Loading OBJ models *(TAGE)*

```
. . .
public class myGame extends VariableFrameRateGame
{  . . .
   private GameObject robot, terr, water;
   private ObjShape robS, terrS, waterS;
   private TextureImage robottx, hills, grass;

   private float robotHeightAdjust = 0.5f;
   . . .

   @Override
   public void loadShapes()
   {  robS = new ImportedModel("robot2019.obj");
      terrS = new TerrainPlane(1000);
      waterS = new Plane();
   }  }

   @Override
   public void loadTextures()
   {  robottx = new TextureImage("robot.jpg");
      hills = new TextureImage("hills.jpg");
      grass = new TextureImage("grass.jpg");
   }

   @Override
   public void buildObjects()
   {  // build robot avatar
      robot = new GameObject(GameObject.root(), robS, robottx);
      initialTranslation =
         (new Matrix4f()).translation(0,robotHeightAdjust,0);
      robot.setLocalTranslation(initialTranslation);
      initialRotation =
       (new Matrix4f()).rotationY((float)java.lang.Math.toRadians(180.0f));
      initialScale = (new Matrix4f()).scaling(0.2f, 0.2f, 0.2f);
      robot.setLocalScale(initialScale);
      robot.setLocalRotation(initialRotation);
      robot.getRenderStates().setModelOrientationCorrection(
        (new Matrix4f()).rotationY((float)java.lang.Math.toRadians(90.0f)));
//    robot.getRenderStates().hasLighting(true);
//    robot.getRenderStates().isEnvironmentMapped(true);
      . . .
   }

   @Override
   public void update()
   {  . . .
      // update altitude of robot based on height map
      Vector3f loc = robot.getWorldLocation();
      float height = terr.getHeight(loc.x(), loc.z());
      robot.setLocalLocation(
         new Vector3f(loc.x(), height + robotHeightAdjust, loc.z()));
      . . .
   }
```

(add these lines to make the dolphin chrome)

Loading Animated RKM models *(TAGE)*

```
public class myGame extends VariableFrameRateGame
{  . . .
   private GameObject robot, terr, water;
   private AnimatedShape robS;
   private ObjShape terrS, waterS;
   private TextureImage robottx, hills, grass;
   . . .

   @Override
   public void loadShapes()
   {  robS = new AnimatedShape("robot.rkm", "robot.rks");
      robS.loadAnimation("WAVE", "robotWave.rka");
      robS.loadAnimation("WALK", "robotWalk.rka");
      terrS = new TerrainPlane(1000);
      waterS = new Plane();
   }  }

   @Override
   public void buildObjects()
   {  // build robot avatar (NO CHANGE)
      robot = new GameObject(GameObject.root(), robS, robottx);
      . . .
   }

   @Override
   public void update()
   {  . . .
      robS.updateAnimation();
      . . .
   }

   @Override
   public void keyPressed(KeyEvent e)
   {  switch (e.getKeyCode())
      {  case KeyEvent.VK_W:
         {  robS.stopAnimation();
            robS.playAnimation("WALK", 0.5f,
                                    AnimatedShape.EndType.LOOP, 0);
            break;
         }
         case KeyEvent.VK_V:
         {  robS.stopAnimation();
            robS.playAnimation("WAVE", 0.5f,
                                    AnimatedShape.EndType.LOOP, 0);
            break;
         }
         case KeyEvent.VK_S:
         {  robS.stopAnimation();
            break;
         }
      }
      super.keyPressed(e);
   }
```