**Preguntas ejercicio en clase 27/04/23**

1. **¿Qué es la vista perspectiva y en qué situaciones se aplica?**

* La perspectiva busca lograr crear una ilusión de profundidad en una superficie tridimensional, básicamente busca que los objetos que estén más cerca parezcan más grandes de lo que realmente son, mientras que los objetos más lejanos se verán muy pequeños, de esta manera se crea esa ilusión de profundidad.

Además, es útil para aplicar en situaciones en las cuales se requiera una representación de una escena 3D, muchos ejemplos podemos verlos en películas, videojuegos, etc. También es muy usada en el ámbito del dibujo técnico, ya sea para el plano arquitectónico o sencillamente planos ingenieriles.

1. **¿Qué es la vista ortográfica y en qué situaciones se aplica?**

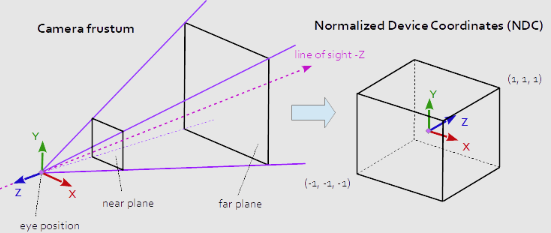
* La vista ortográfica busca mostrar una proyección cubica del objeto, esto con el fin de poder observar que los objetos tienen el mismo tamaño sin importar la distancia a la que se encuentren, de esta forma, se elimina la ilusión de la vista perspectiva.

Se aplica en situaciones en las cuales se requiere una representación precisa de los objetos en una escena, nuevamente es utilizado en el ámbito de la arquitectura o ingeniería, ya sea en la creación de planos que permitan ver la longitud de las líneas para poder replicar exactamente lo necesitado, y así poder crear representaciones exactas en una escena 3D.

1. **¿Cómo se calcula una vista en perspectiva en la computación gráfica y qué parámetros se utilizan en su cálculo?**

* Para calcular una vista perspectiva en computación gráfica, se requiere de algunos parámetros que definen la posición y orientación de la cámara y de la misma escena, estos parámetros son: La posición de la cámara, la cual se refiere a la posición de la cámara en el espacio 3D, el punto al que apunta la cámara se llama punto de mira, el ángulo de visión, es el ángulo de apertura de la cámara, los planos de recorte cercano y lejano, se refieren a las distancia minina y máxima de la cámara a los objetos y la relación de aspecto, que es la relación entre el ancho y alto de la imagen.

Dichos parámetros son usados para calcular la matriz 4x4 de proyección en perspectiva, en donde cada punto en la escena 3D se proyecta sobre un plano 2D mediante las líneas de proyección que se convierten en puntos de fuga, estos puntos se encuentran en el ángulo de visión de la cámara, que se extienden hasta el punto de mira, finalmente, este proceso nos crea la sensación de profundidad y perspectiva en nuestra escena.



1. **¿Cuáles elementos intervienen en la configuración de las vistas referidas y qué significado tiene cada uno de ellos en THREE.js?**

Para la vista perspectiva, usamos el siguiente código



Allí intervienen los siguientes elementos:

**Fov:** Campo de visión de la cámara en vertical

**Aspect:** Aspecto frustum de la cámara

**Far:** Cámara frustum plano lejano

**Near:** Cámara frustum plano cercano

Para la vista ortográfica, usamos el siguiente código:

****

Allí intervienen los siguientes elementos:

**Left:** Plano izquierdo

**Right:** Plano derecho

**Top:** Plano superior

**Bottom:** Plano inferior

**Near:** Plano cercano

**Far:** Plano lejano

**BIBLIOGRAFIA**

* EsDesign(2020), *Perspectiva* [**https://www.esdesignbarcelona.com/actualidad/ilustracion/perspectiva-dibujo-tecnico**](https://www.esdesignbarcelona.com/actualidad/ilustracion/perspectiva-dibujo-tecnico)
* TheCube3DAnimation(2019), *Vista perspectiva y ortográfica* [**https://www.thecube3danimation.com/blog-1/2019/2/6/vista-en-perspectiva-u-ortogrfica-al-esculpir**](https://www.thecube3danimation.com/blog-1/2019/2/6/vista-en-perspectiva-u-ortogrfica-al-esculpir)
* HektorDocs(2022), *Matriz de proyección.*

[**https://docs.hektorprofe.net/graficos-3d/17-matriz-de-proyeccion/**](https://docs.hektorprofe.net/graficos-3d/17-matriz-de-proyeccion/)

* GeeksForGeeks(2018) *Perspective Projection and its Types* [**https://www.geeksforgeeks.org/perspective-projection-and-its-types/**](https://www.geeksforgeeks.org/perspective-projection-and-its-types/)
* Three.js. *Perspective Camera* [**https://threejs.org/docs/#api/en/cameras/PerspectiveCamera**](https://threejs.org/docs/#api/en/cameras/PerspectiveCamera)
* Three.js. *OrthographicCamera*

<https://threejs.org/docs/#api/en/cameras/OrthographicCamera>