neues in Zustand 2: Ball dazu, vertikale Flugbahn

```
14 def neuerball():
15     ballmittelpunkt = [random.randrange(ballradius, fensterbreite - ballradius),
16                       fensterhoehe - paddlehoehe - ballradius]
```

```

17     return pygame.draw.circle(surface=fenster, color="red", center=ballmittelpunkt,
18                               radius=ballradius)

34 ball = neuerball()
35 ballrichtung = [0, -1]

56     ball = ball.move(ballrichtung)
57     if ball.top <= 0:
58         punkte = punkte + 1
59         ballrichtung[1] = - ballrichtung[1]
60     if ball.bottom >= fensterhoehe:
61         ballrichtung[1] = - ballrichtung[1]

63     pygame.draw.circle(surface=fenster, color="red", center=ball.center, radius=ballradius)

65     neuespielerposition = paddle.move(paddle_bewegung)
66     if 0 <= neuespielerposition.left and neuespielerposition.right <= fensterbreite:
67         paddle = neuespielerposition

```

3 Neues in Zustand 3: Schrift

```

14 def schreibe(s, x, y, farbe = "white", groesse = 32):
15     # font = pygame.font.Font('freesansbold.ttf', groesse)
16     font = pygame.font.Font(None, groesse)
17     text = font.render(s, True, farbe, "black")
18     text_rechteck = text.get_rect()
19     text_rechteck.center = (x, y)
20     fenster.blit(text, text_rechteck)

64     schreibe("Leben: " + str(leben), fensterbreite // 2, 32)
65     schreibe("Punkte: " + str(punkte), fensterbreite // 2, 64)

```

4 Neues in Zustand 4: Ball-Paddle-Check mit Konsequenzen

```

15     # font = pygame.font.Font('freesansbold.ttf', groesse)

71     if ball.bottom >= fensterhoehe - paddlehoehe:
72         if paddle.left <= ball.right and ball.left <= paddle.right:
73             ballrichtung[1] = - ballrichtung[1]
74         else:
75             leben = leben - 1
76             if leben <= 0:
77                 schreibe("Leben: " + str(leben), fensterbreite // 2, 32)
78                 schreibe("GAME OVER", fensterbreite // 2, fensterhoehe // 2, farbe = "red", groesse = 12)
79                 pygame.display.flip()
80                 pygame.time.delay(2000)

82         else:
83             ball = neuerball()
84             ballrichtung = [0, -1]

```

5 Neues in Zustand 5: Ball mit schräger Flugbahn

```

15     # font = pygame.font.Font('freesansbold.ttf', groesse)

43     ballrichtung = [1, -1]

69     if ball.left <= 0 or ball.right >= fensterbreite:
70         ballrichtung[0] = - ballrichtung[0]

84         ballrichtung = [1, -1]

```

6 Ab hier alle Dateien vollständig

7 Zustand 1: Paddle

```
import pygame
import random

fensterbreite = 1000
fensterhoehe = 600

paddlebreite = 200
paddlehoehe = 20

ballradius = 20

leben = 3
punkte = 0

pygame.init()
pygame.display.set_caption("My first game: Solo pong")
clockobject = pygame.time.Clock()

fensterdimensionen = (fensterbreite, fensterhoehe)
fenster = pygame.display.set_mode(fensterdimensionen)

paddledimensionen = (paddlebreite, paddlehoehe)
paddlestartposition = ((fensterbreite - paddlebreite) // 2, fensterhoehe - paddlehoehe)
paddle = pygame.Rect(paddlestartposition, paddledimensionen)
pygame.draw.rect(surface=fenster, color = "green", rect = paddle)

paddle_bewegung = [0, 0]

while True: # game loop
    for ereignis in pygame.event.get(): # Schleife über alle passierten
        # events (Tastatur, Maus, etc.)
        if ereignis.type == pygame.QUIT: # Fensterschliess-Symbol angeklickt
            exit()
        elif ereignis.type == pygame.KEYDOWN: # Taste gedrückt
            if ereignis.key == pygame.K_q: # Taste "q"
                exit()
            if ereignis.key == pygame.K_LEFT: # Pfeiltaste nach links
                paddle_bewegung[0] = paddle_bewegung[0] - 1
            if ereignis.key == pygame.K_RIGHT:
                paddle_bewegung[0] = paddle_bewegung[0] + 1
        elif ereignis.type == pygame.KEYUP: # Taste losgelassen
            if ereignis.key == pygame.K_LEFT:
                paddle_bewegung[0] = paddle_bewegung[0] + 1
            if ereignis.key == pygame.K_RIGHT:
                paddle_bewegung[0] = paddle_bewegung[0] - 1

    fenster.fill("black")

    paddle = paddle.move(paddle_bewegung)
    # neuespielerposition = paddle.move(paddle_bewegung)
    # if 0 <= neuespielerposition.left and neuespielerposition.right <= fensterbreite:
    #     paddle = neuespielerposition
    pygame.draw.rect(surface=fenster, color="green", rect=paddle)

    clockobject.tick(400) # maximal 400 frames pro Sekunde (framerate)
    # je höher die Zahl, desto schneller das Spiel
    pygame.display.flip() # update des Fensterinhalts
```

8 Zustand 2: Ball dazu, vertikale Flugbahn

```
import pygame
import random

fensterbreite = 1000
fensterhoehe = 600

paddlebreite = 200
paddlehoehe = 20

ballradius = 20

leben = 3
punkte = 0

def neuerball():
    ballmittelpunkt = [random.randrange(ballradius, fensterbreite - ballradius),
                      fensterhoehe - paddlehoehe - ballradius]
    return pygame.draw.circle(surface=fenster, color="red", center=ballmittelpunkt,
                              radius=ballradius)

pygame.init()
pygame.display.set_caption("My first game: Solo pong")
clockobject = pygame.time.Clock()

fensterdimensionen = (fensterbreite, fensterhoehe)
fenster = pygame.display.set_mode(fensterdimensionen)

paddledimensionen = (paddlebreite, paddlehoehe)
paddlestartposition = ((fensterbreite - paddlebreite) // 2, fensterhoehe - paddlehoehe)
paddle = pygame.Rect(paddlestartposition, paddledimensionen)
pygame.draw.rect(surface=fenster, color = "green", rect = paddle)

paddle_bewegung = [0, 0]

ball = neuerball()
ballrichtung = [0, -1]

while True:
    for ereignis in pygame.event.get():
        if ereignis.type == pygame.QUIT:
            exit()
        elif ereignis.type == pygame.KEYDOWN:
            if ereignis.key == pygame.K_q:
                exit()
            if ereignis.key == pygame.K_LEFT:
                paddle_bewegung[0] = paddle_bewegung[0] - 1
            if ereignis.key == pygame.K_RIGHT:
                paddle_bewegung[0] = paddle_bewegung[0] + 1
        elif ereignis.type == pygame.KEYUP:
            if ereignis.key == pygame.K_LEFT:
                paddle_bewegung[0] = paddle_bewegung[0] + 1
            if ereignis.key == pygame.K_RIGHT:
                paddle_bewegung[0] = paddle_bewegung[0] - 1

    fenster.fill("black")

    ball = ball.move(ballrichtung)
    if ball.top <= 0:
        punkte = punkte + 1
        ballrichtung[1] = - ballrichtung[1]
    if ball.bottom >= fensterhoehe:
        ballrichtung[1] = - ballrichtung[1]
```

```

pygame.draw.circle(surface=fenster, color="red", center=ball.center, radius=ballradius)

neuespielerposition = paddle.move(paddle_bewegung)
if 0 <= neuespielerposition.left and neuespielerposition.right <= fensterbreite:
    paddle = neuespielerposition
pygame.draw.rect(surface=fenster, color="green", rect=paddle)

clockobject.tick(400)
pygame.display.flip()

```

9 Zustand 3: Schrift

```

import pygame
import random

fensterbreite = 1000
fensterhoehe = 600

paddlebreite = 200
paddlehoehe = 20

ballradius = 20

leben = 3
punkte = 0

def schreibe(s, x, y, farbe = "white", groesse = 32):
    # font = pygame.font.Font('freesansbold.ttf', groesse)
    font = pygame.font.Font(None, groesse)
    text = font.render(s, True, farbe, "black")
    text_rechteck = text.get_rect()
    text_rechteck.center = (x, y)
    fenster.blit(text, text_rechteck)

def neuerball():
    ballmittelpunkt = [random.randrange(ballradius, fensterbreite - ballradius),
                      fensterhoehe - paddlehoehe - ballradius]
    return pygame.draw.circle(surface=fenster, color="red", center=ballmittelpunkt,
                              radius=ballradius)

pygame.init()
pygame.display.set_caption("My first game: Solo pong")
clockobject = pygame.time.Clock()

fensterdimensionen = (fensterbreite, fensterhoehe)
fenster = pygame.display.set_mode(fensterdimensionen)

paddledimensionen = (paddlebreite, paddlehoehe)
paddlestartposition = ((fensterbreite - paddlebreite) // 2, fensterhoehe - paddlehoehe)
paddle = pygame.Rect(paddlestartposition, paddledimensionen)
pygame.draw.rect(surface=fenster, color = "green", rect = paddle)

paddle_bewegung = [0, 0]

ball = neuerball()
ballrichtung = [0, -1]

while True:
    for ereignis in pygame.event.get():
        if ereignis.type == pygame.QUIT:
            exit()
        elif ereignis.type == pygame.KEYDOWN:
            if ereignis.key == pygame.K_q:
                exit()
            if ereignis.key == pygame.K_LEFT:

```

```

        paddle_bewegung[0] = paddle_bewegung[0] - 1
    if ereignis.key == pygame.K_RIGHT:
        paddle_bewegung[0] = paddle_bewegung[0] + 1
    elif ereignis.type == pygame.KEYUP:
        if ereignis.key == pygame.K_LEFT:
            paddle_bewegung[0] = paddle_bewegung[0] + 1
        if ereignis.key == pygame.K_RIGHT:
            paddle_bewegung[0] = paddle_bewegung[0] - 1

fenster.fill("black")

schreibe("Leben: " + str(leben), fensterbreite // 2, 32)
schreibe("Punkte: " + str(punkte), fensterbreite // 2, 64)

ball = ball.move(ballrichtung)
if ball.top <= 0:
    punkte = punkte + 1
    ballrichtung[1] = - ballrichtung[1]
if ball.bottom >= fensterhoehe:
    ballrichtung[1] = - ballrichtung[1]

pygame.draw.circle(surface=fenster, color="red", center=ball.center, radius=ballradius)

neuespielerposition = paddle.move(paddle_bewegung)
if 0 <= neuespielerposition.left and neuespielerposition.right <= fensterbreite:
    paddle = neuespielerposition
pygame.draw.rect(surface=fenster, color="green", rect=paddle)

clockobject.tick(400)
pygame.display.flip()

```

10 Zustand 4: Ball-Paddle-Check mit Konsequenzen

```

import pygame
import random

fensterbreite = 1000
fensterhoehe = 600

paddlebreite = 200
paddlehoehe = 20

ballradius = 20

leben = 3
punkte = 0

def schreibe(s, x, y, farbe = "white", groesse = 32):
    # font = pygame.font.Font('freesansbold.ttf', groesse)
    font = pygame.font.Font(None, groesse)
    text = font.render(s, True, farbe, "black")
    text_rechteck = text.get_rect()
    text_rechteck.center = (x, y)
    fenster.blit(text, text_rechteck)

def neuerball():
    ballmittelpunkt = [random.randrange(ballradius, fensterbreite - ballradius),
                      fensterhoehe - paddlehoehe - ballradius]
    return pygame.draw.circle(surface=fenster, color="red", center=ballmittelpunkt,
                              radius=ballradius)

pygame.init()
pygame.display.set_caption("My first game: Solo pong")
clockobject = pygame.time.Clock()

```

```

fensterdimensionen = (fensterbreite, fensterhoehe)
fenster = pygame.display.set_mode(fensterdimensionen)

paddledimensionen = (paddlebreite, paddlehoehe)
paddlestartposition = ((fensterbreite - paddlebreite) // 2, fensterhoehe - paddlehoehe)
paddle = pygame.Rect(paddlestartposition, paddledimensionen)
pygame.draw.rect(surface=fenster, color = "green", rect = paddle)

paddle_bewegung = [0, 0]

ball = neuerball()
ballrichtung = [0, -1]

while True:
    for ereignis in pygame.event.get():
        if ereignis.type == pygame.QUIT:
            exit()
        elif ereignis.type == pygame.KEYDOWN:
            if ereignis.key == pygame.K_q:
                exit()
            if ereignis.key == pygame.K_LEFT:
                paddle_bewegung[0] = paddle_bewegung[0] - 1
            if ereignis.key == pygame.K_RIGHT:
                paddle_bewegung[0] = paddle_bewegung[0] + 1
        elif ereignis.type == pygame.KEYUP:
            if ereignis.key == pygame.K_LEFT:
                paddle_bewegung[0] = paddle_bewegung[0] + 1
            if ereignis.key == pygame.K_RIGHT:
                paddle_bewegung[0] = paddle_bewegung[0] - 1

    fenster.fill("black")

    schreibe("Leben: " + str(leben), fensterbreite // 2, 32)
    schreibe("Punkte: " + str(punkte), fensterbreite // 2, 64)

    ball = ball.move(ballrichtung)
    if ball.top <= 0:
        punkte = punkte + 1
        ballrichtung[1] = - ballrichtung[1]
    if ball.bottom >= fensterhoehe - paddlehoehe:
        if paddle.left <= ball.right and ball.left <= paddle.right:
            ballrichtung[1] = - ballrichtung[1]
        else:
            leben = leben - 1
            if leben <= 0:
                schreibe("Leben: " + str(leben), fensterbreite // 2, 32)
                schreibe("GAME OVER", fensterbreite // 2, fensterhoehe // 2, farbe = "red", groesse = 128)
                pygame.display.flip()
                pygame.time.delay(2000)
                exit()
            else:
                ball = neuerball()
                ballrichtung = [0, -1]

    pygame.draw.circle(surface=fenster, color="red", center=ball.center, radius=ballradius)

    neuespielerposition = paddle.move(paddle_bewegung)
    if 0 <= neuespielerposition.left and neuespielerposition.right <= fensterbreite:
        paddle = neuespielerposition
    pygame.draw.rect(surface=fenster, color="green", rect=paddle)

    clockobject.tick(400)
    pygame.display.flip()

```

11 Zustand 5: Ball mit schräger Flugbahn

```
import pygame
import random

fensterbreite = 1000
fensterhoehe = 600

paddlebreite = 200
paddlehoehe = 20

ballradius = 20

leben = 3
punkte = 0

def schreibe(s, x, y, farbe = "white", groesse = 32):
    # font = pygame.font.Font('freesansbold.ttf', groesse)
    font = pygame.font.Font(None, groesse)
    text = font.render(s, True, farbe, "black")
    text_rechteck = text.get_rect()
    text_rechteck.center = (x, y)
    fenster.blit(text, text_rechteck)

def neuerball():
    ballmittelpunkt = [random.randrange(ballradius, fensterbreite - ballradius),
                      fensterhoehe - paddlehoehe - ballradius]
    return pygame.draw.circle(surface=fenster, color="red", center=ballmittelpunkt,
                              radius=ballradius)

pygame.init()
pygame.display.set_caption("My first game: Solo pong")
clockobject = pygame.time.Clock()

fensterdimensionen = (fensterbreite, fensterhoehe)
fenster = pygame.display.set_mode(fensterdimensionen)

paddledimensionen = (paddlebreite, paddlehoehe)
paddlestartposition = ((fensterbreite - paddlebreite) // 2, fensterhoehe - paddlehoehe)
paddle = pygame.Rect(paddlestartposition, paddledimensionen)
pygame.draw.rect(surface=fenster, color = "green", rect = paddle)

paddle_bewegung = [0, 0]

ball = neuerball()
ballrichtung = [1, -1]

while True:
    for ereignis in pygame.event.get():
        if ereignis.type == pygame.QUIT:
            exit()
        elif ereignis.type == pygame.KEYDOWN:
            if ereignis.key == pygame.K_q:
                exit()
            if ereignis.key == pygame.K_LEFT:
                paddle_bewegung[0] = paddle_bewegung[0] - 1
            if ereignis.key == pygame.K_RIGHT:
                paddle_bewegung[0] = paddle_bewegung[0] + 1
        elif ereignis.type == pygame.KEYUP:
            if ereignis.key == pygame.K_LEFT:
                paddle_bewegung[0] = paddle_bewegung[0] + 1
            if ereignis.key == pygame.K_RIGHT:
                paddle_bewegung[0] = paddle_bewegung[0] - 1

fenster.fill("black")
```



```

schreibe("Leben: " + str(leben), fensterbreite // 2, 32)
schreibe("Punkte: " + str(punkte), fensterbreite // 2, 64)

ball = ball.move(ballrichtung)
if ball.top <= 0:
    punkte = punkte + 1
    ballrichtung[1] = - ballrichtung[1]
if ball.left <= 0 or ball.right >= fensterbreite:
    ballrichtung[0] = - ballrichtung[0]
if ball.bottom >= fensterhoehe - paddlehoehe:
    if paddle.left <= ball.right and ball.left <= paddle.right:
        ballrichtung[1] = - ballrichtung[1]
    else:
        leben = leben - 1
        if leben <= 0:
            schreibe("Leben: " + str(leben), fensterbreite // 2, 32)
            schreibe("GAME OVER", fensterbreite // 2, fensterhoehe // 2, farbe = "red", groesse = 128)
            pygame.display.flip()
            pygame.time.delay(2000)
            exit()
        else:
            ball = neuerball()
            ballrichtung = [1, -1]

pygame.draw.circle(surface=fenster, color="red", center=ball.center, radius=ballradius)

neuespielerposition = paddle.move(paddle_bewegung)
if 0 <= neuespielerposition.left and neuespielerposition.right <= fensterbreite:
    paddle = neuespielerposition
pygame.draw.rect(surface=fenster, color="green", rect=paddle)

clockobject.tick(400)
pygame.display.flip()

```