

## PRÁCTICA 1

### MODELADO DE OBJETOS 3D, ESTÁTICOS Y ARTICULADOS

#### OBJETIVOS

- Aprender a configurar el ordenador para desarrollar en Three.js con Javascript
- Comprender la estructura del software y los distintos módulos de una aplicación gráfica
- Aprender a diseñar e implementar diversos métodos y técnicas orientadas a modelar objetos 3D, estáticos y articulados.

#### CONFIGURACIÓN DEL ORDENADOR PARA PROGRAMAR CON THREE.JS

Se debe hacer tanto si se usa el ordenador del aula como si se usa el propio portátil del estudiante. Lo que se explica es válido tanto para Linux como para Windows.

- Descargar de la plataforma el Código para las prácticas, un archivo denominado `ejerciciosThree.zip` y descomprimirlo en la cuenta del estudiante, en la carpeta donde se vaya a trabajar.
  - Se tendrá la siguiente estructura de carpetas  
`ejerciciosThree/`  
`ejerciciosThree/libs/`  
`ejerciciosThree/utils/`  
`ejerciciosThree/imgs/`  
`ejerciciosThree/models/`  
`ejerciciosThree/introduccion/`  
`ejerciciosThree/grapadora/`
  - Las carpetas `libs` y `utils` contienen bibliotecas que son necesarias para los ejercicios.
  - La carpeta `imgs` contiene las imágenes necesarias para los ejercicios.
  - La carpeta `models` contiene un modelo 3D que puede ser incorporado a un escenario
  - Las otras 2 carpetas son 2 aplicaciones:
    - La carpeta `grapadora` contiene los archivos `html` y `javascript` del ejemplo de la grapadora que se explica en clase de teoría.
    - La carpeta `introduccion` contiene los archivos `html` y `javascript` del primer ejercicio.
    - **Cada ejercicio deberá ir en una nueva carpeta similar a éstas.** Con su correspondiente archivo `html` y los archivos `javascript` necesarios.

- Para ejecutar las aplicaciones en un navegador se necesita un servidor web local. Se puede usar cualquiera que se esté usando para otras asignaturas. Si no se tiene ninguno, se puede usar uno que proporciona python. Si no se tiene python ya instalado en el ordenador, instalarlo. En los ordenadores del aula ya está instalado. Si el estudiante usa su propio ordenador en Linux seguro que se encuentra fácil en los repositorios de la distribución que esté usando el estudiante. Si el estudiante usa Windows o Mac lo puede descargar de <https://www.python.org/downloads/>
- Una vez instalado:
  - Abrir una terminal de órdenes y situarse en la carpeta `ejerciciosThree`
  - Lanzar un servidor web local ejecutando una de las siguientes órdenes. La apropiada depende de la versión de python que se tenga:
    - `python -m SimpleHTTPServer`
    - `python -m http.server`
- Abrir el navegador y cargar la página <http://localhost:8000>
  - **Ya se puede abrir cualquier aplicación que se tenga en la carpeta de trabajo navegando hacia dicha carpeta.**
- **En firefox puede dar un error al ejecutar los ejemplos.** Si vas a usar este navegador entra en la configuración y pon a `true` la variable de configuración `dom.importMaps.enabled`

## EJERCICIOS

La siguiente relación de ejercicios tiene como objetivo que el estudiante practique los diferentes conceptos y técnicas vistas en clase para modelar objetos 3D, estáticos y articulados, implementándolos con la biblioteca `Three.js`.

Cada ejercicio se dedica a una técnica distinta y va acompañado por un vídeo que muestra ejemplos de objetos modelados por dicha técnica.

En cada ejercicio, los estudiantes deben modelar algún objeto 3D siguiendo la técnica concreta que se propone en el ejercicio con estos 2 objetivos:

- Aprender a usar dicha técnica para modelar objetos 3D
- Modelar objetos 3D para incorporar al juego final

**Los ejercicios no tienen que entregarse ni serán evaluados como parte de la práctica 1.**

**La evaluación del aprendizaje de modelado 3D  
se llevará a cabo mediante un examen**

## EJERCICIO 1

Tanto el ejemplo de la grapadora como el ejercicio 1 muestran una aplicación mínima usando `Three.js` para la programación gráfica y la biblioteca `dat.gui.js` para la interfaz gráfica de usuario (GUI).

Son los únicos ejemplos en los que se proporciona el código fuente. El estudiante debe estudiarlos y comprender su diseño.

### Descripción de las aplicaciones de ejemplo

Se trata de aplicaciones sencillas (la grapadora y el ejercicio 1) pero que incluyen bastantes aspectos de interés además de servir de marco para el desarrollo del resto de ejercicios. En concreto:

- Muestran cómo se aplican las transformaciones geométricas básicas a una figura, como son el escalado, rotación y traslación. Las cuales, y **dentro de un mismo nodo**, con independencia del orden en el que se escriban en el código, **siempre** se aplican en el mismo orden. Primero el escalado, después las rotaciones, y por último la traslación. Y las rotaciones se aplican siempre en el siguiente orden: primero se aplica la rotación sobre el eje Z, luego sobre el eje Y, y por último sobre el eje X. **Insisto en este aspecto porque es una diferencia importante con respecto a vuestra experiencia con OpenGL en Informática Gráfica, y suele ser una fuente de error al principio en estas prácticas.**
- Muestra cómo crear un material basado en color (el de la caja) y un material basado en una textura (el del suelo).
- Muestra cómo añadir luces a la escena, en concreto una luz ambiental y una luz focal (spotLight).
- Muestra cómo crear una cámara en perspectiva y cómo controlarla con el ratón. El botón izquierdo realiza órbita, la rueda hace zoom, y el botón derecho reencuadra la imagen.
- Muestra cómo capturar el evento de cambio de tamaño de la ventana del navegador para ajustar la cámara en consecuencia.
- Muestra cómo crear una interfaz gráfica de usuario con la biblioteca `dat.gui.js`.

### Diseño de la aplicación

Sobre el diseño de estas aplicaciones de ejemplo, consultar el material que se explica en clase de teoría relativo a la grapadora.

**El código de las aplicaciones está documentado**, con comentarios que ayudan a la comprensión de los mismos. **Se recomienda al estudiante su lectura detallada.**

## EL RESTO DE EJERCICIOS

El resto de los ejercicios se encuentran en la relación adjunta. Para cada ejercicio se proporciona una posible solución en forma de vídeo. Los vídeos se encuentran en Prado.

No obstante, como se ha comentado, no se trata de obtener lo que muestran los vídeos, sino **el objetivo es aprender a modelar objetos 3D y crear objetos para incorporar al juego final.**

## EJECUCIÓN, MENSAJES DE ERROR Y DEPURADOR

**Para ejecutar el programa de un ejercicio**, solo hay que navegar hacia la carpeta que contiene dicho ejercicio. Y si ya tenemos cargada dicha página, solo hay que recargarla. Pero, **IMPORTANTE**, hay que realizar la recarga de la página forzando a que se lean de nuevo los archivos y que el navegador no use las versiones que tiene en su caché.

Si se hace así puede ocurrir que se haya hecho un pequeño cambio en el código fuente y que, al recargar la página, dicho cambio no se muestre porque el navegador ha releído el archivo de su caché y no del disco duro, donde sí están los cambios. Averiguad en el navegador que estéis usando, cómo hay que recargar la página para forzar a que se lean de nuevo los archivos y no se use la caché.

Otro aspecto importante es tener siempre abierta la **consola de las herramientas del desarrollador**. En Chrome se obtiene con F12, averiguad cómo se obtiene en el navegador que estéis usando. Es en esa consola donde se mostrarán los mensajes de error que se produzcan al ejecutar un programa. Como sabéis, es fundamental saber cuál es el error y en qué archivo y línea se ha producido para poder solucionarlo.

Si para la detección y corrección del error tenéis que recurrir al **depurador**, en esta url tenéis tutoriales sobre el uso del depurador al programar en Javascript en diversos navegadores:

- <https://threejs.org/manual/#en/debugging-javascript>

## EVALUACIÓN

Como se comentaba en otro apartado de este guion, los ejercicios no requieren ser entregados.

La evaluación de esta parte se llevará a cabo mediante un **examen en ordenador sobre modelado 3D**.