



# DIBUJANDO SPRITES

APRENDE A PROGRAMAR TUS PROPIOS VIDEOJUEGOS

BOOTCAMP UTN 2019

# SPRITE

Un sprite es la representación de una imagen o textura en la pantalla de juego

# SPRITE

Una misma textura (archivo de imagen) puede tener múltiples representaciones en la pantalla según:

- posición
- ángulo
- escala
- opacidad
- etc...

# FORMATO PARA TEXTURAS

- Conviene utilizar el formato PNG
  - Compresión **sin pérdida**
  - Soporta **transparencias**

# UTILIZAR SPRITES

1. Importar el módulo FlxSprite
2. Declarar variable de tipo FlxSprite
3. Inicializar (crear) el sprite
4. Agregar sprite a la escena

# UTILIZAR SPRITES

## 1. IMPORTAR EL MÓDULO FLXSPRITE

```
import flixel.FlxCprite;
```

# UTILIZAR SPRITES

## 2. DECLARAR VARIABLE DE TIPO FLXSPRITE

```
var hero: FlxSprite;
```

# UTILIZAR SPRITES

## 3. INICIALIZAR

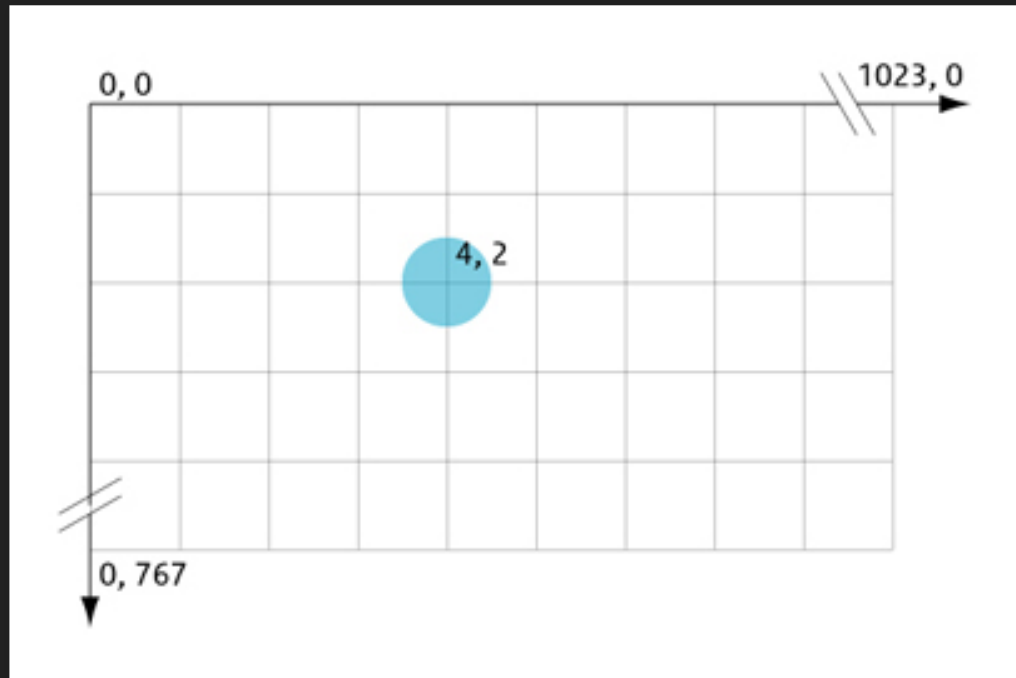
```
hero = new FlxSprite(100, 100, "assets/images/hero.png");
```

- Los sprites son *objetos*
- Los objetos se inicializan con **new**
- Se pasan parámetros al crear
  - x: Float
  - y: Float
  - textura: String



# UTILIZAR SPRITES

## POSICIÓN EN LA VENTANA



# UTILIZAR SPRITES

## POSICIÓN EN LA VENTANA

- El origen es la esquina superior izquierda
- El eje  $x+$  se extiende hacia la derecha
- El eje  $y+$  se extiende hacia abajo

# UTILIZAR SPRITES

## 4. AGREGAR SPRITE A LA ESCENA

```
add(hero);
```

- La escena guarda una **lista** de sus actores
- Una vez que agregamos un sprite a la escena, ella se encarga de **dibujarlo y actualizarlo automáticamente**

# EJEMPLO COMPLETO

```
import flixel.FlxState;
import flixel.FlxSprite;

class PlayState extends FlxState
{
    var hero: FlxSprite;

    override public function create():Void
    {
        super.create();
        hero = new FlxSprite(100, 100, "assets/images/hero.png");
        add(hero);
    }

    override public function update(elapsed:Float):Void
    {
        super.update(elapsed);
    }
}
```

[https://github.com/pabab/curso-haxeflixel/raw/master/dist/ejemplos/simple\\_sprite.zip](https://github.com/pabab/curso-haxeflixel/raw/master/dist/ejemplos/simple_sprite.zip)

# LA CLASE ASSETPATHS

```
hero = new FlxSprite(100, 100, AssetPaths.hero__png);
```

Con el autocompletado es menos probable equivocarse el nombre del archivo

# PROPIEDADES DE FLXSPRITE

Propiedad	Tipo	Descripción
x	Float	Coordenada x del vértice superior izquierdo del sprite
y	Float	Coordenada y del vértice superior izquierdo del sprite
angle	Float	Rotación del sprite (en grados)
alpha	Float	Opacidad. 1: opaco. 0: transparente. 0.5: 50% transparente
scale.x	Float	Escala en x del sprite. 1: tamaño original. 2: doble tamaño. 0.5: la mitad
scale.y	Float	Escala en y del sprite. 1: tamaño original. 2: doble tamaño. 0.5: la mitad
flipX	Bool	Si la textura debe dibujarse invertida horizontalmente
flipY	Bool	Si la textura debe dibujarse invertida verticalmente
visible	Bool	Si el sprite es o no visible

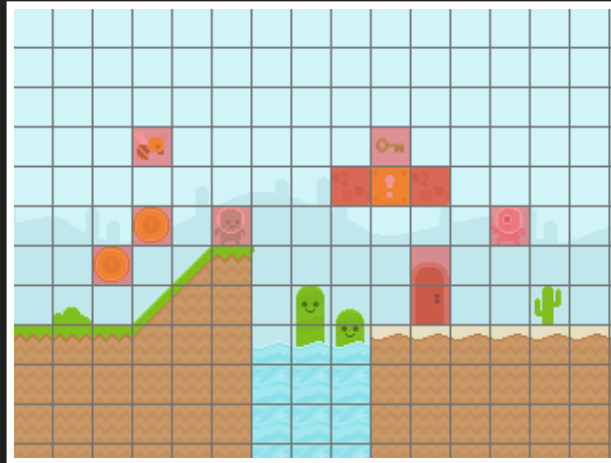
<https://api.haxe.flixel.com/flixel/FlxSprite.html>

# ARMAR LA ESCENA

# DESAFÍO #1

## ARMAR LA ESCENA

A modo de ayuda, el ejemplo contiene algunos sprites ya agregados. También puede utilizarse la imagen *help.png* a modo de guía para ubicar los sprites. El tamaño de cada celda de la grilla es de 21 px.



[https://github.com/pabab/curso-haxeflixel/raw/master/dist/desafios/pintar\\_escena.zip](https://github.com/pabab/curso-haxeflixel/raw/master/dist/desafios/pintar_escena.zip)