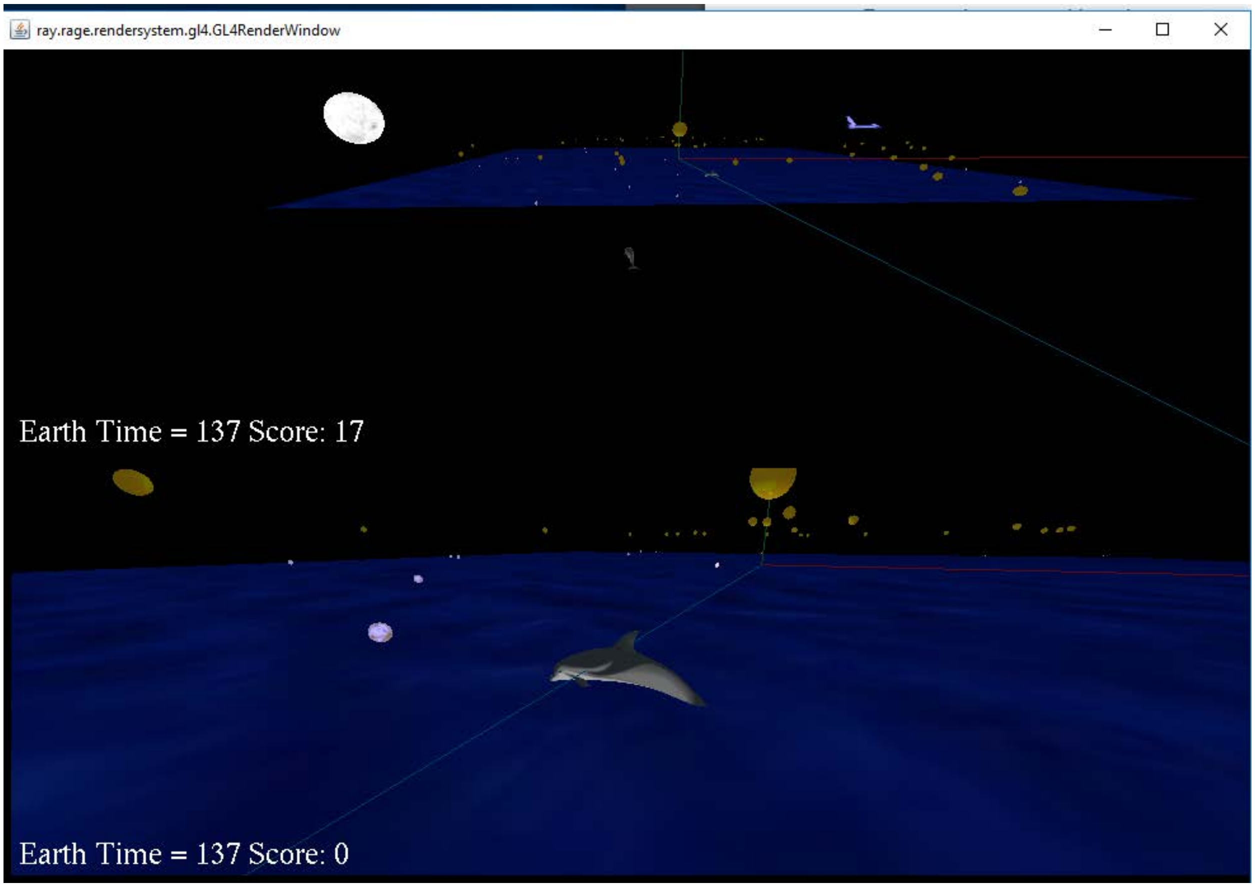
Gadfrey Balacy

Typical Scene:



Earth Time = 137 Score: 17

Earth Time = 137 Score: 0

How to compile and Run program in the Command Line:

Compile:

First Command: javac myGameEngine*.java

Second Command: javac a1*.java

Run:

Java –Dsun.java2d.d3d=false a1.MyGame

Bat Files:

Instead of running the game from the command line.

First Double Click: compile.bat

Then Double Click: run.bat

How Game is Played:

The game is played by two players. Using a gamepad for player1 (player 1 is the top screen). While player two (player2 is the bottom screen) plays the game by keyboard. The goal of the game is to be the fastest to get 21 coins and win the game. How you get the coins is by hitting or be in close proximity. Once the coin is hit or in close proximity with an avatar. The coin float away slowly and changes the speed in which the orbiting spheres orbit. Each coin is one point.

The Input For Player 1 (GamePad Logitic Dual Action)

Toggle: This provides a means to be able to control the avatar or just the node camera.

Input: RB in the Logitech Game Pad

If toggle = false

Dolphin Node Inputs GamePad: RT – move the dolphin forward. LT – move the dolphin backward. X-Axis – yaw the dolphin left or right. Z-Axis – move the orbit camera either left or right in a circular fashion targeting the dolphin-node. RZ-Axis – zooms in and out the camera in relation to the target.

If toggle = true

Z-Axis – move the orbit camera and also move the dolphin in yaw fashion either left or right in a circular fashion.

The Input For Player 2 (KeyBoard)

Toggle: This provides a means to be able to control the avatar or just the node camera.

Input: Spacebar

If toggle = false

Dolphin Node Inputs KeyBoard: W – move the dolphin forward. S – move the dolpin backward. D – yaw the dolphin to the right. A – yaw the dolphin to the left. Left Arrow Key – move the orbit of the camera to the left in a circular fashion. Right Arrow Key – move the orbit of the camera to the right in a circular fashion. Up Arrow Key – Zoom In the camera toward the target node. Down Arrow Key – Zoom Away from the target node.

If toggle = true

Right Arrow Key – move the orbiting camera to the right in circular fashion as well as the player avatar.

Right Arrow Key – move the orbiting camera to the left in circular fashion as well as the player avatar.

Description of the node controller:

My node controller – the node controller that I created moves an object in a positive vertical manner. It provided an illusion in which it seems that the object floats away.

Rotation controller that control the “star nodes”. The children orbits the parent node and rotate in by the y axis.

Waypoint controller controls the shuttle object. It provided the illusion of a shuttle flying away and disappearing in the distance.

Description of your use of group/child node relationships

I only use one relationship of group/child that would be with the “sun node” and its children “stars node” that orbit above the game plane. The rotation speed changes every time a coin float away. Sometimes those speed changes are minor.

Description of Camera

The camera I use was an orbit camera. It allows you to move in circular fashion around the player avatar. Also allows you to zoom in and out from the player avatar. I did create a constraint in which the camera cannot zoom past the player’s avatar.

Special

Added a third node controller to control the shuttle for a visual effect

Requirements were not able to get working: I’m not sure if this counts as not meeting the requirement. Instead of using two triangles object to create a ground plane. Instead I created a square object and use that as the ground plane.

Assets:

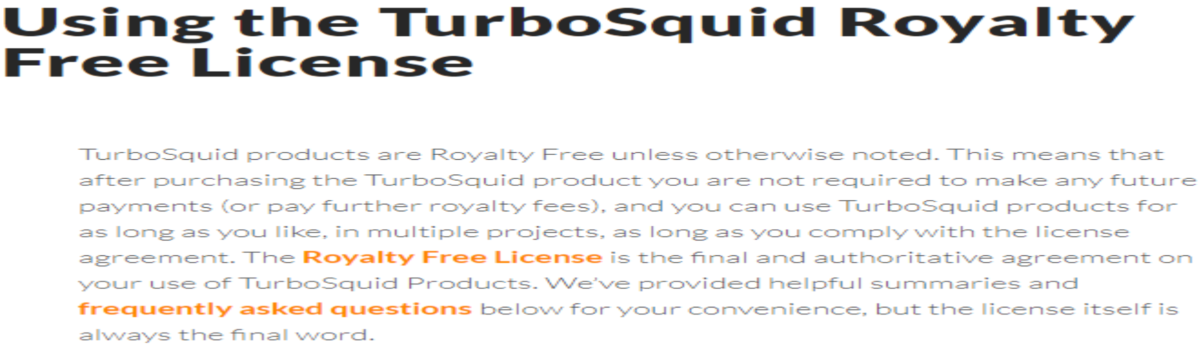
Objects: dolphinHighPoly.obj from the java gaming assets folder.

Objects: sphere.obj from the java gaming assets folder.

Objects: earth.obj from the java gaming assets folder.

Objects: shuttle.obj – Nasa 3d resources, http://nasa3d.arc.nasa.gov/. Nasa website states these are available for public non-commercial and commercial use.

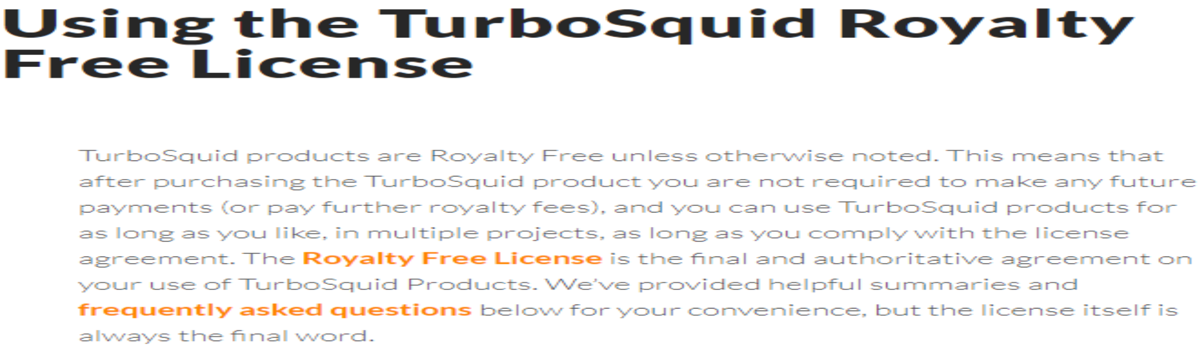
Objects: dogecoin.obj – Royalty Free License https://www.turbosquid.com/3d-models/free-max-mode-doge-coin/787687



Using the TurboSquid Royalty Free License

TurboSquid products are Royalty Free unless otherwise noted. This means that after purchasing the TurboSquid product you are not required to make any future payments (or pay further royalty fees), and you can use TurboSquid products for as long as you like, in multiple projects, as long as you comply with the license agreement. The **Royalty Free License** is the final and authoritative agreement on your use of TurboSquid Products. We’ve provided helpful summaries and **frequently asked questions** below for your convenience, but the license itself is always the final word.

Texture: 300coin.png Royalty Free License https://www.turbosquid.com/3d-models/free-max-mode-doge-coin/787687



Using the TurboSquid Royalty Free License

TurboSquid products are Royalty Free unless otherwise noted. This means that after purchasing the TurboSquid product you are not required to make any future payments (or pay further royalty fees), and you can use TurboSquid products for as long as you like, in multiple projects, as long as you comply with the license agreement. The **Royalty Free License** is the final and authoritative agreement on your use of TurboSquid Products. We've provided helpful summaries and **frequently asked questions** below for your convenience, but the license itself is always the final word.

Texture: Spstob_1.jpg - Nasa 3d resources, http://nasa3d.arc.nasa.gov/. Nasa website states these are available for public non-commercial and commercial use.

Manual Made Object: X, Y, Z axis are made by me, Square Plane is made by me.

Textures: X_Axis.png, Y_Axis.png, Z_Axis.png is made by me.

Texture: stars.png created by me.

Material: moon.mtl, sphere.mtl, dolphin.mtl from the assets folder.