



Einstieg

Warnung: Dieses Kapitel enthält wichtige Schritte in Text-Form welche nicht im Video aufgeführt sind.

Im ersten Schritt werfen wir einen genaueren Blick auf die Funktionen von Unity und lernen die einzelnen Möglichkeiten, der Game Engine kennen. Im folgenden Video erhaltet ihr daher einen generellen Überblick über das Programm selbst.

Scene View: In diesem Fenster wird das Level gebaut und angepasst. Spiel-Elemente können hier in der Spielumgebung platziert und positioniert werden. Im Moment wird nur die Kamera angezeigt, da wir noch nichts hinzugefügt haben.

Game View: Diese Ansicht zeigt, was die Kamera gerade sieht. Das entspricht auch dem, was der Spieler am Ende sehen wird, sobald das Spiel fertig ist. Dieses Fenster ist gerade noch leer, da wir noch nichts hinzugefügt haben.

Hierarchy: Hier sind alle Objekte gelistet, die sich gerade in der **Scene View** befinden.

Um die voran gegangenen Elemente auszuprobieren, können wir mal ein Viereck erstellen. Um dies zu tun, müssen wir in der **Hierarchy** einen Rechtsklick machen, dann unter *2D Object -> Sprites -> Square* klicken. Dies erschafft ein Viereck, der nun in der **Hierarchy** und der **Scene View** zu sehen ist. In der **Scene View** man das Viereck auch durch das gedrückt halten bewegen. Wenn wir jetzt zu der **Game View** wechseln, sehen wir auch, dass der Spieler ein Viereck sieht.

Inspector: In diesem Element werden Informationen zu dem Objekt angezeigt, welches gerade ausgewählt ist. Jedes Objekt hat z.B. eine Transform Komponente in der drin steht, wo es sich in der Szene gerade befindet, ob es in einem bestimmten Winkel rotiert wurde oder wie groß es gerade ist. In der oberen linken Ecke des Unity Fensters sind ein paar Tools zu sehen, mit denen man die Werte der Transform Komponente beeinflussen kann. Sobald man eines der Tools angewählt hat, kann man in der **Scene View** das Objekt mit der Hilfe des Tools verändern. Zu sehen ist wie sich auch die Werte der Transform Komponenten ändert, wenn wir das Objekt beeinflussen. Darüber hinaus besitzt unser Viereck auch eine Sprite-Renderer Komponente in der Einstellungen drin sind, mit denen man das Aussehen des Vierecks beeinflussen kann. So können wir hier die Form und Farbe ändern.

Project: Das ist der Ort, an dem alle Dateien, wie Skripte, Grafiken und Sounds zu sehen sind, die für das Projekt genutzt werden.



Hier können wir mal unsere ersten Assets rein machen. Dafür erstellen wir mal einen neuen Ordner, indem wir im Assets Bereich Rechts klicken und dann unter *Create* den Punkt *Folder* auswählen. Da wir gleich ein paar Grafiken hinzufügen wollen, nennen wir diesen neuen Ordner *Art*. Um unsere Grafiken aus einer der früheren Kapiteln hinzuzufügen können wir diese nun vom Dateimanager in diesen neuen Ordner ziehen. Jetzt sind diese Grafiken im Projekt und wir können sie für unser Spiel verwenden. Zum Test können wir mal das Frosch Sprite in die **Scene View** ziehen. Dieses Sprite wird jetzt wie das Viereck in der **Hierarchy**, der **Scene View** und der **Game View** angezeigt. Wir können das neue Objekt in der **Hierarchy** umnennen, indem wir es rechtsklicken und auf *rename* drücken. Wir nennen es *Player*.

Console: Dieses Fenster wird später eine wichtige Rolle spielen. Jetzt ist es noch leer, aber später sind hier Fehlermeldungen und Informationen zu den Skripten drin, die ausgeführt werden.

Layout: Die Anordnung der Fenster kann man frei modifizieren, indem man sie einfach umherzieht. So kann man ein passendes Layout für sich zusammen stellen. Wenn aussersehen ein Fenster gelöscht wurde, kann man dieses unter dem Tab *Window -> General* neu öffnen. Man kann auch das Standart Layout wiederherstellen, indem man rechts oben das Layout Dropdown Menü öffnet und dort auf Default klickt.

Zur Übung fügen wir mal den Spieler, den Boden und eine Plattform ein. In diesem Teil ist noch nicht so wichtig, dass die Funktion jedes Elements klar ist sondern, dass wir ein wenig Übung mit der Navigation bekommen.

Wir fangen mit dem Boden an. Dafür machen wir einen Rechtsklick auf die freie Fläche der **Hierarchy** View. Dann klicken wir auf *2D Objekt -> Physics -> Static Sprite*. Das neue Objekt nennen wir *Ground*, indem wir es direkt am Anfang so nennen oder im nach hinein umbenennen indem wir einen Rechtsklick tätigen und dann auf *rename* drücken. Wenn das getan ist skalieren wir den Boden noch so, sodass er nur einen kleinen Bereich des unteren Bildschirmes einnimmt und sich dabei aber über die komplette Länge erstreckt.

Die gleichen Schritte wiederholen wir für die Plattform auch, mit dem Unterschied, dass wir das Objekt diesmal *Platform* nennen und dass wir sie zu einer Plattform skalieren die in der Luft schwebt. Wenn wir den Boden oder die Plattform ausgewählt haben sehen wir im **Inspector** einen Teil der *Sprite Renderer* heißt. Hier gibt es ein *Color* Feld in dem wir eine gewünschte Farbe auswählen können. Wählt hier eine Farbe für den Boden und die Plattform die euch gefällt.



Jetzt fügen wir den Spieler hinzu. Dafür erstellen wir mit einem weiteren Rechtsklick ein Leeres Objekt indem wir die Option *Create Empty* auswählen. Diesem Objekt geben wir den Namen Player. Im **Inspector** haben wir die Möglichkeit eine neue Komponente hinzuzufügen indem wir auf den *Add Component* - Knopf drücken. Zuerst fügen wir die *Rigidbody 2D* Komponente hinzu. Als zweites brauchen wir dann die *Capsule Collider 2D* Komponente. Dort geben wir für die *Size* einen y-Wert von 2 an. Jetzt ziehen wir von der **Project** View das Frosch Sprite welches wir zuvor verwendet haben auf das neu *Player* Objekt in der **Hierarchy**. Dieses sollte jetzt im Player liegen. Das Sprite müssen wir in der **Hierarchy** nun auswählen. Mit dem ausgewählten Sprite gehen wir dann in den **Inspector**. Dort drücken wir in der *Transform* Komponenten auf die drei Punkte in der oberen rechten Ecke und drücken *Reset*. Hiermit haben wir das Projekt so vorbereitet damit wir mit dem nächsten Kapitel weiter machen können.

Da wir nun jedes Element in Unity kennengelernt haben könnt ihr ein wenig mit diesen herumprobieren, um euch mit den Funktionen vertraut zu machen, bevor wir im nächsten Kapitel damit anfangen unser erstes Spiel zu programmieren.



Aufgabe

Die Aufgabe dieses Mal ist es mit der Hilfe des Textes und des Tutorial Videos die verschiedenen UI-Elemente von Unity zu verstehen. Diese zu verstehen und verwenden zu können wird für die folgenden Kapitel wichtig sein. Dann fügen wir noch den Boden, eine Plattform und den Spieler ein um sie im nächsten Kapitel benutzen zu können.