**GETAFEVANIA (Tema6)**

Juego Nuevo**: Plataformas** en 2D y con Versión 2017.

Caracteristicas de un Juego 2D:

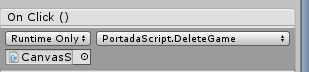
* Son Sprites imágenes planas
* La forma de componer el escenario mediante Tiles (Azulejos)
* Hay técnicas específicas para que los fondos se muevan a cierta velocidades.
* Marcador de Vidas, Balas
* Plataforma Movil
* Animar el nuestro personaje

Crear una Portada para nuestro Juego.

* Hay una página de <https://es.cooltext.com/> para generar nuestros textos.
* Metemos un CANVAS:
  + Escalar con Space wih Screen y le ponemos la resolución. Todos los componentes se Escalan con la pantalla (de pixeles se distribuyen proporcionalmente en la pantalla).
  + Canvas Scaler Canvas y tenemos que modificar el Match para que nos diga que queremos que se escale. Para que me haga el ajuste en ancho o alto a 0.5
    - Si tenemos el Constan Pixel Size se pone en un sitio en concreto (están las mariposas juntas)
    - Si las Anclas están distribuidas como pasa con el título.
    - Si ponemos el Canvas a **Scale with Screen size** ponemos ajustar el alto o el ancho, normalmente lo pondremos 0.5
* Creamos un Botón y nos lo descargamos del <https://kenney.nl/assets> **UI Pack**.
* En mi CANVAS nos creamos un elemento de tipo Imagen.
* En el CANVAS desaparece el Transform y aparece el **Rect Transform**.
  + Al principio el CANVAS tiene 3 modos de proyección (Renderizado):
    - **Screen Space – Overlay** por encima de todo. Siempre veré por encima de la Escena el CANVAS. Sobre la pantalla siempre visible.
      * Lo usare para poner vidas del juego, botones…
      * Cuanta munición tengo, que arma tengo seleccionada…..
      * Para menús…
    - **Screen Space-Camara** sobre qué cámara quiero que se proyecte la interfaz gráfica
      * Tengo varias cámaras y una de las Camaras tengo una vista del vehículo desde dentro y ahí quiero que se proyecte información de la velocidad……
    - **Wold Space**, el Rect Transform ha dejado de estar bloqueado así que el CANVAS pasa a formar parte del mundo así que se ha convertido como un GO y ese CANVAS lo puedo coger que sea dependiente de otro GO como que vaya pegado a la cabeza de mis personajes para saber sus vidas…..
* En el CANVAS poner una imagen
  + Tiene unos manejadores en forma de Mariposas. Son las Anclas para que se queden anclado a una determinada posición.
  + Le ponemos un **Source Image**
  + Podemos dar a **Preserve Aspect** para que me ponga la imagen son su tamaño origina.
* En la pantalla de Game donde pone el Free Aspect es el tamaño de nuestra ventana de juego.
  + Pero nosotros no trabajaremos con el Free Aspect si no con pantallas normales
  + Podemos poner 1024x768 y asi vemos como quedaría mi interfaz con esa resolución.
  + Podemos crear una nueva Interfaz como Lenovo7 de 1024x600
* Le ponemos una imagen de fondo **BackGround** y la ponemos en la Carpeta Sprites.
  + Me lo pone delante de nuestra imagen así que tendremos que cambiarle de orden para que me lo ponga en el fondo.
  + Modifico las mariposas para que se me queden a las esquinas
  + Y poniendo 0 me ajusta la imagen a esas anclas



* Le ponemos un Título que este arriba centrado.
* Y ponemos un Botón para Resetear el Juego. Lo situaremos arriba a la derecha.
  + **Interactable** si podemos hacerle el click al objeto o no.
  + Podemos hacer **transiciones** como cuando pasemos por encima o presionamos…..
  + **Color Multiplier** para multiplicar el color
  + **Fade Duration** cuánto tarda en hacer las transiciones de un color a otro.
  + **Navigation** cuando quiero con el tabulador y de un elemento a otro
  + **On Click** 
    - Hay gente que el Script lo asocia al Canvas
    - Oooo a un GO vacio
    - Realmente puede estar el Script en cualquier sitio. Lo único que tendríamos que hacer es arrastrar el GO donde este el Script

 así que nos creamos un GO vacío **CanvasScriptContainer**.

* Tenemos que hacer algunos personajes que se muevan hacia la izquierda y que cuando salgan de la pantalla tenemos hacer que se vuelvan a la posición inicial.

xxxxxxxxxxxxxxxxxxxxxxxx

* Vamos a crear un Panel (es un contenedor).
* El panel está anclado a los extremos pero queremos que ocupe solo una parte. Y nos serviría para meter los botones
* Poner el Panel a 1/3 de la pantalla Min 0,33 y de Max 0,66 en la X y en la Y. Le ponemos en la RectTransform a 0.
* Le agregamos un layout para decirle como están distribuidos lo que metamos. Vertical Layout Panel
* Metemos el Botón dentro del panel
* El panel lo ponemos en el Centro y duplicamos botones. Y podemos poner un Pading para hacer espaciados.
* MainScene nueva
* Build Settings y añadimos las dos escenas, arrastrando.
* Al Botón Start le agregamos el OBJ CanvasScriptScoiner que tiene el script de Portada y contiene un método de Start y se lo añadimos en el onclick.
* Escena MainScene: Coger una portada y añadir un Sprite 2D y le añadimos nuestra imagen al Sprite.

Canvas tendrá: Vidas con muñequitos, Puntuación.

El Canvas lo configuramos:

* Como queremos que se renderize lo hacemos en overlay para que aparezca por delante y en Scale with Screen size
* Quitamos EventSystem porque no usaremos botones ni audio.....
* Buscar en Kenny una plataforma en 2D como plataformer art deluxe y lo descargamos o lo cogemos la descargamos desde internet.
* Organizamos carpetas en Sprites Enviroment y Player
* Cogemos un personaje como el Base Pack/Player p1\_front y lo metemos como Player.
* Añadir otro Sprite 2D y le añadimos el p1\_front
* A nuestro Player le ponemos un RigidBody 2D para que no se caiga.
* Un corazón para coger vidas y objeto para la puntuación.
* Script de MarcianoScript para que se mueva en horizontal nuestro Player.
* Crear en el Canvas un Texto y no descargamos la fuente desde el google fonts y se lo asignamos.

NOTA: Descargas Fuentes: google Fonts y fontsquirrel

En el personaje vamos a poner los puntos y la vida

* PlayerScript:
* Variable entera vidas, vidasMaximas.
* Movimiento en Veritcal y Horizontal con las Axix al personaje.
  + Si va hacia arriba
* La puntuación se tiene que actualizar Método de ModificarPuntuacion
* Agregramos un Tag a nuestra Calabaza que será Calabaza y cuando hago un collisionEnter 2D incrementopuntuacion y destroy de la Calabaza.

NOTA: Lo mejor será hacerlo en la calabaza pero lo dejamos así.( esto lo hicimos en los enemigos)

* Ahora tenemos que poner un TEXT en la pantallas y para ello tenemos que tener una referencia del text
* Tendré que arrastrar el texto del Canvas a la variable que hemos creado en el Player.
* Si hacemos movimientos Físicos tendremos que modificar el Update a FixedUpdate

Vamos a duplicar plataforma para hacer que nuestro personaje salte y la ponemos a distinto nivel hacia arriba.

Comportamiento Jugador

Cuando le demos a la Tecla Arriba podemos hacer:

* + Subir Infinita NOOO
  + Subir hasta que subimos a un limite
  + Salto y hasta que no toque el suelo no puede volver a saltar.
  + Hacemos muñeco y ponemos un GO vacío en los Pies y le hacemos un overlapSphere muy pequeño y comprobamos si estoy chocando con el suelo
  + Doble Salto NOOO…….
* GO Vacío en los pies del Player.

Mirar programas

TILES

* Quitamos todos los OBJ y solo dejamos el Player y el Canvas.
* Cogemos del Platformer-art-completed-pack-0 que lo tendremos descargado anteriormente y nos vamos a la carpeta /BasePack/Tiles:
* Y cogemos Grass, liquid, lock. Y los arrastramos a una carpeta que se llama Tiles.
* Crear componente 2D TileMap (Grid). Crea una rejilla y un mapa para los azulejos
* Abrimos una ventana dentro del Editor de Window/Tile Palette
* Creamos la paleta que se llame Plataforma y la guardamos en una carpeta que se llame \_GameAssets/Paletas
* Creamos otra carpeta en \_GameAssets/Tiles para guardar todo
* Selecciono todos los \_GameAssets/Sprites/Tiles y los arrastro a mi paleta y me aparecerá para que los guarde en \_GameAssets/Paletas o TILES ¿?????
* Cojo un tile de la paleta (herramienta3) y lo arrastro a mi cuadrado pero no me ocupa todo.
* El Tile ocupa 70. Selecciono todos los Sprites/tiles y ponemos 70 en vez de 100.
* A dibujar nuestra plataforma.
* No se manejan como GO se puede hacer un colider, pero no se puede interactuar con lo GO. Podría arrastrar un Sprite pero no está encajado en el TileMap.
* Puedo hacer una brocha como un pintador de GO.
* Se nos cae el personaje:
* Asi que hay que agregarle un collider.
* Añadir al TileMap un componente TileMapCollider 2D. Nos ha puesto un colider a cada uno de los Tiles
* Composite Collider 2D nos genera el RigidBody y lo marcamos como Static. Lo que hace es agrupa todas las geometrías juntas.
* Y en el Tile Map Collider chequear el Used by Composite.
* Vamos hacer el salto PlayerScript
* Comentamos el EstaEnSuelo que está basada en el Tag
* Vamos hacer una basada en los Layer.
* Ponemos un Physcis.OverLapCicle y le añadiremos como otro parámetros la Capa con la que colisionara nuestro personaje
* Agregar una Capa Suelo.
* Al player en Unity le ponemos el Layer Suelo y en TileMap le ponemos en la CapaSuelo.
* Composite Collider le ponemos un Polígonos en Geometry Type
* Vamos hacer que salte con el espacio.
* Podríamos hacer que la cámara sea hija del Player. Pero lo dejamos como estaba.
* Bajarnos de la tienda de Shop old (tienda antigua cámara roja 2017): Cinemachine unity essentials Script para que arrastra la Main Camara no crea ninguna cámara es una cámara virtual.
* Aparece un menu nuevo de Cinemachine, movimientos, para que te sigan...
* Create 2D Camera y aparecerá un nuevo icono CM vcam1:

Vamos a configurarlo:

* Follow nos seguira al Player
* Dead Zone Witdth y Height: Aparece un cuadradito pequeño que es como mi zona de confor, la zona muerta, se puede minimizar el tamaño para que me mueva y la cámara no se mueve en esa zona
* Zona Soft Azul cuando me salgo de la dead zone me sigue la cámara pero va despacio.
* Zona de la cámara sígueme a saco para que siga la cámara a tope.
* LookaheadTime y el Smoothing para que se mueva más la cámara con nuestro muñeco
* Damping la cámara no deja de salirte de tu zona muerta a 0 y si le pones > 0 te atrae como un muelle
* Lens ortografic size para ajustar la escena para que se vea la zona de juego.
* Look At, tileMap ¿????
* Game Windows Lines para desactivar y que no se vean tantas líneas.

COGER COSAS

Para colisionar necesito un GO podemos coger los sprites y hacerlo GO pero no lo vamos hacer con las Brochas.

* En Tile Paletes hay Default Brush y vamos a crear una brocha para generar un GO.

En 2017 hay que añadirle otra cosa más

ht[tps://github.com/Unity-Technologies/2d-extras-2017 B](tps://github.com/Unity-Technologies/2d-extras-2017%20B)ranches y coger 2017 La descargamos el Zip y luego la descomprimos y cogemos la carpeta TileMap y lo arrastramos a \_GameAssets

* Crear una carpeta \_GameAssets/Brushes
* Crear una brocha Prefab Brush.
* Habrá que hacer reglas de dibujo, va haciendo como preguntas de que tienes dibujado a la derecha o a la izquierda.....Se puede hacer un generador de plataformas. Te sirve para no trabajar de más pero a lo mejor el trabajo es arduo.
* Crear Prebaf Brush que se llame Pinchos
* Meter un Sprite a la Escena y le hacemos Prefab y la quitamos de la escena.
* Vamos al Brush Pinchos y le arrastramos el prefab.
* En el Tile Palete aparece un Default Brush y aparece nuestro Pinchos y lo seleccionamos y si seleccionamos el pincel y pintarlo me aparecerán los GO en el TileMap.

Zona de pinchos para que te quite vidas cuando pases por encima.

Para hacer descargas: itch.io Download Gothicvania Patreon's Collection e

Dibujar un Mapa y todo con Brush

* Zona de pinchos cuando pase el player una fuerza hacia atrás y le quitamos vida.
* Corazones + vida
* Cajas de puntos para incrementar puntos

3 corazones array GO y se desabilitan.

// Dia5 Noviembre

Crear carpeta Atlas:

* Window/SpritePacker open proyect aparece en el Inspector y pone Sprite Packer always Enabled para confirmar que esta asi
* Nos vamos a la carpeta del Atlas y creamos un Sprite Atlas que se llame AtlasJuego
* En AtlasJuego hay un + e Incluimos las carpetas que necesitemos. Y le damos a Pack Prewiew.
* Opciones:
* Incluirlas en el ejecutable
* Rotaciones para que incluya la rotaciones
* Packing para que los ponga muy juntos
* Padding espacios
* Genera un Sprite paralelo chiquitin
* Filter mode si se pixela y se ven los border sucios, si se ponen en point quedan los bordes cuadrados y se verán mas cuadrados y más nítidos.
* Si vemos que no caben todos los Sprites se pueden generar mas Atlas

NOTA: tile Palete el 5 hace un Pick y con la tecla CTRL, arco un área y cuando suelto me hace un copy de la plataforma seleccionada. También se puede borrar un área.

EFECTORES

Crear otro TileMap que se llame TileMapAgua.

El TileMap es una serie de Tiles agrupados en un GO. Me puedo crear varias TileMaps como si fueran capas.

* Los Efectores son una colección de componentes y se va a generar una fuerza física:
* PlatformEffector se atraviesa desde abajo la plataforma pero no desde arriba
* SurfaceEffector2D es una cinta deslizadora se marca la velocidad y el sentido, te transporta.
* PointEffector2D te atrae o repele los objetos
* BouyancyEffector2D para que algo flote.
* AreaEffector2D
* El tileMap esta vacío y le agregamos unos efectos
* Añadir el Bouynacy Effects2D, protesta porque necesitamos un Collider.
* Añadir TileMap Collider. Colisionador para los TileMap. Genera un collider para cada uno de los Tiles.
* El Boynacy protesta pero le marcamos el Use by effector y también habrá que marcar el trigger.
* Vamos hacer que nuestro jugador se vaya hacia atrás.
* Si yo no estoy pulsando la fuerza de la física se ejecutaran sin más:
* Donde no estoy saltando aplico la xPos \* speed hay que añadírselo.

Mathf método matemático que me da el valor absoluto de una variable > 0.01f se le aplicaran las fuerzas físicas.

* Para que el Player flote también hay que cambiar el Surface Level en valores más pequeños. Aparecerá una línea para ajustar hasta que nivel quieres que baje.
* El Player tiene una masa pero se le puede dar un Use auto mass para que coja el que considere.
* TileMapAgua se puede poner un Flow (balanceo del agua)
* Vamos al TileMaPlataformas y le añadimos Plataform Efector 2D y protesta porque necesita un collider que se use como efector.
* Poner el Collider Composite como Use Efector.
* Y el tipo de Geometría le ponemos Polígonos.

NOTA - Tb hay Joins para las lianas.......

ANIMACIONES

* Cogemos al Flyfly1 y el Flyfly2 del platformer-art-complete-pack-0
* Crear una carpeta de Animation.
* Seleccionamos los dos sprites y lo ponemos en la Escena y me genera una animacion y lo guardamos en Animation.
* Si le damos a ejecutar ya la mosca aletea.
* Animator:
* Tiene un Any State es cualquier estado
* Entry estado de comenzar y cuando empieza me hace una animacion al moscardon. Podemos tener muchos estados de la animacion de la mosca: Volando, Muriendote, muerto.
* Y solo tendre un Animator por cada GO.

Tendremos un Animator y con tantos estados como animaciones tenga nuestro personaje.

* Le cambiamos el nombre a Volando (sprite que parece un Play).
* Window Animation:
* Seleccionamos nuestra GO Mosca y aparecerá en la Ventana de Animation nuestra animación.
* Si le cambiamos el Sample ira más despacio o más deprisa.
* Vamos hacer el Script para que vaya de Dcha-Izquierda.
* 2 puntos de referencia 2 GO vacío para los límites.
* Cogeremos la X.
* La altura GO limite dcho e izqdo es relevante.

PLAYER

* Añadir Componente Animator.
* Creamos una Animacion Andar.
* Nos cargamos el GO del inspector
* Asignamos el Animacion de andar al Componente Animator.
* Pero su comienzo será de quieto.
* Máquina de Estados:
* Any State (Cualquier estado cuando te mueres de repente.....)
* Exit para salir de las transiciones
* Entry es la entrada y le ponemos que sea el PlayerWalking.
* Coger un tile y vamos hacer que se sube y baja mediante una animacion. Nos evita tener que hacer Script es lo mismo.
* Le añadimos un componente Animator
* Creamos una AnimatorController en la carpeta de Animations.
* Abrimos la ventana de Animator y arrastramos el AnimatorController(Cuadraditos pequeños). Y se le pone una flecha entre el Entry y el Animator
* En el GO CandadoAzul que tiene un Animator le arrastramos el AnimatorController.
* Abrimos la Ventana Window/Animation. Tiene que estar seleccionado el GO.
* Agregarle una Propiedad. Seleccionar Transform, position
* Le damos a Grabar.
* Cogemos el GO y le cambiamos la posición y dejamos de grabar.
* Queda raro, asi que vamos hacer otra animacion que
* Nos ponemos en el 00:30 y le damos agregar otra posición el máximo que queremos que sea el máximo.

Animacion Pl*ayer*

* En Animator Tenemos que meter el PlayerWalking
* Hacemos 2 transiciones de parado a Andando y de andando a parado.
* Tenemos que poner una condición para que empiece andar.
* Vamos a Parametros que está a la izquierda de la ventana de Animator y le damos al +
* Le creamos un parámetro de tipo bool.
* Seleccionamos la línea del Animator y le añadimos una condición de andando = true
* Y hacemos otra transición de andando a parado y le añadimos condición andando = false.
* En el Script Player necesitamos cambiar de animación.
* Hay Exit Time desactivar para que comience la animación a andar.

Estructuras- enumeraciones

Cuando definimos una clase lo que hacemos es ampliar nuestro lenguaje, una de ellas es la enumeración. Contendrá los posibles valores que tiene el Estado.

Un Estado Player será en Pausa, AndandoDerecha.....Le voy a conceder un estado al Player, aunque puedo tener el Saltando y andando entonces tendría que crear otro tipo de estado.

Si hacemos una enumeracion inicializo el estado a EstadoPlayer.Pausa para inicializarlo a Pausa. Para cuando queremos cambiar los estados de algo (Encendido, apagado.....).

Si tenemos solo dos opciones nos haremos un boleano. Pero no es nuestro caso.

-Hemos arreglado para que vaya a la derecha-Izquierda?????

* Vamos hacer que la mosca me pique:
* Añadir un collider 2D a la mosca
* Programar el daño de la mosca al player (Mejor hacer la colisión en la mosca y el daño en el Player. Valdría también que el Player se haga el daño sin pasar por la mosca pero me gusta menos)
* Tengo una mosca y un Player son GO (un contenedor).
* La mosca tiene un transform. Sprite Renderer, Animator, Script Mosca y el Box Collider2D.

* El Player tiene: transform, Renderer, Rigidbody, Box Collider2D, Script.
* Hemos hecho un GO vacío que dependa de la mosca para la parte superior y si le toca a la mosca el player por arriba matara a la mosca.
* Y otro Parte inferior y quitamos el Box collider de la mosca. Y hacemos un Script de :
* Programo on collisionEnter esto me da una referencia contra el que he chocado.
* Comprueba con quien me estoy colisionando contra el Player.

Le hago un daño al Player y llamo un método público que está en Player.

* Le decimos que nos muestre el componente del PlayerScript.
* Player y programar el Hacer daño tendremos
* Vidas serán los corazones.
* Vamos a tener una barra de salud con una saludmaxima de 100. Cuando no tenga Salud a 0 le quitaría 1 vida.
* Mosca terminar el Script del Daño.
* Cuando choque el Player con la parte inferior le lanzara hacia atrás.

Como le estamos aplicando un velocity para que vaya hacia delante. Tenemos que hacer un estado de repelido para que no haga nada al teclado y volver activar cuando este en estado de Reposo.

Si va andando hacia la derecha la repulsa ser a la izquierda y al revés.

* Recogemos su RigidBody para hacer la fuerza
* Creamos un nuevo estado de Sufriendo y cuando le quitemos vida es estado lo ponemos Sufriendo y el el FixedUpdate le hacemos un return cuando el estado == Sufriendo.
* Falta cuando estoy quieto y me da la mosca y ver hacia donde me da la mosca y poner el contrario

VIDAS

* Creamos un panel, es un contenedor de cosas y vamos a meter un array de corazones y que se coloquen todas a la vez. Nos servirá para poner más vidas.
* De 300x50.
* Meter un componente Layout para distribuir los componentes
* Metemos las imagenes de los corazones
* Creamos un UIScript para que instancia de tantas imágenes como vidas tenga el jugador:
* Using UI.
* Array de imágenes. de 5 Vidas.
* El primero se llamara ImagenVida y lo convertimos en un Prefab y nos los cargamos del panel todas las imágenes de vida.
* Instanciamos las vidas en el Start.
* Ahora tenemos que colocar las instancias en el panel
* Recorremos el array.
* Y los hacemos hijos del panel para que se coloquen ya que tiene el Layout horizontal.
* El Script se lo asignamos al Canvas.

* Tendríamos que coger las vidas del Player y rellenar el array con ese valor (luego lo hacemos).

En nuestro Script del Player tenemos que recoger las vidas. Asi que hay que hacerlo con un nuevo método para recoger esa vida.

Esto se hace para que no nos equivoquemos. Y esto es lo más correcto.

* Haremos un GetVidas en PlayerScript
* Y la llamamos desde UIScript y rellenamos el Array con ese valor.

Examen2: Interfaz usuario, moviles, iluminacion e intelegencia.

Google play hay una aplicación unity remote 5 hay que instalarla en el móvil pero va lento. Le manda la señal al ordenador y va mandando como pantallazos al móvil.

IOS Unity remote

VIDA

Script UIScript

* MAL(al ponerlo en el Update): Lo pondremos en el Update. Tenemos dos opciones:
* Opcion1:
* Recorremos el array a partir de la vida que tengo. Si tengo 4 vidas tengo que quitar el 5.
* Opcion2:
* Es poner semitransparente las vidas.
* Cogemos un corazón
* BoxCollider2D OnTrigger2D
* Hacemos un Script para que nos de Salud CalabazaVidaScript.
* Y le ponemos un TriggerEvent.
* En el Player tendremos que hacer un método de IncrementoSalud
* Mirar Codigo.--------------------

- Destruir el corazón

Mirar Codigo.

CHECKPONTS/GUARDAR ESTADO con JSON

* Cogemos un Sprite (un cuadrado..)
* Añadimos un Boxcollider2d, que sea trigger y con un Script CheckPointScript con un triggerEnter2D.

PlayerPref es un API que nos va a permitir escribir y leer en un fichero.

Puede leer float, int, string.... Puedo recuperar el estado donde estaba. Se crea un objeto JSON con todo, se reinstancia todo y se reinstaura el juego a como se quedó.

Solo podrá acceder a los archivos del juego que ha generado los PlayerPrefs.

* Le ponemos el Script al CheckPoint y ejecutamos.
* El Fichero lo deja en el registro regedit desde la barra en windows:
* Lo deja en ....
* Lo leemos desde el Script del Player en el Start.
* Tenemos que controlar que no estén vacías por si es la primera vez que ha funcionado.
* Con solo preguntar por un parámetro así sabemos que lo tenemos relleno.
* Hay que validar si existen esas claves.
* Y ponemos estas posiciones en el Player.

NOTA: HAY UN PROGRAMA: FIREBASE (de Google) es una base de datos. Para hacer aplicaciones distribuidas. Sistemas de mensajería.

Tenemos BD y a Pablo y a Alex y están jugando a un juego y quieren hacer un nivel de chateria ( no en juegos en red, funciones online pero no juego en online).

Base de datos distribuidos en cloud, los ficheros son JSON y cuando Alex hace una modificación inmediatamente lo notifica a Pablo.

VAMOS HACERLO BIEN

* Creamos un Script GameControler.
* Esta clase tendría cosas relacionadas del juego NO a los GO.
* Almacenaremos la posición de checkpoint.
* Van a tener 2 metodos:
* StorePosition guardamos la positionx x ey
* GetPosition que retornara un Vector2 y cogemos el Script que pusimos en el Player.
* Hay que tener cuidado con las cadenas de caracteres, para ello habría que hacer constantes y así no nos equivocaríamos.

Lo que le dice al Sistema Operativo no va a ocupar ms de 4Bit Private para que no se vea desde fuera y constante para que no se pueda cambiar en tiempo de ejecución.

El Static yo puedo llamar a la clase sin hacer una instancia de la Clase.

EJ: Un método para aumentar la vida max de los malos:

* Se podría hacer uno a uno con u metodo para aumentar vida.
* Si la ponemos Static y la hacemos general y lo podemos llamar desde cualquier sitio y desde cualquier momento.

Rehacemos el código:

* En el CheckPoint comentamos lo de PlayerPrefs y llamamos a nuestro metodo static GameControler.
* En el Start del Player recogemos la posición y se la ponemos.
* Problemas obtención posición:
* La primera vez que ejecuta el juego no hay datos.
* Otra ejecución: No hemos pasado por el checkpoint y no hay datos.
* Hemos pasado por un checkpoint y hay datos:
* Gamecontroller nos tiene que dar la posición

/ SI NO LO ENCUENTRA DEVUELVE UN VECTOR2 A 0

- Tenemos que condicionar en el Start del Player

- Mirar codigo

CAMBIO ARMA

Vamos al Proyecto del Castillo

Vamos hacer que aparezcan el número de balas.... Lo vamos hacer más fácil.

* Cogemos 3 pistolas 500x500 en png y les ponemos Sprite 2d and UI Aplicar. Se podría hacer en un Array pero vamos hacer esto:
* Crear un Canvas:
* UI Imagen y la ponemos abajo en el centro.
* Y le arrastramos el Sprite de la Pistola.
* Duplicamos dos veces más y les asignamos a las pistolas
* Tengo varias imágenes y se podrían desactivar pero también puedo cambiar el renderizado.
* Nos cogemos el arma1 y dejamos activa y el resto desactivada el Image para que no aparezca la imagen.
* Esta asociada la seleccion del arma en el Player.
* Vamos hacer desde el Canvas que active o desactive el canvas.
* Creamos un Script HUDScript, va a gestionar la parte de l ainterfaz grafica:
* Le vamos a pasar un número al método Activar Arma para activar las armas.
* Primero hacemos un método DesactivarArmas y desactiva todo y luego activamos la que toca.
* Activamos desde ¿????

* Al Canvas hay que arrastrale las imagenes las que están en el Canvas
* En el Player hay que meter el Canvas que es el que tiene asociado el Script HUBScript.

ESTADOS ENEMIGOS

* En el moscardon nos creamos un Canvas para hacer una barra de vida o lo que sea.

UI se usan para Menus, energía (estados en general...) y para meter información en la zona de juego (dialogos, barra de vida....)

* Le ponemos a World Space: Aparece un transform para poder posicionarlo. Se podría hacer que nuestro personaje podría pulsar el botón.....
* Le vamos a poner en el Canvas un Slider (para marcar niveles).
* El Canvas lo ponemos en la Posición 0,0.
* El Slider lo escalamos a 0,01 en X e Y. Tiene que salir en el centro de la Mosca.
* Modificar la Posición del Slider de la Y para que este encima de la cabeza de la mosca. Le cambiamos el Width para que se ajuste más a la mosca.
* Background la ponemos en Rojo que es el que vaya apareciendo
* Fill Area de color verde. Es la cantidad de vida.
* El Slider se le da unos valores mínimos y máximos. Mínima de 0 y máxima de 100. Value es el máximo así que lo ponemos en 100, es lo más normal.
* Desactivamos el Interactable porque no lo vamos a modificar por código y para que no se vea esa bolita blanca???Preguntar a Fernando

* Vamos hacer que la barra por tiempos se vaya descontando. En mosca Script
* Método QuitarVida.
* Tenemos que coger el Slider de la mosca pero no con un Find

Cuando rota la Mosca rota la barra de vida y no queda bien habrá que arreglarlo ??????

* Con esto nos da opciones para hacer muchas cosas como: un teclado de un móvil...........lo que se nos ocurra.

PINCEL AUTOMATICO

2d-extras (Rule Tile). Lo hemos descargado anteriormente.

Nos ayuda para automatizar los brochazos y los azulejos. Todo esto se puede programar pero con el Rule tile vamos hacer unas reglas aunque está un poco limitada.

* EJEMPLO
* Carpeta nueva de TileRules.
* Crear Rule Tile
* Metemos un tile que no tenga bordes que es el que hemos dibujado en nuestro centro de plataforma pej:
* Añadimos otro que tenga una esquina de terminación a la izquierda y le ponemos la regla de flechita:
* Flecha verde es que tengo algo.
* Flecha roja no tengo nada.
* Abrir el TilePalette y seleccionado nuestra paleta de la Tiles con todo seleccionado (active Tilemap...) y arrastramos nuestro Rule tile a nuestra Tile Palete,
* Y ahora como siempre selecciono mi tile y pinto.