



## Einstieg

Für den nächsten Schritt bauen wir uns Animationen zusammen. Hierfür benutzen wir die Bilder, die im Kurs-Ordner drinnen sind. Um dies zu verwenden, ziehen wir sie jetzt alle Dateien, die im *FrogJump* Ordner drinnen sind in unser Unity Projekt hinein, und zwar in unseren *Art* Ordner.

Nun müssen wir die *Animator* Komponente zu unserem Spieler hinzufügen. Um das zu tun, gehen wir in der **Hierachy** auf das *PlayerSprite* Objekt, welches sich im Player befindet. Sobald dieses ausgewählt ist klicken wir im **Inspector** auf *add Component* und suchen dort nach *Animator* um es dann hinzuzufügen.

Jetzt benötigen wir ein neues Fenster, und zwar das *Animation* Fenster. Wenn du es nicht finden kannst, klicke oben im Unity Fenster auf *Window -> Animation -> Animation*. Dann sollte ein neues Fenster auftauchen, welches man so positionieren kann wie man will.

In diesem Fenster kann man nun einen neuen Animationsclip erstellen. Diese Clips speichern wir in einem neuen Ordner mit dem Namen *AnimationClips* den wir unter dem Ordner *Art* erstellen. Der neue Animations Clip den wir erstellen nennen wir *walk*. Im **Animations** Fenster erscheint nun eine Timeline. Jeder Strich in dieser Timeline steht für eine Sekunde. Der senkrechte weiße Strich zeigt, wo wir uns gerade im Clip befinden. Diesen Strich kann man auch mit der Maus bewegen. Im selbigen Fenster befindet sich auch ein Dropdown Menü, welches alle Animationen für das ausgewählte Objekt enthält. Der Knopf, der für uns gerade wichtig ist, ist der rote Knopf. Wenn wir diesen drücken wird die Timeline rot und alles was wir nun mit dem Frosch machen wird aufgenommen. So würden Änderungen an Größe und Rotation nun Teil der Animation werden.

Um diese Funktion mal auszuprobieren, machen wir das folgende:

Wir gehen zum ersten Frame der Timeline. Jetzt ändern wir die Rotation des Frosches. Es erscheint eine Raute auf der Timeline, welche für die Änderung der Rotation steht. Wenn wir dies jetzt an einem anderen Frame wiederholen und die Animation abspielen, indem wir im **Animations** Fenster den Play-Knopf drücken, sehen wir, wie sich die Rotation vom einen auf den anderen Frame ändert. Wir sehen auch, dass der Übergang zwischen den zwei Positionen fließend verläuft.

Um jetzt mit unserem Projekt fortzufahren, löschen wir zuerst die Keyframes, welche den Frosch rotieren wieder aus der Timeline. Nun gehen wir im **Project** Fenster zu der Stelle, an der die Bilder für die Lauf-Animationen liegen. Diese liegen im Ordner *Art -> Animations -> Frog -> walking-loop*. Die wählen wir nun alle aus, indem wir das erste Bild anwählen und dann die Umschalt-Taste gedrückt halten



und das letzte Bild anwählen. Und jetzt können wir alle auf einmal in die Timeline ziehen. Unity wird uns dann automatisch eine Animation daraus erstellen. Es wird ein Bild pro Frame abgespielt. Das können wir auch sehen, wenn wir wieder auf die Play-Taste drücken. Denn dann fängt der Frosch an zu laufen. Auch sehen wir, dass der Frosch gerade zu schnell läuft. Um das zu ändern, wählen wir in der Timeline alle Keyframes aus, indem wir alle mit der Auswählbox erfassen, die erscheint wenn die linke Maustaste gedrückt gehalten wird. Links und rechts der ausgewählten Keyframes erscheinen, dann blaue Striche. Diese kann man verschieben, um alle ausgewählten Keyframes in die Länge zu ziehen oder um gekehrt die ganze Animation kürzer zu machen. Dies wird dann auch die Animation verlangsamen, da ein Laufzyklus dann über mehr Frames verläuft und somit länger geht. Dies muss nun so angepasst werden, dass es für den Frosch natürlich aussieht.

Wenn wir nun das Spiel mal über den Play-Knopf oben laufen lassen, sehen wir, dass der Frosch nun eine Lauf-Animation hat. Wir sehen aber auch, dass der Frosch nicht mehr aufhören will zu laufen. Hierfür benötigen wir jetzt eine weitere Animation in der, der Frosch einfach nur steht. Um einen neuen Animationsclip zu erstellen, müssen wir zurück in das **Animations** Fenster gehen und dort auf das Dropdown Menü drücken. Dort haben wir die Option, einen neuen Clip zu erstellen. Den neuen Clip nennen wir *idle*. Jetzt wollen wir mal auf eine andere Weise einen Keyframe hinzufügen. Und zwar drücken wir nun auf den Knopf *Add Property* dort können wir die Eigenschaft auswählen, die wir animieren wollen. In unserem Fall wäre das *Sprite Rendere -> Sprite*. Wir sehen, dass er das Idle Sprite nun in die Animation gepackt hat.



## Aufgabe

Erstelle eine einen Animations Clip für das Laufen und den Idle State des Frosches , damit dieser nicht nur über den Boden rutscht. Passende Assets werden über den Kurs bereitgestellt.