

Explicación PAG 7

Clases Añadidas

-PAGMaterial

Para empezar necesitábamos una clase para almacenar y manipular los distintos parámetros de los materiales de los objetos . Para ello he creado la clase PAGMaterial que engloba esos parametros. Esta clase está directamente relacionada con el modelo. En mi caso es el modelo el que va a ser creado directamente con un material , por lo que he creado en la clase modelo un constructor a la que le pasamos un material.

-PAGLuz

Ahora necesitábamos almacenar los distintos parámetros de intensidades de las luces, por lo que creamos la clase PAGLuz con los parámetros que necesitan TODAS las luces , para así poder hacer una herencia.

-PAGfoco.

Hereda de la clase PAGLuz y simplemente añadimos los parámetros especiales del foco (dirección ,posición y ángulo)

-PAGLuzPuntual

Hereda de la clase PAGLuz y simplemente añadimos los parámetros especiales de la luz puntual (posición).

Modificaciones en otras clases

-PAGValues

Para empezar cambiamos en PAGValues para añadir una nueva enumeración que van a necesitar tanto las luces como los shadersPrograms, el PAGtipoLuz , con esto podremos saber de qué tipo son las luces y los programs.

-PAGRenderer

En esta clase vienen las principales modificaciones para empezar pasamos de tener un solo programa a varios de ellos , por lo que creamos un vector con punteros a shadersPrograms. Modificamos la forma en la que añadimos los programas , para ahora añadirlos al vector. Creamos también la relación entre las luces y el renderer , para ello añadimos otro vector de luces y los métodos correspondientes para añadirlas al vector.

Finalmente creando una función que nos permite trabajar con el multipagrendering y modificamos el refreshCallback para recorrer los vectores de luces y programas y si ambos tipos coinciden entrar a la parte de asignar los uniforms correctos a cada tipo.

-Main

Creamos un material para el objeto , los distintos shaders programs para cada tipo de luz , y todas las luces que queramos haciendo uso de los métodos anteriormente mencionados del `pagrender`.

-PAGVao

Añadimos un nuevo método para insertar triángulos con sus normales calculadas al vector, y en el `activarBuffers()` le decimos que ahora tenemos tanto vértices , como normales.