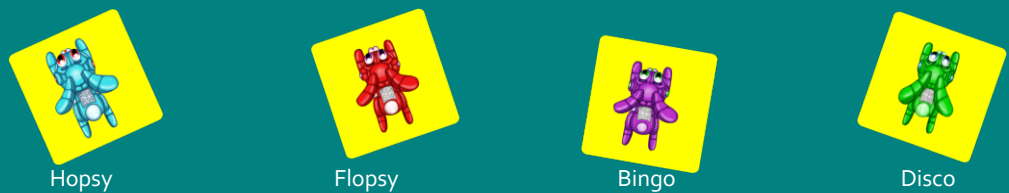


# Coder Bunnyz

Programmierspaß für junge Coder



## Anleitung

v1.0



## Inhalt

Einleitung	1
Bestandteile des Spiels	1
Aufbau	3
Spielablauf	8
Spielarten	11
1. Einfaches Coding	12
2. Erweitertes Coding	14
3. Strategie & Code-it!	17
4. Coding für Profis	18
Fragen & Antworten	24
Kurzübersicht	28
1. Marken (Tokens)	28
2. Karten (Cards)	29
3. Labyrinth (Maze)	30

## Einleitung

Willkommen auf der Coder Bunnyz Farm, dem Zuhause unserer besonderen Bunnyz: Hopsy, Flopsy, Bingo und Disco! Lerne zu „coden“ (das ist der Englische Begriff für programmieren), während du deinen Bunny zu einem lustigen Ziel führst - dem Jahrmarkt, dem Park, der Schule oder dem Zoo. Füttere sie unterwegs mit Karotten (ihr Lieblingsfutter), damit sie genug Energie haben, um das Ziel zu erreichen. Achte auf Pfützen und Zäune! In diesem spannenden Abenteuer auf dem Bauernhof musst Du Deinem Hasen helfen, über Pfützen zu springen und Zäune zu umgehen.

## Bestandteile des Spiels

Das Spiel beinhaltet: **1)** Coder Bunnyz Code Karten, **2)** Coder Bunnyz Marken, **3)** das Coder Bunnyz Spielbrett, und **4)** einen Würfel.

### 1. Coder Bunnyz Code Karten

Das Spiel enthält 4 farbige Kartendecks - Türkis, Rot, Lila und Grün:



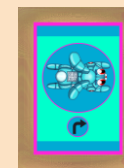
Jedes Deck hat 54 Karten. Die Basiskarten beinhalten:



**Vorwärts  
bewegen (18)**  
(Move Forward)



**links  
drehen (9)**  
(Turn Left)



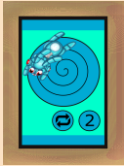
**rechts  
drehen (9)**  
(Turn Right)



**springen (6)**  
(Jump)



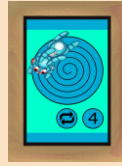
Die fortgeschrittenen Karten in jedem Stapel enthalten:



**Wiederholen 2 (2)**  
(Repeat 2)



**Wiederholen 3 (3)**  
(Repeat 3)



**Wiederholen 4 (4)**  
(Repeat 4)



**Funktion (f)**  
(Function)



## 2. Coder Bunnyz Spielmarken (Tokens)

Das Spiel enthält 6 Arten von Spielsteinen - Bunny, Bug-Fixit! (Fehlerbehebung), Karotten, Ziel, Zaun und Pfütze:



4 Bunny Marken  
(einer von jeder Farbe)



4 Bug-Fix Marken  
(einer von jeder Farbe)



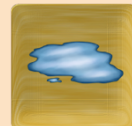
4 Karotten Marken  
(eine von jeder Farbe)



4 Ziel Marken  
(Jahrmkt, Park, Schule, Zoo)



16 Zaun Marken



8 Pfützen Marken

## 3. Coder Bunnyz Spielbrett

Das Spiel enthält ein (anpassbares) Spielbrett:



Füge dem Spielbrett Spielmarken hinzu, um es anzupassen, wenn Du das Spiel einrichtest. Keine Sorge: wie das genau funktioniert erfährst Du später im Kapitel Aufbau (Seite 3)

Auf Seite 30 und auf [coderbunnyz.com](http://coderbunnyz.com) findest Du Ideen, wie Du verschiedene Labyrinth benutzen kannst, um das Spielbrett anzupassen.

## 4. Würfel

Das Spiel enthält einen 6-seitigen Würfel mit spezieller Nummerierung: 1-1-2-2-3-3.

## Aufbau

### 1. Spielteile auswählen:

Jeder Spieler wählt einen Bunny (Spielmarke) und nimmt die passende Farbe des BugFixit! Spielsteins, des Karotten-Steins und des Coder Bunnyz Code-Kartensatzes.


### Hinweis:

- Für Anfänger:
  - legt die Wiederholungs- (↺) und Funktionskarten (f) zunächst beiseite. Diese Karten werden erst benötigt, wenn Ihr mehr Erfahrung mit dem Spiel habt.
  - Jeder Spieler sollte zu Beginn nur die Basiskarten erhalten: Vorwärts bewegen (↑), links drehen (↶), rechts drehen (↷) und springen (↲).
- Entferne die farbigen Bunnyz und andere Spielmarken, die nicht in Gebrauch sind, für die Spieler die nicht am Spiel teilnehmen

- Wähle das Ziel aus: Der Spieler, der die höchste Zahl würfelt, darf das Ziel bestimmen: Jahrmarkt (Carnival), Park, Schule (School) oder Zoo. (Bei Gleichstand wird so lange gewürfelt, bis der Gleichstand gebrochen ist).



## 2. Spielteile anordnen:

- Platziere die ausgewählte Ziel-Marke in der Mitte des Spielbretts
- Platziere den Hasen (Bunny) in seiner farbigen Heimatbasis (die farbigen äußeren Quadrate des Spielbretts). Innerhalb der farbigen Heimatbasis kommt der Hase jedes Spielers auf das Feld, das dem gewählten Ziel entspricht. Ordne die farbigen Code-Kartensets in 4 Stapeln an: 
- Die Karotte eines jeden Spielers wird gleichmäßig auf das Spielbrett gelegt und nicht in die Homepage eines Spielers.

-Dieser Platz ist absichtlich frei-

Bitte blättere um, um ein Spielbrett zu sehen, dass bereits für ein Spiel eingerichtet wurde. (übernächste Seite)

## Aufbau für ein Spiel mit 4 Spielern

Hopsy



Flopsy



Bingo



Disco



Das sind unsere 4 Bunnyz (Hasen)

### Coder Bunnyz Spielkarten (Roter Spieler)

Diese werden in 4 Stapeln angeordnet:



### Bug-Fixit! (Fehlerbehebung) (Roter Spieler)



### Karottenstein (Roter Spieler)

Jeder Spieler muss seine farbige Karotte essen, bevor er zum Zielort fährt. Achte bitte darauf, dass die 4 Karotten gleichmäßig auf dem Spielfeld verteilt werden: jeder Spieler muss die gleiche Strecke zurücklegen, um seine Karotte zu erreichen.



#### Tips:

- Für das erste Spiel: die Karotten-Marken in der Nähe des Zielortes platzieren (einfache Einrichtung). Versuche schwierigere Setups erst nach ein paar Runden.
- Anfänger sollten in Betracht ziehen zunächst ohne die Karotten-Marken zu spielen: gewinnen indem Du das Ziel direkt erreichst
- Bei Spielern mit unterschiedlichen Spielstärken: gib dem fortgeschrittenen Spieler eine schwierigere Aufstellung (in diesem Fall werden die Karotten-Spielmarken nicht gleichmäßig auf dem Brett platziert).

### Heimatbasis (Roter Spieler)

(farbiger äußerer Rand des Spielfeldes)

Jeder Bunny startet das Spiel in der Heimatbasis seiner Farbe. Bunnyz dürfen sich innerhalb ihrer eigenen Home Base bewegen, dürfen aber nicht in die Home Base einer anderen Farbe gehen.

### Bunny Spielmarke (Roter Spieler)

- Dies ist die Spielmarke, die immer auf dem Feld in der Farbe der Heimatbasis beginnt, die mit der Zielmarke in der Mitte des Spielbretts übereinstimmt. In diesem Beispiel befindet sich der rote Hase in der roten Heimatbasis, oben auf dem Feld mit dem Jahrmarkt (Carnival).
- Das „Bunny Token“ ist die Spielmarke, die jeder Spieler bewegt, wenn er der Code-Karten-Sequenz folgt, die er auslegt.

### Ziel Spielmarke (hier: Jahrmarkt)

(in der Mitte des Spielfeldes)

Sei der erste hier und gewinne das Spiel, aber vergiss nicht vorher Deine Karotte zu essen!



## SPIELABLAUF

Jetzt da dein Board eingerichtet ist, bist du bereit, Coder Bunnyz zu spielen!

### 1. Ziel

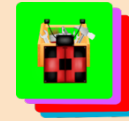
**Das Ziel des Spiels ist es, dass dein Bunny seine farbige Karotte isst und das Ziel zuerst erreicht.**

- Die Spieler legen Karten aus, die die Codefolge (=das Programm) bilden, die den Hasen (Bunny) von der Heimatbasis zu seinem farbigen Karottenstein und dann zum Zielstein führt.
- Jede Karte erlaubt es dem Hasen, sich zu bewegen oder die Richtung zu ändern.
- Um das Ziel zu erreichen, muss jeder Bunny Vorwärtsbewegungen, Drehungen und Sprünge verwenden, um seine Karotte zu essen, um Karotten anderer Farben zu vermeiden und um anderen Bunnyz auszuweichen. Auf dem Weg zu den fortgeschrittenen Levels muss Dein Bunny über die Pfützen springen und sich auch um Zäune herumbewegen.
- Der Spieler, der mit seinem Bunny das Ziel als erster erreicht, gewinnt! Die anderen Spieler spielen weiter, um zu sehen, wer auf Platz 2 und 3 kommt.

## 2. Bunnyz bewegen

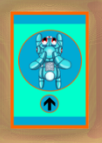
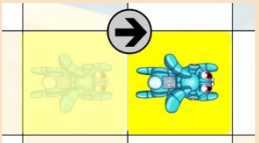



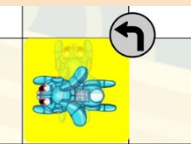

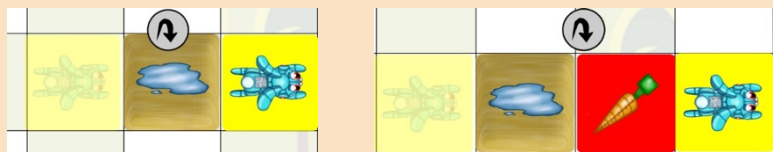


Du wirst Deine farbige Bunny-Spielmarke während des Spiels auf dem Spielbrett bewegen.



Verwende die Bug-Fixit!-Spielmarke, wenn Du während des Spiels feststellst, dass es einen Bug gibt, d.h. dass Du einen Fehler gemacht und die falsche(n) Karte(n) ausgelegt hast, um Dich wieder auf den richtigen Weg zu bringen.

- Immer wenn Du an der Reihe bist, kannst Du mit dieser Spielmarke so viele zuvor gemachte Züge wie Du möchtest rückgängig machen.
- Du kannst die Bug-Fixit! Spielmarke so oft Du willst und in so viele Runden wie erforderlich benutzen. Nachteil: die Benutzung nimmt Deinen ganzen Zug in Anspruch.
- Denke bitte daran, dass Du in einer Runde keine anderen Karten nutzen kannst, wenn Du die Bug-Fixit! Spielmarke benutzt! Denke voraus, um eine korrekte Reihenfolge der Karten zu planen und die Bug-Fixit! Karte möglichst selten zu nutzen!
- Lege Deine Coder Bunnyz Kodekarten aus, um eine Codefolge zu erstellen. Bewege anschließend Deine Bunny-Marke auf dem Spielfeld entsprechend der Reihenfolge der zuvor ausgelegten Karten .

Karte	Bewegung
<b>Vorwärts bewegen</b> 	<p>Bewege Dein Bunny vorwärts (vorwärts ist immer die Richtung, in die die Augen Deines Bunny zeigen)</p> <p>Gehe vorwärts zum Karottenspielstein Deiner Farbe, um ihn zu essen, und lege ihn dann auf die Seite. Gehe vorwärts auf die Zielmarke, um das Spiel zu gewinnen!</p> 
<b>Rechts drehen</b> 	<p>Drehe Dein Bunny nach rechts. (Bleibe auf demselben Feld)</p> 
<b>Links drehen</b> 	<p>Drehe Dein Bunny nach links. (Bleibe auf demselben Feld)</p> 
<b>Springen (Jump)</b> 	<p>Lasse Deinen Bunny über ein oder mehrere Felder springen (immer nach vorne, und in das nächste freie Feld). Sage langsam das Wort "Hüpfen" mit Deiner besten lustigen Stimme 😊</p> <p>Bunnyz können über Pfützen springen, über andere Bunnyz und über Karotten, die anderen Bunnyz gehören. Die Spielmarke, über die du gesprungen bist, bleibt auf dem Spielbrett.</p>  <p>Bunnyz können NICHT über Zäune springen! (Die Spieler müssen sich drehen und vorwärtsbewegen, um Zäune zu umgehen).</p>

### 3. Spielen

Coder Bunnyz ist ein Spiel, bei dem die Spieler der Reihe nach vorgehen:

1. Der Spieler der gerade dran ist würfelt
2. Der Spieler nimmt vom farbigen Kartenstapel die gleiche Anzahl von Karten auf, wie sie die Würfel anzeigen, legt sie in einer Reihe aus und bewegt anschließend seinen Hasen (während er das Wort "Hüpfen" mit seiner lustigsten Stimme sagt).



**Hinweis:** Der Spieler muss die gleiche Anzahl von Karten nehmen, wie durch die Würfel angezeigt werden, kann aber jede Art von Karten auswählen. Versuche Karten auszuwählen, die eine Sequenz bilden, um Deine farbige Karotte und dann das Ziel zu erreichen.

3. Der nächste Spieler ist an der Reihe (der nächste Spieler ist immer der Spieler, der links vom aktuellen Spieler sitzt).
4. Wechselt euch solange ab, bis der Hase eines Spielers seine farbige Karotte gegessen und das Ziel erreicht hat – dieser Spieler gewinnt! Der Rest der Spieler spielt so lange weiter, bis alle mit ihren Bunnyz das Ziel erreicht haben.

#### SPIELARTEN

- |                                 |                        |
|---------------------------------|------------------------|
| Level 1: Einfaches Coding       | (für Anfänger)         |
| Level 2: Erweitertes Coding     | (für Fortgeschrittene) |
| Level 3: Strategie und Code-it! | (für Experten)         |
| Level 4: Professionelles Coding | (für echte Profis)     |



## Level 1: Einfaches Coding

In diesem Level verwenden die Spieler nur die grundlegenden Code-Karten (Vorwärts bewegen, links drehen, rechts drehen und springen), aber nicht die Funktions- oder Wiederholungs-Code-Karten.

### 1.1 Einführung in Befehlsfolgen (Sequence)

(Vorwärts bewegen, links drehen, rechts drehen)



**Vorbereitung:** Jeder Spieler legt 3 Kartenstapel aus: Vorwärts bewegen, links- und rechts abbiegen. Benutze Deine Bunny, Ziel, Karotten und Bug-Fixit!-Spielmarken. Ein Spieler, der als Moderator ausgewählt wurde, legt die 4 Karotten-Steine gleichmäßig auf das Spielfeld, so dass jeder Spieler die gleiche Strecke zurücklegen muss, um seine Karotte zu erreichen.

**Vorschlag:** platziere bei den ersten Spielen die Karotten in der Nähe des Ziels (das ist für Anfänger

etwas leichter). Versuche nach ein paar Runden eine schwierigere Aufstellungen, wie z.B. die Karotten von ihrer farbigen Heimatbasis aus über das Brett zu legen (siehe Bild links).

**Um zu spielen:** Spiele wie in SPIELABLAUF (Seite 6) beschrieben. Jeder Spieler (an der Reihe) würfelt die Würfel. Von den farbigen Kartenstapeln nimmt der Spieler die gleiche Anzahl von Karten auf, wie sie die Würfel anzeigen, legt sie in einer strategischen Reihenfolge seiner Wahl aus und bewegt den Hasen entsprechend der Reihenfolge (während er das Wort "Hüpfen" mit seiner lustigsten Stimme sagt). Die Runde endet, wenn alle Bunnyz ihre Karotten gegessen haben und dann das Ziel erreicht haben.

### 1.2 Einführung in Sprünge (Jump)



**Aufstellung:** Jeder Spieler legt 4 Kartenstapel aus: Vorwärts gehen, links drehen, rechts drehen und springen. Benutze die Bunny, Ziel, Karotten, Pfütze und Bug-Fixit! Marken (Tokens). Der Moderator legt die Pfützen und Karotten für alle Spieler gleichmäßig auf. Für weitere spannende Labyrinth-Ideen schau auf Seite 30 oder unter [www.coderBunnyz.com](http://www.coderBunnyz.com) nach, oder werde kreativ und erstelle eigene Labyrinthe! (für fortgeschrittene Spieler)

**Spiele:** funktioniert genauso wie bei 1.1 (Einführung in Sequenzen) – mit einem Unterschied: ab jetzt kannst Du Sprungkarten nutzen um über Pfützen zu springen, die Dir den Weg versperren oder um zur Möhre und/oder zum Ziel zu gelangen. Bunnyz können hier auch über Zäune springen.

### 1.3 Einführung von wenn-dann Bedingungen (if-then-else)





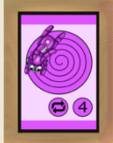



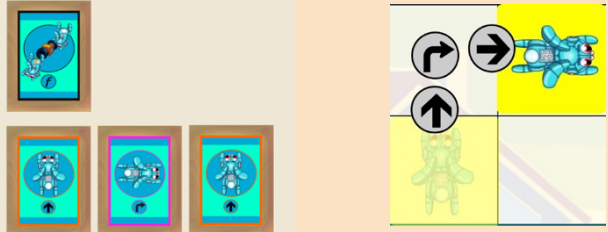
**Aufstellung:** jeder Spieler legt 4 Kartenstapel aus: Vorwärts gehen, links drehen, rechts drehen und springen. Benutze die Bunny, Ziel, Karotten, Pfützen und Bug-Fixit! Marken (Tokens). Der Moderator legt die Pfützen und Karotten für alle Spieler gleichmäßig auf. (identisch zu 1.1 & 1.2)

**Spiele:** funktioniert genauso wie bei 1.2 (Einführung in Sprünge) – mit einem Unterschied: Du kannst Bedingungen „einbauen“ und bei Bedarf die Richtung wechseln (z.B. den Bunny drehen) Bunnyz können nicht über Zäune springen -diese sind einfach zu hoch!

Spiele mehrmals bevor Du Dich an schwierigere Aufgaben machst! Sobald Du bereit bist, kannst Du andere Ziele und verschiedene Anordnungen von Zäunen, Pfützen und Karotten ausprobieren oder eigene Labyrinthe erstellen.

## Level 2: Erweitertes Coding

In diesem Level verwenden die Spieler zusätzlich zu den grundlegenden Codekarten auch Wiederholungs- und/oder Funktionskarten.

Karte	Bewegung
<b>Wiederholung</b> (jeder Karte hat eine Nutzungszahl von 2,3 oder 4)	Wiederhole eine Sequenz von Codekarten mehrmals (2, 3 oder 4). Zum Beispiel erlaubt jede der untenstehenden Kombinationen Deinem Bunny, sich 4 Felder vorwärts zu bewegen:  Eine Sequenz (Reihenfolge) von 4 „Vorwärts Bewegen“ Karten 
 	Eine „Wiederholung 2“-Karte mit zwei „Vorwärts bewegen“ Karten 
	Eine „Wiederholung 4“-Karte mit einer „Vorwärts bewegen“ Karten 
<b>Funktion</b> 	Wende eine Funktionskarte auf eine Sequenz von Codekarten an, um diese Sequenz als einen Zug zu spielen. Diese Sequenz wird auch als „Function Pipe“ bezeichnet. Zum Beispiel: Die Funktionskarte, die auf eine Function Pipe (Sequenz) angewendet wird (vorwärts, nach rechts, vorwärts) 

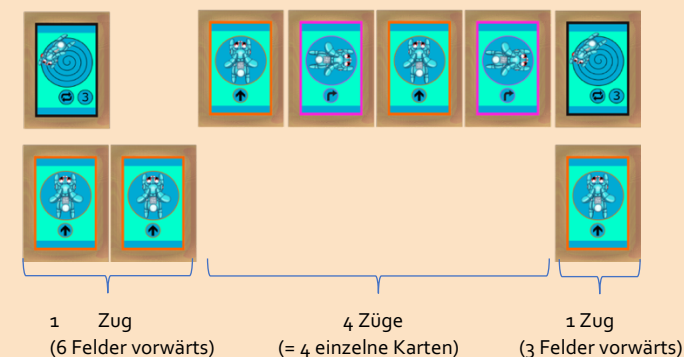
## 2.1 Wiederholung (Einführung von Schleifen/Iterationen)

**Aufstellung:** jeder Spieler legt 5 Kartenstapel aus:

Wiederholung, Vorwärts gehen, links drehen, rechts drehen und springen. Wähle das Ziel aus und lege die Marken je nach gewünschtem Schwierigkeitsgrad für Anfänger, Fortgeschrittene oder Profis aus (je nach Erfahrung der Spieler).

**Spiele:** funktioniert genauso wie in Level 1 (Einfaches Coding) – mit einem Unterschied: Jeder Spieler muss jetzt mindestens einmal eine Wiederholungskarte in seinem Code benutzen um zu gewinnen. Der Spieler kann die Wiederholungskarte an jeder Stelle in seiner Sequenz platzieren.

**Wichtig:** Das Ausspielen von 1 Wiederholungskarte (zusammen mit allen Karten, für die sie gilt) zählt als 1 Zug. D.h. 1 oder mehrere Code-Karten unter einer Wiederholungskarte werden als 1 Zug gezählt. Der untenstehende Code, der Wiederholungskarten verwendet, erlaubt es dem blauen Spieler, das Ziel zu erreichen: (in diesem Beispiel wurde eine 6 gewürfelt)





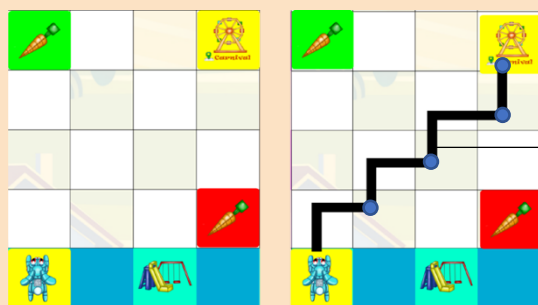
## 2.2 Einführung von Funktionen (oder „Define“)

**Aufstellung:** jeder Spieler legt 5 Kartenstapel aus: Funktion, Vorwärts gehen, links drehen, rechts drehen und springen. (Wiederholung wird nicht genutzt). Ordne die Karten nach der Erfahrung der Spieler an. (genau wie bei 2.1)

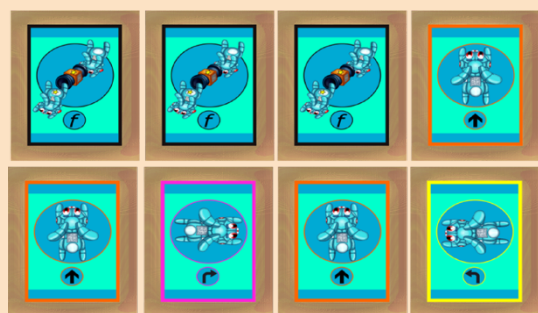
### Spielen:

- Erstelle zunächst den Function Pipe Code. Plane den Code sorgfältig – überlege Dir, welche Reihenfolge (Sequenz) Du immer wieder verwenden willst. Lege den Function-Pipe-Sequenzcode unter den Hauptcode. Jeder Spieler legt nur **eine** Function Pipe pro Spiel an. Der Spieler kann mehr als eine Funktionskarte verwenden, aber jede Funktionskarte folgt der gleichen Reihenfolge der Karten.
- Spiele wie im Kapitel SPIELABLAUF (Seite 6) beschrieben. Die Runde endet, sobald alle Bunzys ihre Karotte gegessen und das Ziel erreicht haben.

Der Code unten benutzt Funktionskarten und erlaubt Hopsy (blauer Spieler) das Ziel zu erreichen nachdem eine 4 gewürfelt wurde:



Bewegung des Bunzy, wenn die komplette Code Sequenz und Function Pipe (siehe unten) gespielt wurde.



Code Sequenz

Function Pipe  $\neq$

## Level 3: Strategie und Code-it!

Dieses Level ist für Spieler ab 8 Jahren gedacht oder für Spieler, die alle vorherigen Stufen abgeschlossen haben. **Wichtig:** Lege die Wiederholungskarten, Funktionskarten und Bug-Fixit!-Marken beiseite um dieses Level zu spielen. Jeder Spieler schreibt eine komplette Code-Sequenz.

### 3.1 Einfarbig (Single Color)

**Aufbau:** Hier verwendest Du nur die grundlegenden Code-Karten. (Vorwärts, rechts- & links drehen, Springen). Jeder Spieler mischt seine Karten und legt sie verdeckt in einem Stapel ab (die Seite "Coder Bunzys Code Card" sollte sichtbar sein). Alle Basiskarten Deiner Farbe werden miteinander gemischt. Wähle und lege die Spielmarken für ein leichtes, mittleres oder schwieriges Labyrinth aus. (je nach Geschicklichkeit der Spieler).

### Spielen:

1. Jeder Spieler würfelt zweimal, addiert die Zahlen und wählt die Anzahl der Karten entsprechend der gewürfelten Augenzahl. Wähle die Karten sorgfältig aus und ordnen sie so an, dass Dein Hase seine Karotte essen kann und das Ziel erreicht.
2. Lege Karten, die Du gerade nicht benötigst auf einen Stapel zur Seite und lege die Karten, die Du für Deinen Code benötigst, in der gewünschte Reihenfolge verdeckt ab.
3. Wechselt euch so lange ab, bis Ihr alle Karten für euren Code zusammen habt. Nutze die Karten aus dem Stapel, der zur Seite gelegten Karten, wenn euch die Karten ausgehen. Sobald Du den vollständigen Code (=Dein Code-it! Programm) erstellt hast, kannst Du Deinen Hasen bewegen.
4. Der erste Spieler, der einen vollständigen Code macht, welcher seinen Hasen zur Karotte und zum Ziel führt, gewinnt! Die anderen Spieler spielen weiter, um zu sehen, wer als nächstes kommt. Sei vorsichtig - es gibt keine Bug-Fixes! Für dieses Level! Wenn Du es nicht schaffst Deine Karotte zu essen oder das Ziel zu erreichen, verlierst Du.

### 3.2 Frei kombinieren (Mix-N-Match)

**Aufbau:** Lege die Karten Wiederholung und Funktion beiseite. Mische alle verbleibenden (einfachen) farbigen Karten zusammen. Gebe die Karten verdeckt an jeden Spieler aus. (Bei z.B. 4 Spielern hat jeder Spieler dann 40 Karten und mehr als eine Kartenfarbe)

**Spiele:** Jeder Spieler würfelt zweimal, addiert die Zahlen und wählt die Anzahl der Karten entsprechend der gewürfelten Augenzahl. Wähle die Karten Deiner Farbe sorgfältig aus und ordne sie so an, dass Dein Hase seine Karotte essen und das Ziel erreichen kann. Gebe die Karten anderer Farben an die entsprechenden Spieler weiter und werfe die Karten Deiner Farbe, die Du nicht benötigst, ab. Lege die Karten, die Du für Deinen Code aufbewahrst, in der gewünschten Reihenfolge verdeckt ab. Bewege Deinen Bunny zunächst nicht!

**Hinweis:** Sobald Du an der Reihe bist, hast Du die Möglichkeit Karten zu verwenden, die Du von anderen Spielern erhalten hast, und/oder Karten von Deinem Deck zu ziehen. Fahre mit Schritt 3 von 3.1 Einfarbig (oben) fort.

## Level 4: Professionelles Coding

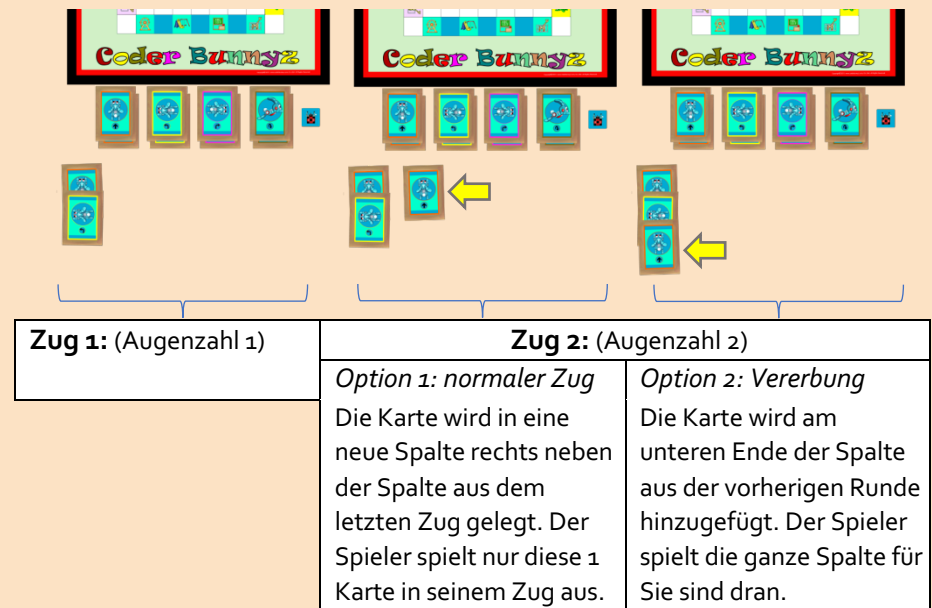
**Aufbau:** Die Wiederholungs- und Funktionskarten werden für alle Spielmöglichkeiten in diesem Level nicht benötigt - diese kannst Du zur Seite legen.

### 4.1 Einführung von Vererbung (Inheritance)

**Aufbau:** jeder Spieler legt 4 Kartenstapel aus: Vorwärts gehen, links drehen, rechts drehen und springen.

#### Spiele:

1. Der Spieler der an der Reihe ist würfelt und nimmt die angezeigte Anzahl an Karten vom farbigen Kartenstapeln auf. Der Spieler legt alle Karten für seinen Zug in einer Spalte (in einer strategischen Reihenfolge seiner Wahl) aus und bewegt seinen Hasen. Dies ist wie ein normaler Zug, außer dass man seine Codefolge in einer Spalte auslegt.
2. Ab dem zweiten Zug haben die Spieler die Möglichkeit, den Code von vorherigen Zügen zu "erben" (inherit). Sobald Sie an der Reihe sind können Spieler:
  - Einen normalen Zug machen: Würfeln und einen neuen Zug machen (genau wie bei einem normalen Zug, aber die Karten in einer neuen Spalte auslegen) **ODER**
  - Vererbung nutzen: Würfeln und füge einem vorherigen Zug neue Karten hinzu (fügen Sie unten an einer Spalte hinzu), dann bewege Deinen Hasen für die Reihenfolge der gesamten Spalte.
3. Das Spiel endet, sobald alle Bunnyz ihre Karotte gegessen und das Ziel erreicht haben.



### 4.2 Paralleles Spiel (Einführung in Parallelisierung)

In diesem Level bewegt jeder Spieler seinen Bunny und seinen Bug. Die Bugs der Spieler können Bunnyz in anderen Farben fressen. Du verlierst das Spiel, wenn Dein Bunny von einem Bug (Käfer) eines anderen Spielers gefressen wird.

**Aufbau:** Folge den Anweisungen zur Einrichtung von Level 4: Pro Coding (Seite 18). Jeder Spieler legt seine Karten in 4 Stapeln auf: Vorwärts bewegen, links drehen, rechts drehen und springen. Platziere Deine Bug-Fixit!-Marke **nicht** neben Deinem Codekartenstapeln. Lege stattdessen Deine Bug-Fixit! Spielmarke in deiner Homepage auf einem Feld deiner Wahl ab.

#### Spiele:

Der Spieler der am Zug ist würfelt 2-Mal und führt **eine** der folgenden Aktionen aus:

- Addiere die Zahlen der Würfel und benutze die Summe für deine Code Sequenz **ODER**
- Verwende die Augenzahl eines Würfels für deine Code-Sequenz und die andere Augenzahl für den Bug-Code. (Hinweis: Du kannst die Summe beider Würfelwürfe **nicht** für den Bug Code verwenden.)

Lege den Bug Code in einer Reihe unter die Code-Sequenz für deinen Bunny:



- Das Spiel endet, sobald alle Bunnyz ihre Karotte gegessen und das Ziel erreicht haben (Gewinner) oder von Bugs gefressen wurden (Verlierer)

#### 4.3 Einführung von Listen (List-Y)

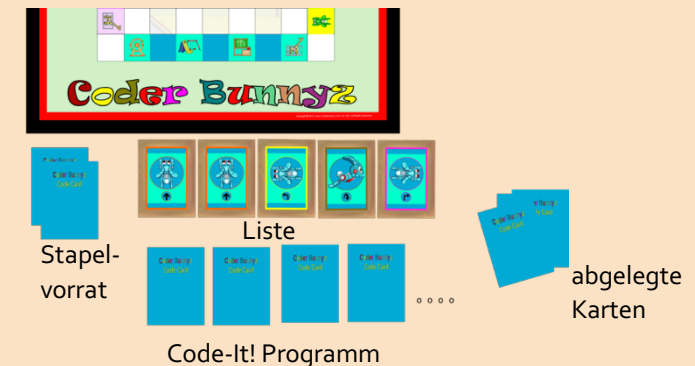
In diesem Level können die Spieler zusätzliche Karten ablegen (eine Liste). Sie können Karten von ihrer Liste nehmen und sie zu ihrem Code-it! Programm nach Bedarf hinzufügen.

**Aufbau:** Folgen Sie den Anweisungen zur Einrichtung von Level 4: Pro Coding (Seite 18). Jeder Spieler mischt seine Karten und legt sie verdeckt in einem Stapel ab (die Seite "Coder Bunnyz Code Card" sollte sichtbar sein). Die Bug-Fixit! Spielmarken werden in diesem Level **nicht** verwendet.

#### Spiele:

1. Jeder Spieler würfelt sobald er an der Reihe ist zwei Mal, addiert die Zahlen und nimmt eine identische Anzahl Karten vom Stapel
2. Mit den ausgewählten Karten beginnt der Spieler ein Code-it! Programm zu erstellen:
  - Der Spieler legt die Karten für sein Code-it! Programm als Befehlsfolge (Sequenz) verdeckt ab.
  - Alle Karten, die er im Moment nicht für sein Code-it! Programm benötigt, werden offen in einer Liste abgelegt.
  - Bewege deinen Bunny noch nicht!

3. Bei jeder Runde nach der ersten würfelt jeder Spieler zwei Mal, addiert die Zahlen und nimmt eine identische Anzahl Karten vom Stapel.
  - Der Spieler ergänzt sein Code-it! Programm mit den gewählten Karten (verdeckt) und fügt die überflüssigen Karten zu seiner Liste (offen) hinzu
  - Der Spieler kann jederzeit Karten aus seiner Liste löschen und zu seiner Code-it! Programm Befehlsfolge hinzufügen. Dies zählt nicht als eigener Zug im Sinne der gewürfelten Augenzahl. (z.B., wenn du 3 gewürfelt hast und 3 Karten vom Stapel nimmst kannst du trotzdem noch beliebig viele Karten von deiner Liste zum Code-it! Programm hinzufügen)
  - Die maximale Listengröße beträgt **5 Karten**. Um zu verhindern, dass deine Liste größer als 5 Karten wird, musst Du entweder Karte(n) ablegen oder Karte(n) zu Deinem Code-it! Programm hinzufügen.



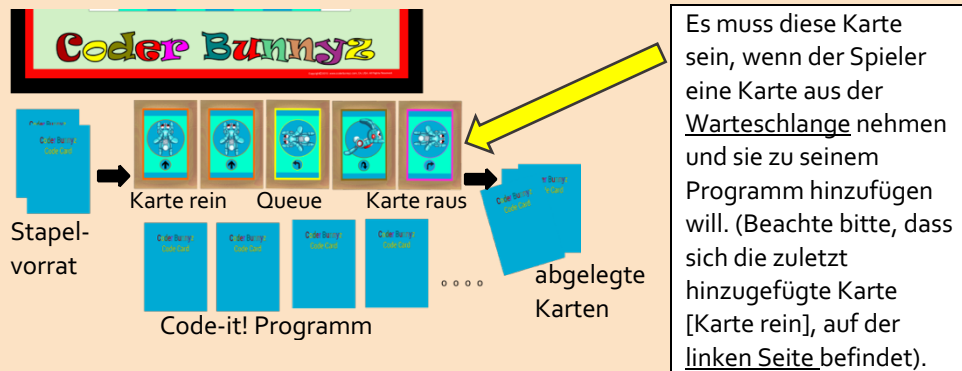
4. Spielt weiter und wechselt euch ab bis ihr alle für euer Programm erforderlichen Karten habt. Mische den Stapel abgelegter Karten gut durch und füge diese dem Stapelvorrat hinzu sobald dieser zur Neige geht. Du kannst Deinen Bunny bewegen, sobald dein Code-it! Programm komplett ist.
5. Der erste Spieler, der ein funktionsfähiges Programm erstellt hat, das den Bunny zur Karotte und anschließend zum Ziel bringt, gewinnt! Die anderen Spieler spielen weiter um zu sehen wer es als nächstes schafft. Achtung: Es gibt in diesem Level keine Bug-Fixits!. Du verlierst, wenn Dein Bunny es nicht schafft die Karotte zu essen und das Ziel zu erreichen.

#### 4.4 Einführung von Warteschlangen (Queue-Y)

In diesem Level können die Spieler zusätzliche Karten ablegen und eine Warteschlange (Queue) bilden. Sie können die erste Karte aus der Warteschlange (die am weitesten rechts liegende Karte in der Reihe) nehmen und sie bei Bedarf zu ihrem Code-it! Programm hinzufügen.

**Aufbau:** Folge den Anweisungen in Kapitel 4.3 (Einführung von Listen)

**Spiele:** Spiele auf die gleiche Weise wie in 4.3 (Einführung von Listen), mit einem Unterschied: der Spieler kann keine Karte aus der Warteschlange frei auswählen und in sein Code-it! Programm einfügen. Vielmehr muss der Spieler die erste Karte in der Warteschlange nehmen, und zwar die Karte, die am weitesten rechts in der Reihe liegt. Der Spieler kann pro Zug so viele Karten aus der Warteschlange nehmen, wie er will, aber er muss immer die Reihenfolge beibehalten, in der er die erste Karte aus der Warteschlange nimmt um diese in sein Code-it! Programm einzufügen, bevor er die nächste Karte aus der Warteschlange nimmt.

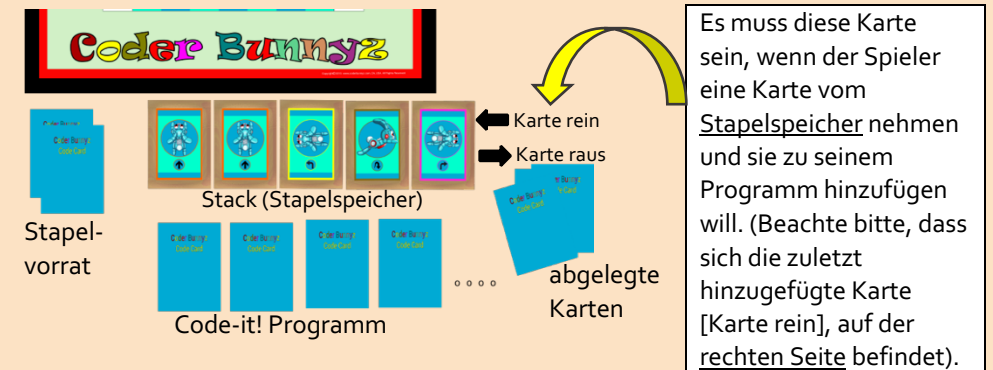


#### 4.5 Einführung des Stapelspeichers (Stack-Y)

In diesem Level können die Spieler zusätzliche Karten in einem Stack (Stapelspeicher) ablegen. Sie können die letzte Karte, die sie zu ihrem Stapel hinzugefügt haben (dies ist die am weitesten rechts liegende Karte in der Reihe), nehmen und sie bei Bedarf zu ihrem Code-it! Programm hinzufügen.

**Aufbau:** Folge den Anweisungen in Kapitel 4.3 (Einführung von Listen)

**Spiele:** Spiele auf die gleiche Weise wie in 4.3 (Einführung von Listen), mit einem Unterschied: der Spieler kann keine Karte aus der Warteschlange frei auswählen und in sein Code-it! Programm einfügen. Vielmehr muss der Spieler die Karte nehmen, die dem Stapelspeicher (Stack) als letztes hinzugefügt wurde. Dies ist die Karte, die am weitesten rechts im Stack liegt. Der Spieler kann pro Zug so viele Karten vom Stapelspeicher nehmen wie er will, aber er muss immer die Reihenfolge beibehalten, in der er die erste Karte vom Stack (Stapelspeicher) genommen hat um diese in sein Code-it! Programm einzufügen, bevor er die nächste Karte vom Stack nimmt.



## Häufig gestellte Fragen (Frequently Asked Questions -FAQ)

### **F : Können sehr junge Kinder das Spiel spielen?**

**A:** Ja, Kinder im Alter von 4 Jahren können Coder Bunnyz spielen. Wenn ein junger Spieler Hilfe braucht, kann ein Erwachsener als Helfer Bunny das Spiel erleichtern.

### **F: Die Regeln sprechen von 4 Spielern. Brauchen wir ständig 4 Spieler?**

**A:** Nein. Ihr braucht keine 4 Spieler. Das Spiel ist für 1-4 Spieler. Jeder Spieler nimmt eine Farbe, wenn es also weniger als 4 Spieler gibt, bedeutet das nur, dass einige Farben nicht gespielt werden. Es kann sogar 1 Spieler spielen.

### **F: Wie wird der Gewinner entschieden?**

**A:** Der Spieler mit dem Hasen, der das Ziel zuerst erreicht, ist der Gewinner des ersten Platzes, der Spieler mit dem Hasen, der das Ziel als zweiter erreicht, ist der Gewinner des zweiten Platzes, und so weiter. Jeder Hase, der seine Karotte erreicht, ist ein Gewinner. (Ausnahme: In Level 1.4: Code-it! Dein erster Programcode! ist jeder Spieler, der beim ersten Versuch die Möhre und dann das Ziel erreicht, ein Gewinner des ersten Platzes)

### **F: Was ist eine Home Base?**

**A:** Die farbige Außenkante des Brettes, die die gleiche Farbe wie der Hase hat, ist die Home Base des Hasen. Jeder Hase beginnt jedes Spiel in seiner farbigen Heimatbasis, auf dem Ziel, das mit dem Zielfeld in der Mitte des Spielbretts übereinstimmt.

### **F: Kann ein Hase in die Homebase eines anderen Hasen gehen?**

**A:** Nein. Jeder Hase kann sich innerhalb seiner farbigen Homebase bewegen, aber nicht in die Homebase eines anderen Hasen gehen; es wäre wie ein Eindringen in das Haus eines anderen Hasen.

### **F: Können Karotten-, Zaun- oder Pfützensteine in einer Homebase platziert werden?**

**A:** Nein. Spielmarken können niemals in einer der Homebases platziert werden.

### **F: Kann der Spieler eine Karotte essen, die eine andere Farbe hat als seine eigene Farbe?**

**A:** Nein. Bunnyz müssen über Karotten springen oder sich um Karotten herumbewegen, die nicht die gleiche Farbe haben wie sie selbst. (Andere Bunnyz sind wie Zäune.)

### **F: Kann der Bunny über das Ziel springen?**

**A:** Nein. Sein Ziel ist es, das Ziel zu erreichen, indem er sich darauf zubewegt. Er kann nicht über das Ziel springen.

### **F: Was passiert, wenn ein Zaun, eine Pfütze oder ein Karottenplättchen einen Hasen daran hindert, sich zu bewegen?**

**A:** Der Helfer-Hase (Moderator) hat eine besondere Fähigkeit, die Einstellungen der Spielmarken nur in Sackgassen zu ändern. Wenn die Zaun-, Pfützen- oder Karottenplättchen so angeordnet sind, dass ein Hase keinen möglichen Weg machen kann (komplette Blockade), dann ist der Hilfshäschen an der Reihe, die Plättchen zu bewegen, um dem Hasen einen Weg zu machen.

### **F: Was passiert, wenn ein Häschen meinen Weg blockiert?**

**A:** Wenn es nicht möglich ist, über den anderen Hasen zu springen oder sich um ihn herum zu bewegen, dann musst du warten, bis der andere Hase sich bewegt und der Weg frei ist. In dieser Situation wechselt man sich weiter ab (würfel weiter und füge deine Kartenfolge oder dein Codeit Karten hinzu! Programm). Wenn du entsperrt bist, bewegen deinen I Hasen: Benutze alle Code-Karten, die dir während der Blockade angesammelt hast. Das können mehrere Züge sein.

### **F: Ok, können Sie klarstellen, wo Sie eine Jump Code Karte verwenden sollen und wo nicht?**

**A:** Sicher, springe über eine Pfütze, über eine andere Karotte eines Hasen oder über einen Hasen. Wenn es 2 (oder mehr) von den gerade aufgelisteten Spielsteinen in einer Reihe gibt, kannst du mit 1 Sprungkarte darüber springen - aber du kannst nicht springen, wenn es kein leeres Feld oder das Ziel auf der anderen Seite des oder der Spielmarken gibt, über die du springen willst. Außerdem kann man nie über einen Zaun, das Ziel oder ein leeres Feld springen.



**F: Was ist, wenn mir die Move Forward Code Cards ausgehen?**

**A:** In dem ungewöhnlichen Szenario, dass dir die Move Forward Code Cards ausgehen, kannst du Karten, die du zu Beginn des Spiels benutzt haben, wiederverwenden.

**F: Was ist, wenn mir die Wiederholungs- oder Funktionscodekarten ausgehen?**

**A:** Wenn dir die Wiederholungs- oder Funktionscodekarten ausgehen, verwende die grundlegenden Codekarten, um Ihre Codesequenz zu vervollständigen.

**F: Können wir in einem Spiel sowohl Wiederholungs- als auch Funktionscodekarten verwenden?**

**A:** Ja. Wiederholungs- und Funktionskarten führen jeweils unterschiedliche Konzepte ein. Du kannst nur eine oder beide in einem Spiel verwenden. Es ist jedoch nicht erlaubt, sowohl eine Funktionskarte als auch eine Wiederholungskarte in einem Zug zu verwenden.

-Dieser Platz ist absichtlich frei-

**F: Kann ein Spieler mehr als 1 Funktionssequenz (Function Pipe) in einem Spiel auslegen?**

**A:** Nein. Jeder Spieler kann nur 1 Funktionssequenz pro Spiel haben. Der Spieler kann diese Funktionssequenz mehrmals pro Spiel verwenden (solange er jedes Mal eine Funktionskarte ausspielt).

**F: Wie viele Karten kann ein Spieler unter einer Repeat 2,3, oder 4 Code Karte auslegen?**

**A:** Du kannst beliebig viele Karten unter einer Wiederholungskarte auslegen, solange es einen Pfad für den Hasen gibt. Alle Karten, die unter einer Wiederholungskarte ausgelegt werden, zählen als 1 Zug.

(Die Zahlen 2,3 und 4 stehen nicht für die Anzahl der Karten in der Reihenfolge. Vielmehr stellen sie die Anzahl der Wiederholungen der Sequenz dar).







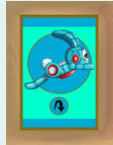






## QUICK REFERENCE

### 1. Spielmarken (Tokens)

Karte	Beschreibung
<b>Bunny (Hase)</b> 	Zu Beginn des Spiels wählt jeder Spieler eine farbige Hasen Spielmarke aus. Dies ist die Spielmarke, die Du während des Spiels auf dem Brett bewegst.
<b>Ziel (Destination)</b> 	Dies ist die Spielmarke, die in der Mitte des Spielbretts platziert wird. Jeder Spieler muss eine Sequenz von Karten legen, die seinen Hasen zum Zielfeld führt (aber erst, nachdem der Hase seine farbige Karotte gegessen hat)! Sei Erster am Ziel und gewinne das Spiel!
<b>Karotte</b> 	Jeder Bunny muss die Karotte seiner Farbe essen, bevor er zum Ziel geht. Jeder Spieler muss eine Sequenz von Karten legen, die seinen Hasen auf das Feld mit dem Karottenplättchen, das die gleiche Farbe wie der Hase hat, führt. Der Hase muss sich umherbewegen oder über andersfarbige Karotten springen.
<b>Bug-Fixit!</b> 	Mit dem Bug-Fixit! Token können Sie so viele Karten (die Du in früheren Runden gespielt hast) rückgängig machen, wie Du möchtest. Du kannst die Bug-Fixit! Spielmarke in beliebig vielen Zügen verwenden, aber die Verwendung der Spielmarke nimmt Deinen gesamten Zug in Anspruch (wenn Du die Bug-Fixit! Spielmarke nutzt, kannst Du keine neuen Karten mehr ausspielen).
<b>Pfütze (Puddle)</b> 	Ein Hase kann nicht auf ein Feld mit einer Pfütze (Puddle) gehen. Der Hase muss sich umherbewegen oder über Pfützen springen, wenn er sie passieren will.
<b>Zaun (Fence)</b> 	Ein Hase (Bunny) kann nicht auf ein Feld mit einem Zaun (Fence) gehen oder darüber springen. Der Hase muss um den Zaun herumgehen, wenn er ihn passieren will.

### 2. Kodekarten

Legen Deine Coder Bunnyz Kodekarten aus, um eine Codesequenz zu erstellen. Bewegen Deine Bunny Spielmarke entlang der Tafel entsprechend der Reihenfolge der ausgelegten Karten.

Basis Kodekarten	
<b>Vorwärts</b> 	 Beweg deinen Bunny ein Feld vorwärts. (Vorwärts ist immer die Richtung, in die die Augen des Häschens schauen)
<b>Rechts drehen</b> 	 Drehe deinen Hasen 90° nach rechts. Bleibe auf demselben Feld.
<b>Springen</b> 	Springt über Felder mit Pfützen, Bunnyz oder Karotten anderer Farben und landet auf dem nächsten leeren Feld. Ihr könnt nicht über Zäune oder das Ziel springen!
<b>Links drehen</b> 	 Drehe deinen Hasen 90° nach Links. Bleibe auf demselben Feld.
Erweiterte Kodekarten	
<b>Wiederholen</b> 	 Wiederhole eine Sequenz von Kodekarten 2,3 oder 4 Mal
<b>Funktion</b> 	 Wenden diese Karte auf eine Sequenz von Kodekarten (Function Pipe) an, um sie in einem Zug auszuspielen.

### 3. Labyrinth

einfach



fortgeschritten



schwierig



Coder Bunnyz, das dazugehörige Logo, das Design der unverwechselbaren Coder Bunnyz-Markenspielkarten und das unverwechselbare Kartendesign & alle Spielmarken sind Warenzeichen von Coder Bunnyz in den Vereinigten Staaten. ©2016 Coder Bunnyz, Santa Clara, CA 95051. Alle Rechte vorbehalten. TM und ® kennzeichnen U.S.-Marken.

Für weitere Einzelheiten wenden Sie sich bitte an:

Web: <http://www.coderbunnyz.com>

Mail: [mom@coderbunnyz.com](mailto:mom@coderbunnyz.com)

Tipps zur Platzierung der Karotten-Marken finden Sie auf Seite 4. Denken Sie daran, dass Sie Ihr eigenes Labyrinth erstellen können oder weitere Labyrinth-Ideen unter [www.coderbunnyz.com](http://www.coderbunnyz.com) erhalten.