



Programmieren  
lernen mit Python

# Snake-Spiel (3/3)

**Kira Grammel, Nina Ihde, Sebastian Serth & Selina Reinhard**  
Hasso-Plattner-Institut  
Universität Potsdam

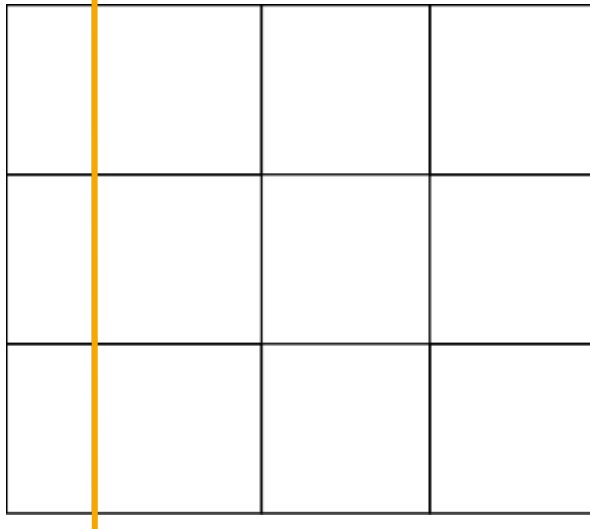
# Kollision mit Fensterrand

```

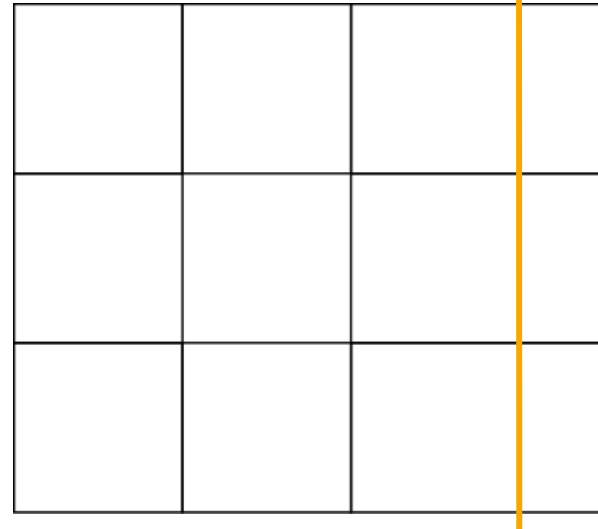
1 def checke_kollision_mit_fensterrand():
2     if kopf.xcor() < -190 or kopf.xcor() > 190 or
       kopf.ycor() < -190 or kopf.ycor() > 190:
3         # Kopf in der Mitte platzieren
4         # Richtung auf "stop" setzen
5         segmente_entfernen()
6         # Ausgabe, dass Spielrunde vorbei ist

```

(-190, \_)



(190, \_)



# Kollision mit Fensterrand

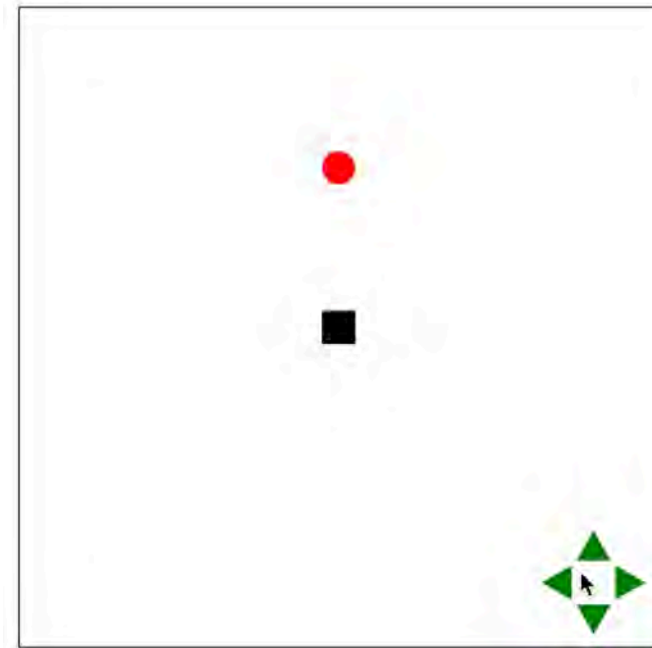
---

```
1 def checke_kollision_mit_fensterrand():
2     if kopf.xcor() < -190 or kopf.xcor() > 190 or
        kopf.ycor() < -190 or kopf.ycor() > 190:
3         spiel_neustarten()
```

```
1 def spiel_neustarten():
2     # Kopf in der Mitte platzieren
3     # Richtung auf "stop" setzen
4     segmente_entfernen()
5     # Ausgabe, dass Spielrunde vorbei ist
```

# Kollision mit Fensterrand

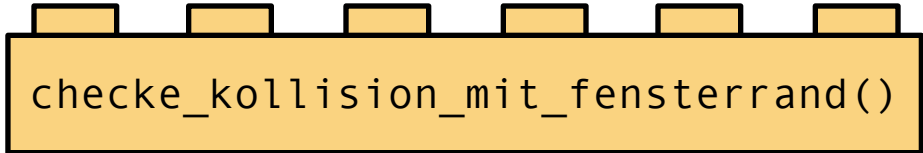
```
1 def checke_kollision_mit_fensterrand():
2     if kopf.xcor() < -190 or kopf.xcor() > 190 or
        kopf.ycor() < -190 or kopf.ycor() > 190:
3         spiel_neustarten()
```



Leider hast Du das Spiel verloren.

# Kollision mit Fensterrand

```
1 def checke_kollision_mit_fensterrand():  
2     if kopf.xcor() < -190 or kopf.xcor() > 190 or  
        kopf.ycor() < -190 or kopf.ycor() > 190:  
3         spiel_neustarten()
```



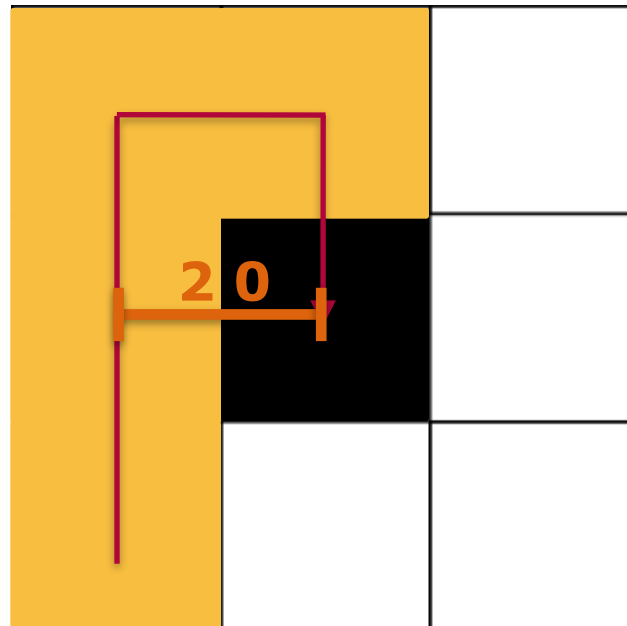
checke\_kollision\_mit\_fensterrand()

# Kollision von Kopf mit Körper

```

1 def checke_kollision_mit_segmenten():
2     for segment in segmente:
3         if segment.distance(kopf) < 20:
4             spiel_neustarten()

```



# Kollision von Kopf mit Körper

---

```
1 def checke_kollision_mit_segmenten():  
2     for segment in segmente:  
3         if segment.distance(kopf) < 20:  
4             spiel_neustarten()
```



```
checke_kollision_mit_segmenten()
```

# Bewegung des Schlangenkörpers

```
1 def koerper_bewegen():  
2     for index in range(len(segmente) - 1, 0, -1):
```

## **range(start, ende, schritt)**

- Gibt Werte von **start** bis **ende** (nicht enthalten) in **schritt** zurück
- **start** ist optional und wird standardmäßig auf **0** gesetzt
- **schritt** ist optional und wird standardmäßig auf **1** gesetzt
- Nur **ende** wird zwingend benötigt: `range(5)`

```
1 for index in range(2, 5, 1):  
2     print(index)
```

2  
3  
4

```
1 for index in range(5, 2, -1):  
2     print(index)
```

5  
4  
3

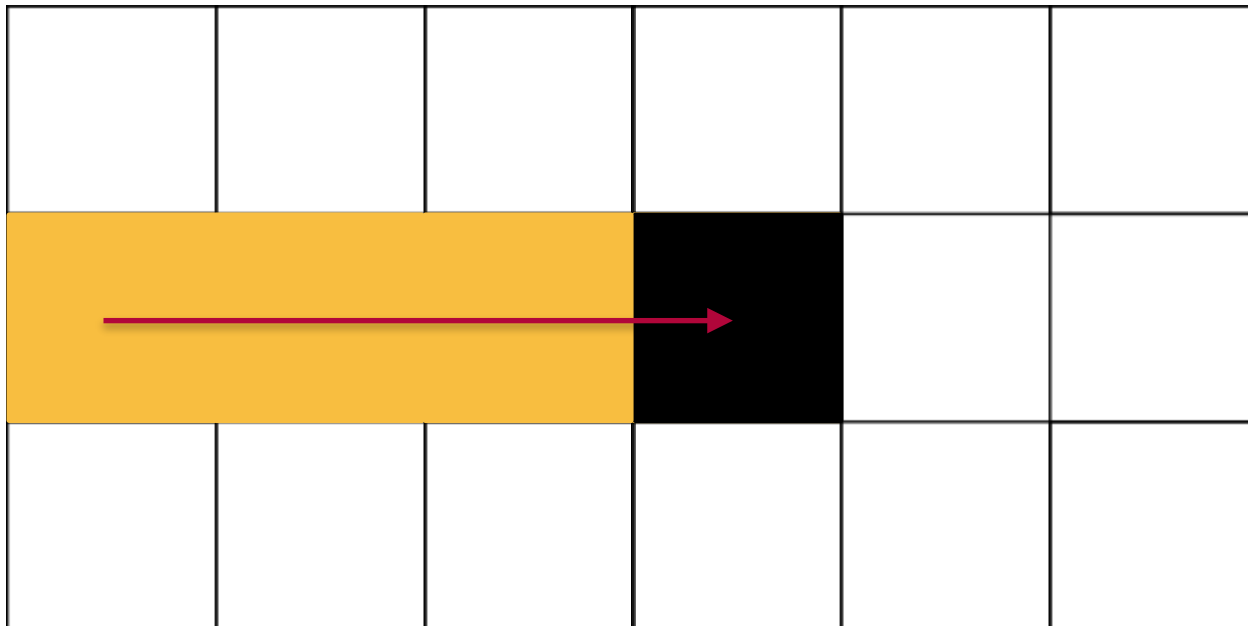


# Bewegung des Schlangenkörpers

```

1 def koerper_bewegen():
2     for index in range(len(segmente) - 1, 0, -1):
3         # Bewege segmente[index] an segmente[index - 1]
4
5
6

```

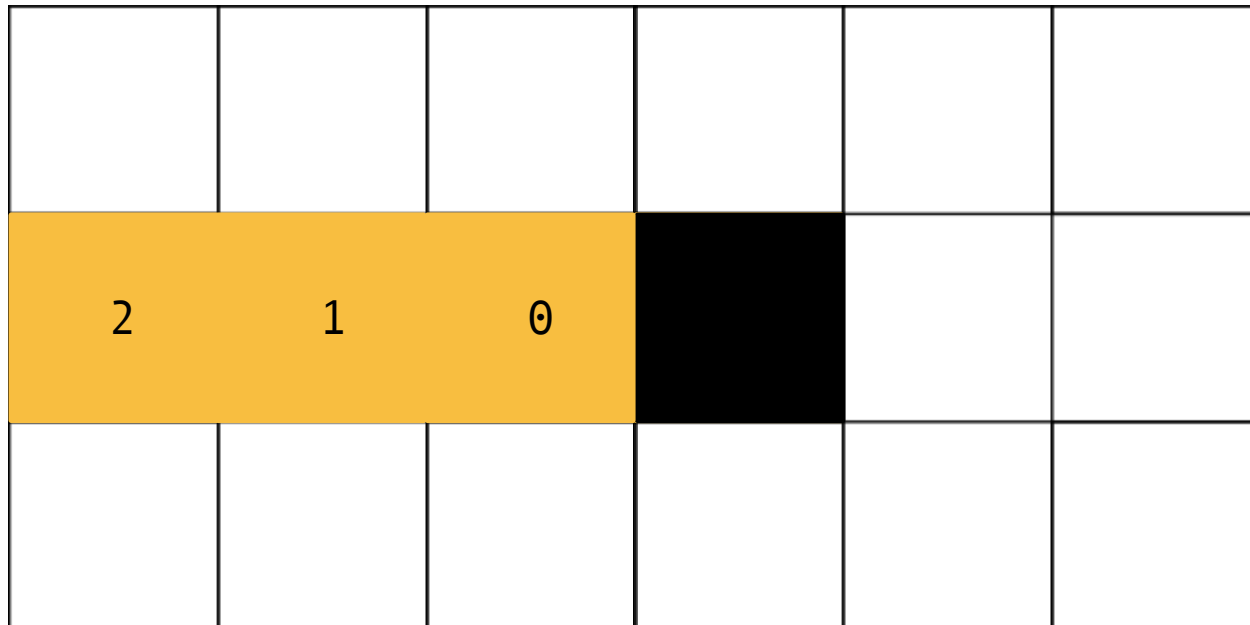


# Bewegung des Schlangenkörpers

```

1 def koerper_bewegen():
2     for index in range(len(segmente) - 1, 0, -1):
3         # Bewege segmente[index] an segmente[index - 1]
4
5
6

```

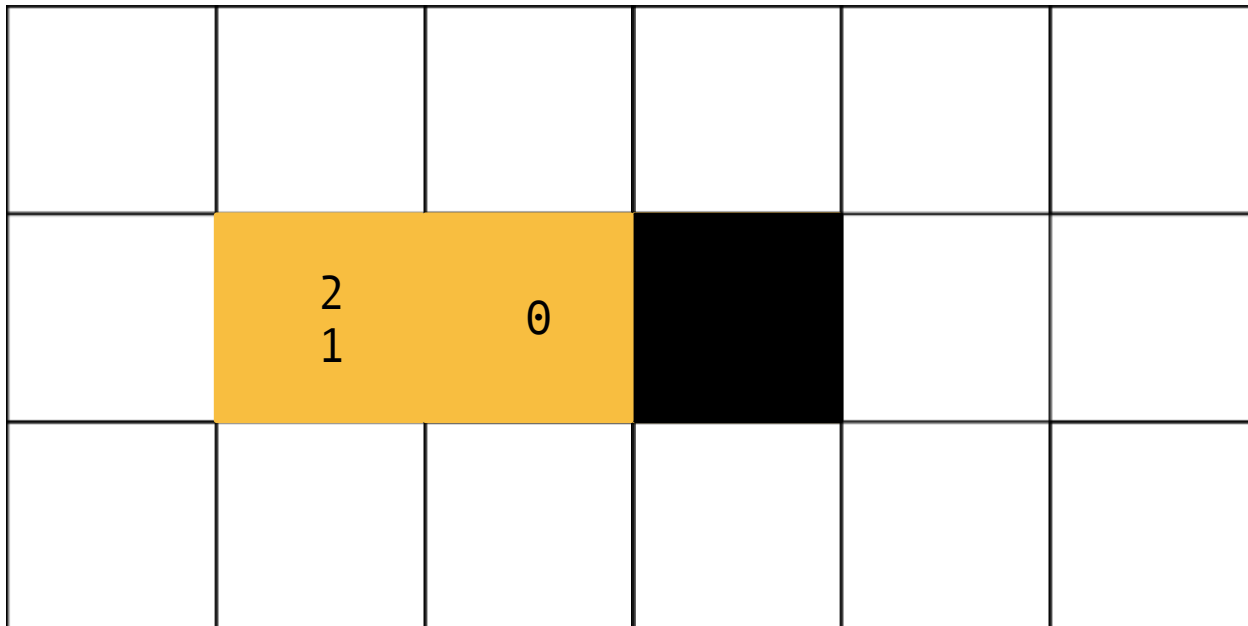


# Bewegung des Schlangenkörpers

```

1 def koerper_bewegen():
2     for index in range(len(segmente) - 1, 0, -1):
3         # Bewege segmente[index] an segmente[index - 1]
4
5
6

```

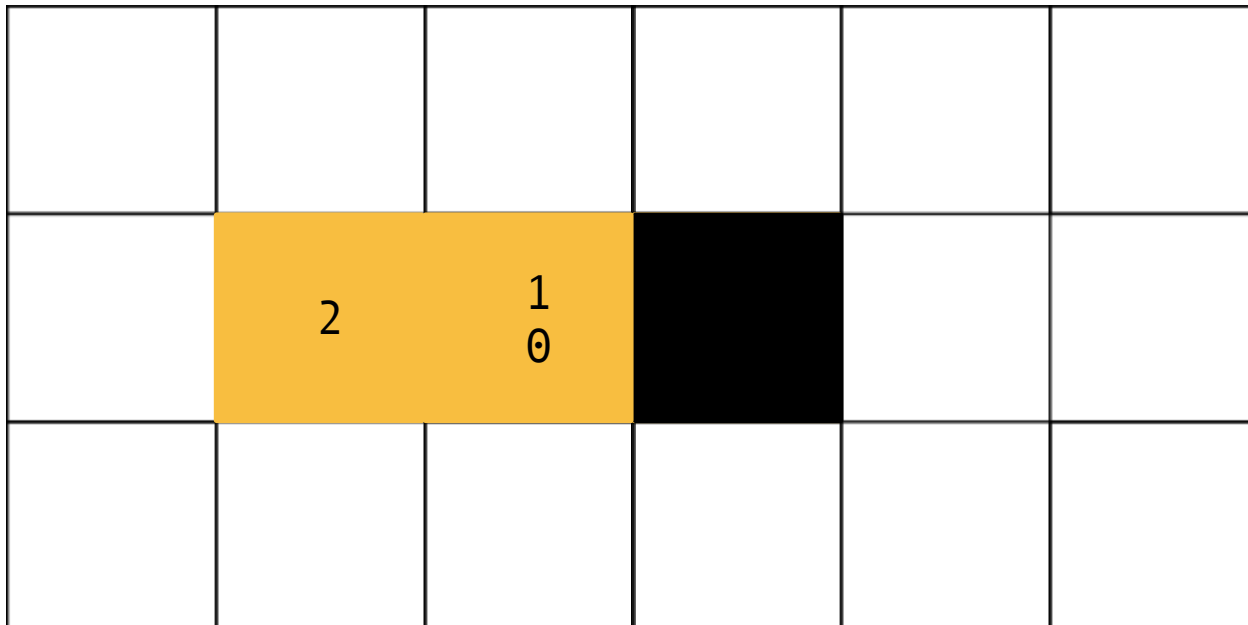


# Bewegung des Schlangenkörpers

```

1 def koerper_bewegen():
2     for index in range(len(segmente) - 1, 0, -1):
3         # Bewege segmente[index] an segmente[index - 1]
4
5     # Überprüfe, ob Schlange nicht nur aus Kopf besteht
6     # Wenn, dann bewege erstes Segment zum Kopf

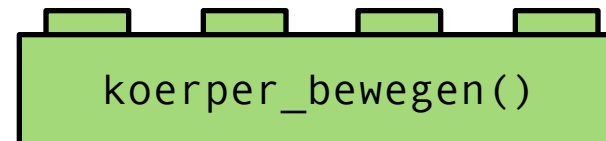
```

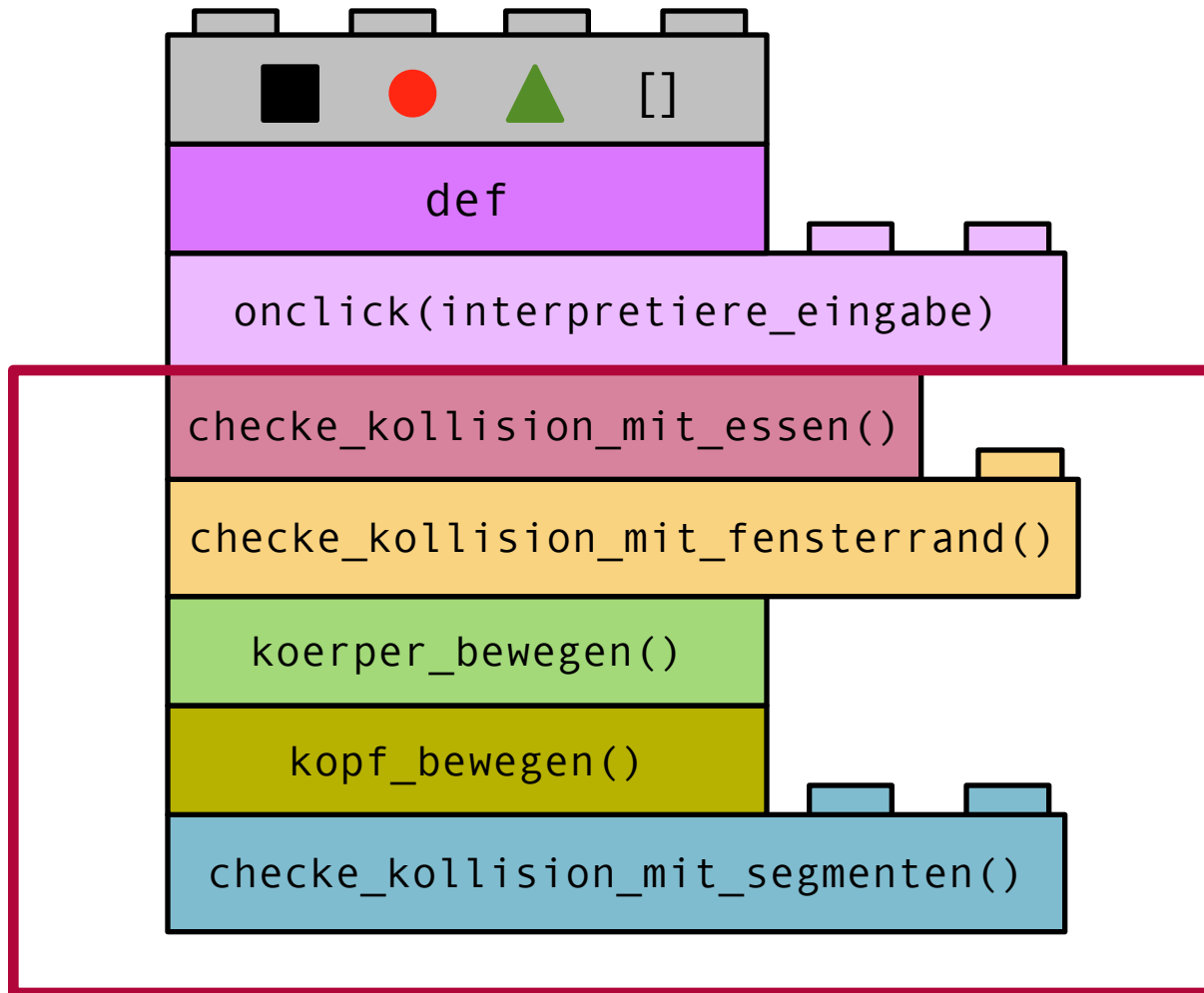


# Bewegung des Schlangenkörpers

---

```
1 def koerper_bewegen():
2     for index in range(len(segmente) - 1, 0, -1):
3         # Bewege segmente[index] an segmente[index - 1]
4
5     # Überprüfe, ob Schlange nicht nur aus Kopf besteht
6     # Wenn, dann bewege erstes Segment zum Kopf
```





# Sich wiederholende Spiellogik

---

```
1 def wiederhole_spiellogik():  
2     checke_kollision_mit_essen()  
3     checke_kollision_mit_fensterrand()  
4     koerper_bewegen()  
5     kopf_bewegen()  
6     checke_kollision_mit_segmenten()
```

# Sich wiederholende Spiellogik

---

```
1 def wiederhole_spiellogik():  
2     checke_kollision_mit_essen()  
3     checke_kollision_mit_fensterrand()  
4     koerper_bewegen()  
5     kopf_bewegen()  
6     checke_kollision_mit_segmenten()
```



wiederhole\_spiellogik()



# Gesamter Spielablauf

