

TAGE

Simple example game

"Hello Dolphin"

```
package myGame;

import tage.*;
import tage.shapes.*;

import java.lang.Math;
import java.awt.*;
import java.awt.event.*;
import java.io.*;
import javax.swing.*;
import org.joml.*;

public class MyGame extends VariableFrameRateGame
{ private static Engine engine;
  public static Engine getEngine() { return engine; }

  private Boolean paused=false;
  private int counter=0;
  private double lastFrameTime, currFrameTime, elapsTime;

  private GameObject dol;
  private ObjShape dolS;
  private TextureImage doltx;
  private Light light1;

  public MyGame() { super(); }

  public static void main(String[] args)
  { MyGame game = new MyGame();
    engine = new Engine(game);
    game.initializeSystem();
    game.game_loop();
  }
```

```
@Override
public void loadShapes()
{ dolS = new ImportedModel("dolphinHighPoly.obj");
}

@Override
public void loadTextures()
{ doltx = new TextureImage("Dolphin_HighPolyUV.png");
}

@Override
public void buildObjects()
{ Matrix4f initialTranslation, initialScale;

  // build dolphin in the center of the window
  dol = new GameObject(GameObject.root(), dolS, doltx);
  initialTranslation = (new Matrix4f()).translation(0,0,0);
  initialScale = (new Matrix4f()).scaling(3.0f);
  dol.setLocalTranslation(initialTranslation);
  dol.setLocalScale(initialScale);
}
```

```
@Override
public void initializeLights()
{ Light.setGlobalAmbient(0.5f, 0.5f, 0.5f);
  light1 = new Light();
  light1.setLocation(new Vector3f(5.0f, 4.0f, 2.0f));
  (engine.getSceneGraph()).addLight(light1);
}
```

```
@Override
public void initializeGame()
{ lastFrameTime = System.currentTimeMillis();
  currFrameTime = System.currentTimeMillis();
  elapsTime = 0.0;

  (engine.getRenderSystem()).setWindowDimensions(1900,1000);

  // ----- positioning the camera -----
  (engine.getRenderSystem().getViewport("MAIN")
    .getCamera()).setLocation(new Vector3f(0,0,5));
}
```

```
@Override
public void update()
{ // rotate dolphin if not paused
  lastFrameTime = currFrameTime;
  currFrameTime = System.currentTimeMillis();
  if (!paused) elapsTime += (currFrameTime - lastFrameTime) / 1000.0;
  dol.setLocalRotation((
    new Matrix4f()).rotation((float)elapsTime, 0, 1, 0));

  // build and set HUD
  int elapsTimeSec = Math.round((float)(elapsTime));
  String elapsTimeStr = Integer.toString(elapsTimeSec);
  String counterStr = Integer.toString(counter);
  String dispStr1 = "Time = " + elapsTimeStr;
  String dispStr2 = "Keyboard hits = " + counterStr;
  Vector3f hud1Color = new Vector3f(1,0,0);
  Vector3f hud2Color = new Vector3f(0,0,1);
  (engine.getHUDmanager()).setHUD1(dispStr1, hud1Color, 15, 15);
  (engine.getHUDmanager()).setHUD2(dispStr2, hud2Color, 500, 15);
}
```

```
@Override
public void keyPressed(KeyEvent e)
{ switch (e.getKeyCode())
  { case KeyEvent.VK_C:
    counter++;
    break;
    case KeyEvent.VK_1:
    paused = !paused;
    break;
    case KeyEvent.VK_2:
    dol.getRenderStates().setWireframe(true);
    break;
    case KeyEvent.VK_3:
    dol.getRenderStates().setWireframe(false);
    break;
    case KeyEvent.VK_4:
    (engine.getRenderSystem().getViewport("MAIN")
      .getCamera()).setLocation(new Vector3f(0,0,0))
    break;
  }
  super.keyPressed(e);
}
```