

# Modell anlegen



0

# Modell anlegen

Mögliche Aktionen überlegen



bewege hoch

tue nichts

bewege runter

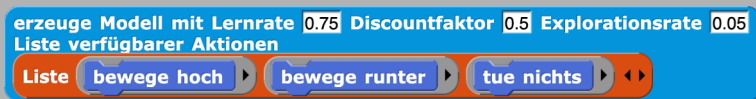


bewege hoch

Rechtsklick > Umringen



Modell erzeugen



Modell speichern

Neue Variable

modell

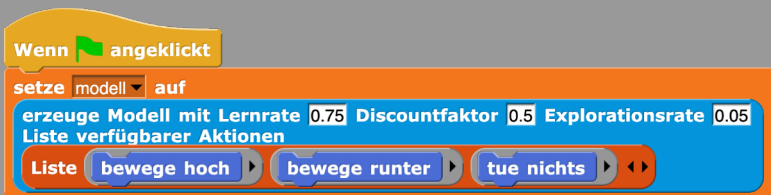
setze modell auf

Neue Variable  
anlegen

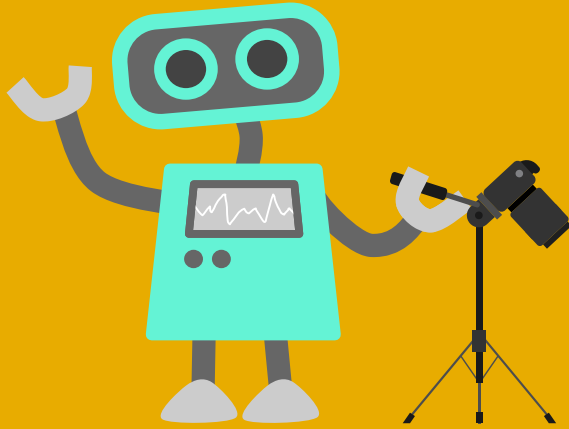
Name vergeben

Modell in Variable  
speichern

Ins Programm einbauen



# Zustand erfassen



1

# Zustand erfassen

## Zustand erfassen

Was ist wichtig, welche Infos braucht der Agent?

Y Position des Balls

Abstand

Y Position des Schlägers

y-Position - y-Position von Ball

## Tipps

Je größer die Anzahl der Zustände, desto länger dauert das Training

y-Position - y-Position von Ball

statt

Liste y-Position y-Position von Ball

Oft reichen wenige Informationen

Eine Liste kann mehrere Einträge enthalten, z.B.



y-Position - y-Position von Ball

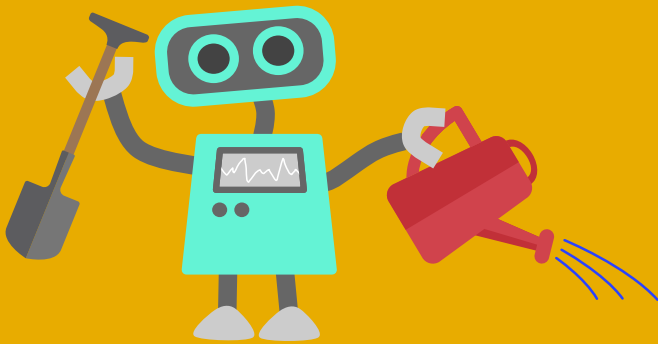
/ 10 gerundet

Werte zu Bereichen zusammenfassen

## Ins Programm einbauen



# Aktion auswählen und ausführen



2

# Aktion durchführen

Beste Aktion ermitteln

Zustand aus Karte 2

**führe beste Aktion aus ▶**

Modell aus Karte 1

Die Variable enthält die beste Aktion, z.B.

**beste Aktion**

**bewege hoch ▶**

Beste Aktion ausführen

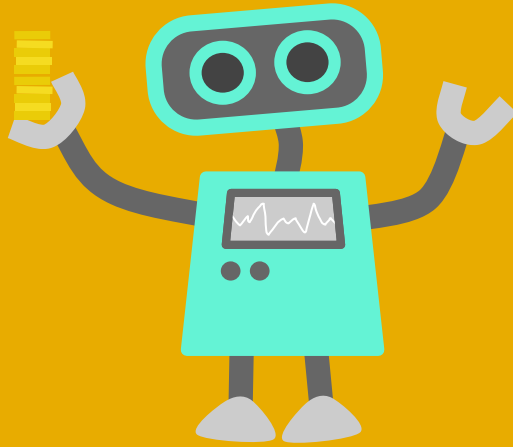
Die Variable berichtet einen Block und kann daher so ausgeführt werden

**führe beste Aktion aus ▶**

Ins Programm einbauen



# Belohnung erhalten



3

# Belohnung erhalten

Belohnung bestimmen

Wann sollten wir den Agenten belohnen,  
wann bestrafen?

Statt

wenn dann sonst

kann auch ein eigener  
Block erzeugt werden.

Belohnung

wenn **berühre** Ball ? dann 2 sonst 0

Tipps

Ball gelangt an rechten Rand

Bedingungen  
verschachteln

wenn **berühre** Ball ? dann 2 sonst  
wenn **x-Position** von Ball > 219 dann -5 sonst 0

Bestrafung abhängig von Entfernung zu Ball

wenn **berühre** Ball ? dann 2 sonst  
wenn **x-Position** von Ball > 219 dann  $-5 \times$  Entfernung zu Ball  
sonst 0

Belohnung und Bestrafung dynamisch  
gestalten

Ins Programm einbauen

**Skriptvariablen** beste Aktion Belohnung Zustand

**fortlaufend**

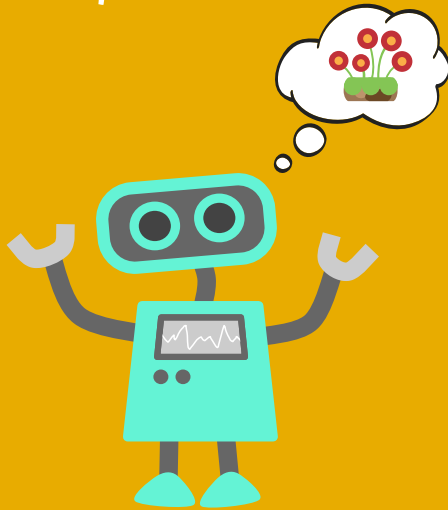
**setze** Belohnung auf

wenn **berühre** Ball ? dann 2 sonst

wenn **x-Position** von Ball > 219 dann  $-5 \times$  Entfernung zu Ball  
sonst 0



# Strategie anpassen

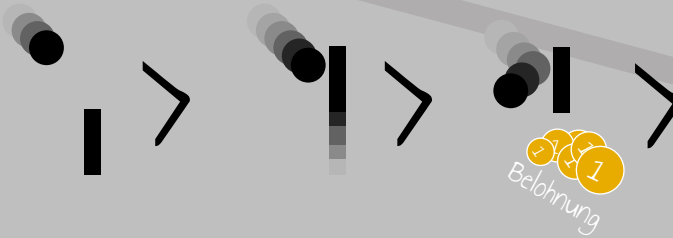


4

# Strategie anpassen

Informationen verarbeiten

alter Zustand      beste Aktion      <sup>neuer</sup>  
~~alter Zustand~~



Lernen durch  
Aktualisieren  
des Modells

Modell aktualisieren



Ins Programm einbauen

