

Programozói dokumentáció

A triviális konstruktorok, valamint a getter-setter függvények nincsenek felsorolva.

Vector2

Tagok:

- double x, y

Funkciók:

- new Vector2(awt.Point): Konverzió a két 2D vektor típus között, mivel ez többször is előfordul a kódban.
- Vector2 add(Vector2): Összeadja az input vektort önmagával, és visszaadja az eredményt.
- Vector2 sub(Vector2): Kivonja az input vektort önmagából, és visszaadja az eredményt.
- Vector2 mul(double): Mindkét koordinátát megszorozza az inputtal, és visszaadja az eredményt.

Line

Tagok:

- Vector2 p1, p2
- Color color

Funkciók:

- static String getColorHex(awt.Color): Konvertálja az input színt “#AARRGGBB” szöveges hexadecimális formátumba.
- static Color getColorFromHex(String): Konvertál egy “#AARRGGBB” formátumú szöveget egy Color objektumba.

Scene

Tagok:

- List<Line> lines
- String version (Ha valaha is olyan változtatásokat vezetnek be, ami megváltoztatná az adattípusokat, és ezzel a fájlformátumot, ennek a segítségével megkülönböztethetőek a különböző verziók, és korábbi Scene-eket is lehet használni.)

Funkciók:

- addLine(Line): Hozzáadja az input Line-t a listához.
- removeLine(Line): Törli az input Line-t a listából, ha benne van.

EditorWindow

Tagok:

- WorkAreaPanel workArea
- Color selectedColor

Funkciók:

- JPanel createColorPickerPanel(): az ablakhoz tartozó szíkválasztó panel létrehozásához szükséges kód
- static JTable getColorJTable(): A szíkválasztó panelhez tartozó táblázat létrehozásához szükséges kód
- void clearScene(): Lecseréli a jelenleg betöltött Scene-t egy üressel
- void saveScene(): Elmenti a jelenlegi Scene-t egy XML fájlba
- void loadScene(): Feldob egy JFileChooser popup-ot, és az alapján betölt egy Scene-t egy XML fájlból.
- static void main(String[]): A fő main függvény, ami megjelenít egy Editor ablakot.

WorkAreaPanel

Tagok:

- Scene currentScene
- Line selectedLine
- double zoomFactor
- Vector2 offset
- Color paintColor
- EditorInputHandler inputHandler

Funkciók:

- Vector2 pointFromViewToSceneSpace(Vector2): Transzformálja az input vektort képernyőkoordinátákból Scene koordinátákba
- Vector2 pointFromSceneToViewSpace(Vector2): Transzformálja az input vektort Scene koordinátákból képernyőkoordinátákba
- double pointToLineDistance(Vector2, Line): Kiszámolja az input pont és vonal távolságát
- void paintComponent(Graphics): Iteratívan kirajzolja a Scene vonalait, megfelelően transzformálva

EditorInputHandler

Tagok

- WorkAreaPanel workAreaPanel
- Vector2 firstPoint
- Vector2 mouseDelta

Funkciók

- void handleMousePress(MouseEvent):
- void handleMouseDrag(MouseEvent):
- void handleMouseWheel(MouseWheelEvent): Az egér inputokat kezelik, a felhasználói kézikönyvben van ezekről több információ.
- void selectLineAtPoint(Point): Kiválasztja a képernyőkoordinátákban megadott inputhoz legközelebb eső vonalat, ha az egy adott minimum távolságnál közelebb van, mint a legkisebb távolság

RendererWindow

Funkciók:

- static void main(String[]): Alternatív main függvény, ha egyből a Renderer-t szeretnéd elindítani.

Camera

Tagok:

- void updateCamera(Map<Integer, Boolean>): Egy időegységgel előrelépteti a kamerát, a jelenleg lenyomott billentyűk alapján.

RendererPanel

Tagok:

- int shaderProgram
- int[] vao, vbo, ebo, ssbo
- GLCanvas glCanvas
- ShaderUtil shaderUtil
- Camera camera
- RendererInputHandler inputHandler
- Timer frameTimer
- Scene scene

Funkciók:

- void initBuffers(Scene scene, GL3 gl): Inicializálja az OpenGL instance-hez tartozó buffereket
- void init(GLAutoDrawable): Compile-olja shadereket és meghívja az initBuffers()-t
- display(GLAutoDrawable): Updateli a kamerát, elküldi az új kamera paramétereket a shader-nek, majd lefuttatja a shader-t.

RendererInputHandler

Tagok:

- Map<Integer, Boolean> keyStates
- Camera camera

Funkciók:

- void initializeKeyStates(): Feltölti a keyStates map-et a használandó billentyű id-kkel.
- void keyPressed(KeyEvent):
- void keyReleased(KeyEvent): Igazra/hamisra állítja a megfelelő flag-et a keyStates-ben.

SceneFileHandler

Funkciók:

- static void saveSceneToXML(Scene, File): Elmenti a Scene-t a fájlba, XML formátumban.
- static Scene loadSceneFromXML(File): Betölt egy Scene-t a fájlból.

ShaderUtil

Funkciók:

- `int initShaders(GL3 gl, String vertexShaderFile, String fragmentShaderFile)`: Beolvassa és compile-olja a két shader-t, majd beállítja, hogy az OpenGL instance-hez tartozzanak
- `String loadShaderSource(String)`: Egy adott elérési útból visszaadja a fájl tartalmát.
- `int createAndCompileShader(GL3 gl, int shaderType, String shaderSource)`: Compile-olja a szöveggént megkapott shader kódot, aminek a típusát a `shaderType` határozza meg.
- `void checkLinkStatus(GL3 gl, int shaderProgram)`: Visszajelzést ad a shader linking sikerességéről.