# Unity Animator

La ventana de Animator se usa para hacer las transiciones entre una animación y otra.

El animator controller es el que controla todo esto y se adjunta a un componente animator.

**ANIMATION**

Sirve para crear animaciones. Con la K se añade un Keyframe. Se puede acceder a cualquier componente del objeto a la hora de animar.

**ANIMATOR CONTROLLER**

Si arrastras la animación al layer te inserta la animación.

Los nodos son cada una de las animaciones. Por defecto están Entry, Exit y AnyState.

* Entry: Simplemente hace la transición al estado por defecto.
* Exit:
* AnyState:

Tiene la parte de los parámetros, estados y las capas.

* Parametros: Son variables que se usan para controlar las transiciones y se puede acceder a ellas desde scripts. Los tipos son Int, Float, Bool, y Trigger (Un bool que se reinicia desde el controlador cuando se efectua una transición, sirve para hacer una transición fija, como por ejemplo cualquier animación que no sea un Loop).

Para acceder a las variables necesitas una clase Animator y con Set(nombre\_variable, valor\_local).

Animator.SetBool(“Fire”, fire);

* Estados: son la transición entre una animación y otro. Aquí dentro se indica el parámetro o parámetros a evaluar y que valor han de tener para que se cumpla la transición.
* Capas: Sirve para separar por capas el personaje. Por ejemplo puedo animar la parte de arriba y la de abajo del cuerpo de forma separada de modo que mientras camina pueda hacer otras cosas sin dejar de caminar.

Además se pueden añadir script a los cambios de estado. Se llaman StateMachineBehaviour.

También se puede Agrupar un grupo de estados para que visualmente sea mas comodo. Los estados agrupados se llaman sub-state.

**ANIMATOR OVERRIDES CONTROLLER**

Se usa para usar el mismo animator controller, con los mismos parámetros y los mismos estados pero con otras animaciones distintas. Se le pasa el Animator controller y después las animaciones.

**ANIMATOR COMPONENT**