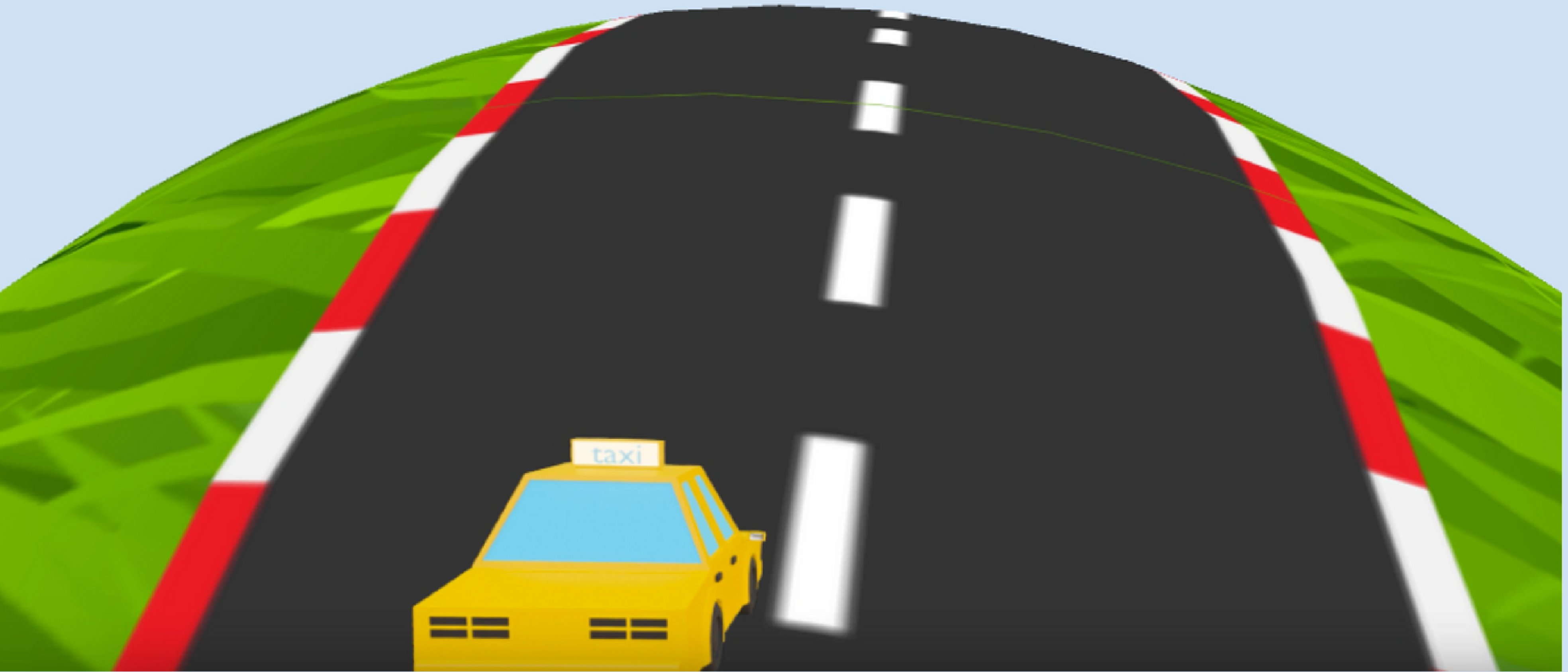


Racing Game

playing by User Gesture

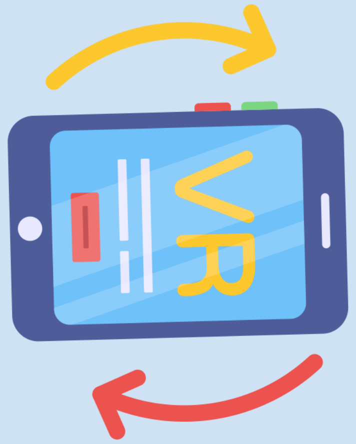
20221124 Bohyun Choi

How to play

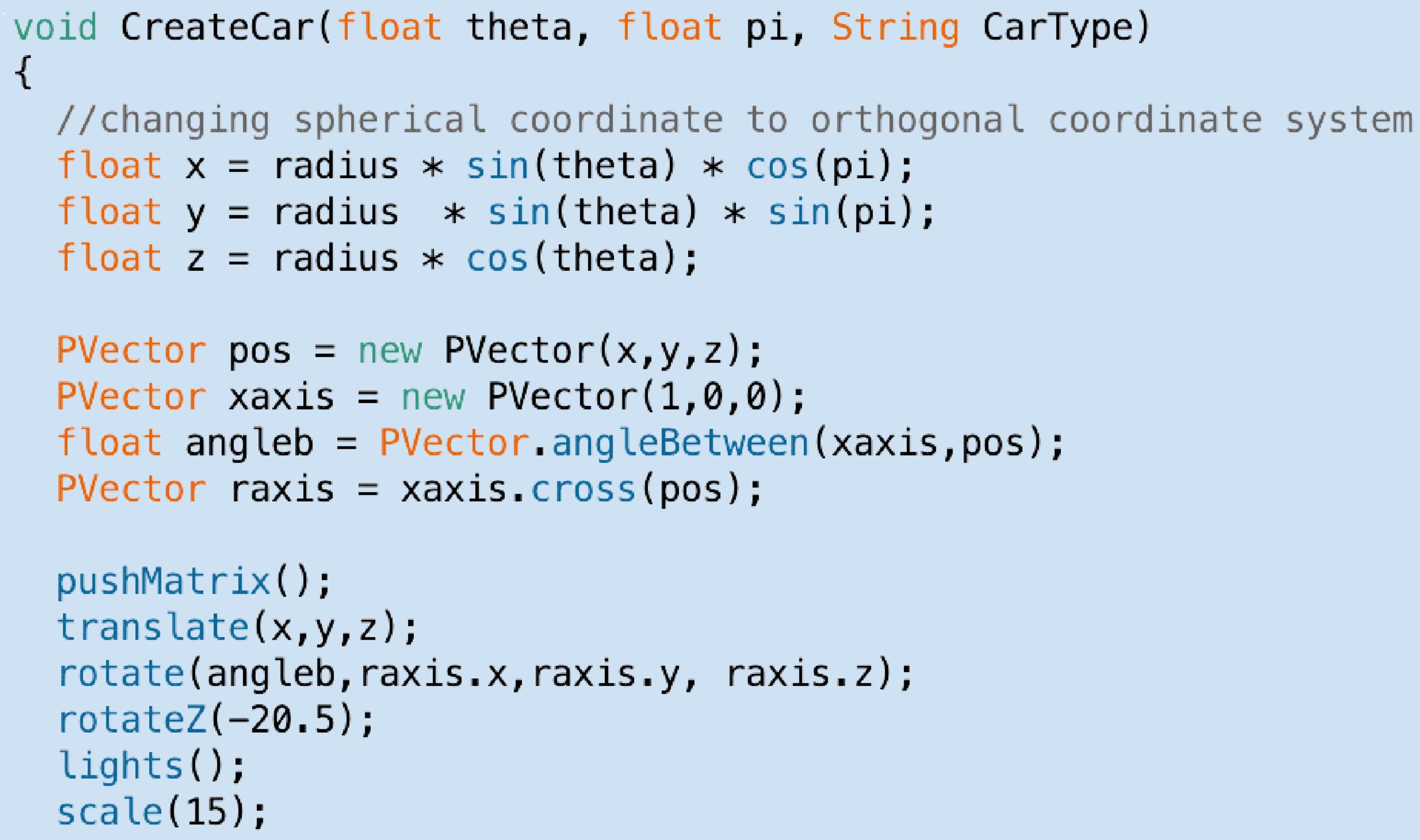
You can adjust the position of your car to the tilt of the phone on both sides.

If you tilt your phone to the left, the car will go to the left,
and if you tilt it to the right, it will go to the right.

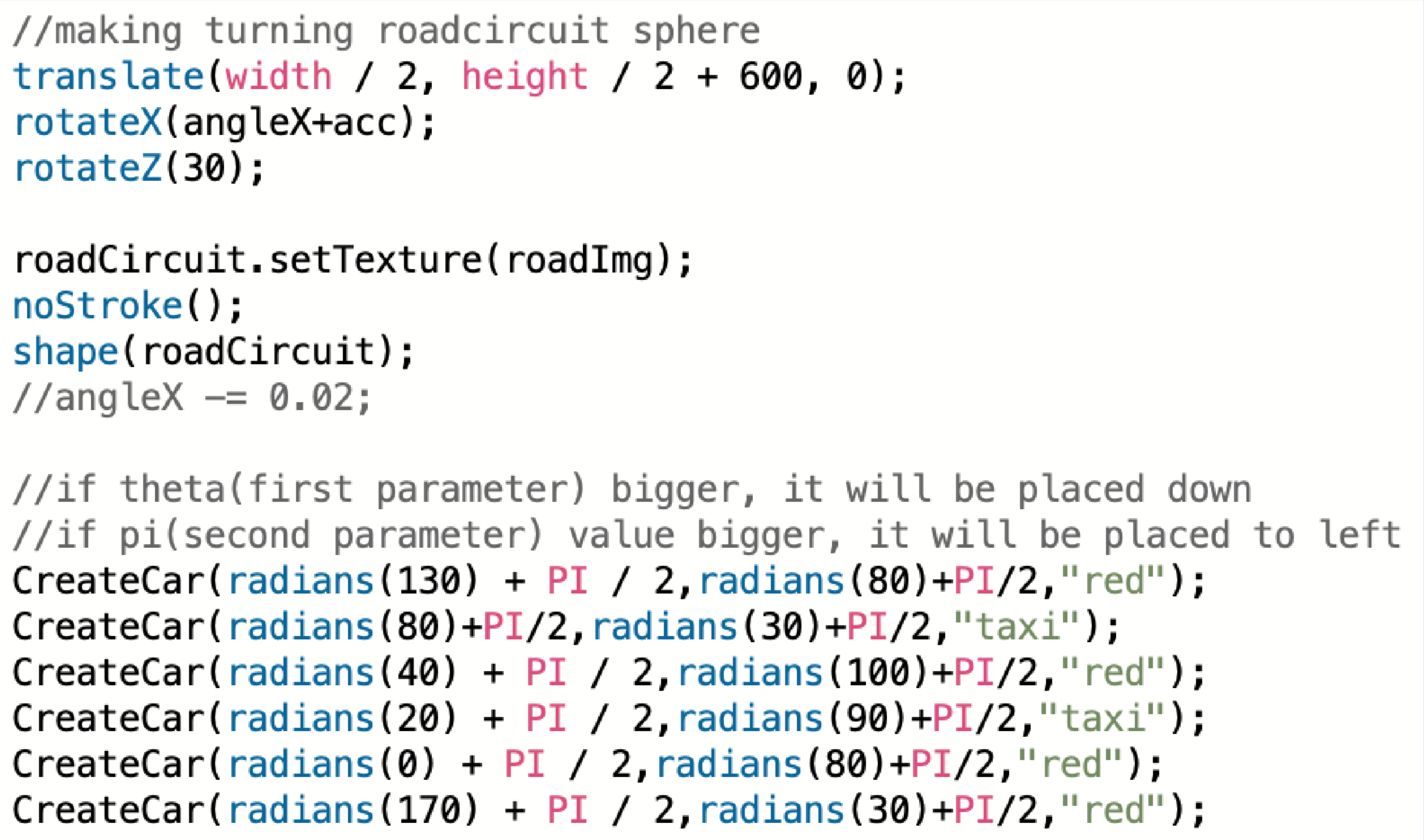
The car slows down when the car goes into the grass.



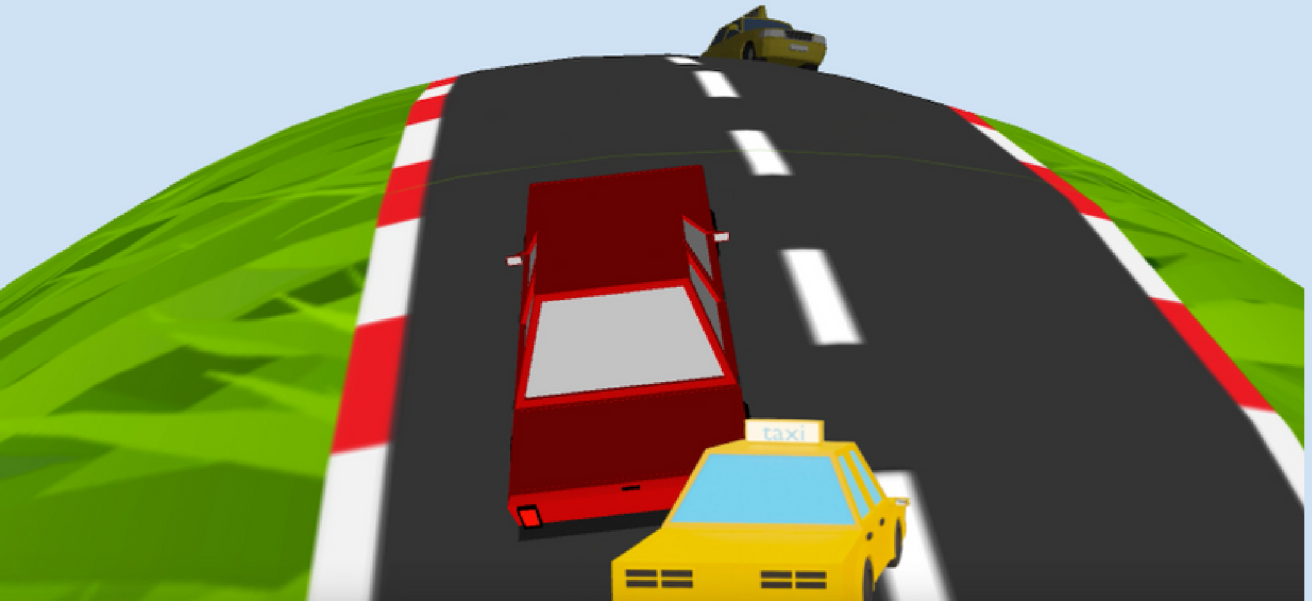
Code



```
void CreateCar(float theta, float pi, String CarType)
{
  //changing spherical coordinate to orthogonal coordinate system
  float x = radius * sin(theta) * cos(pi);
  float y = radius  * sin(theta) * sin(pi);
  float z = radius * cos(theta);

  PVector pos = new PVector(x,y,z);
  PVector xaxis = new PVector(1,0,0);
  float angleb = PVector.angleBetween(xaxis,pos);
  PVector raxis = xaxis.cross(pos);

  pushMatrix();
  translate(x,y,z);
  rotate(angleb,raxis.x,raxis.y, raxis.z);
  rotateZ(-20.5);
  lights();
  scale(15);
```

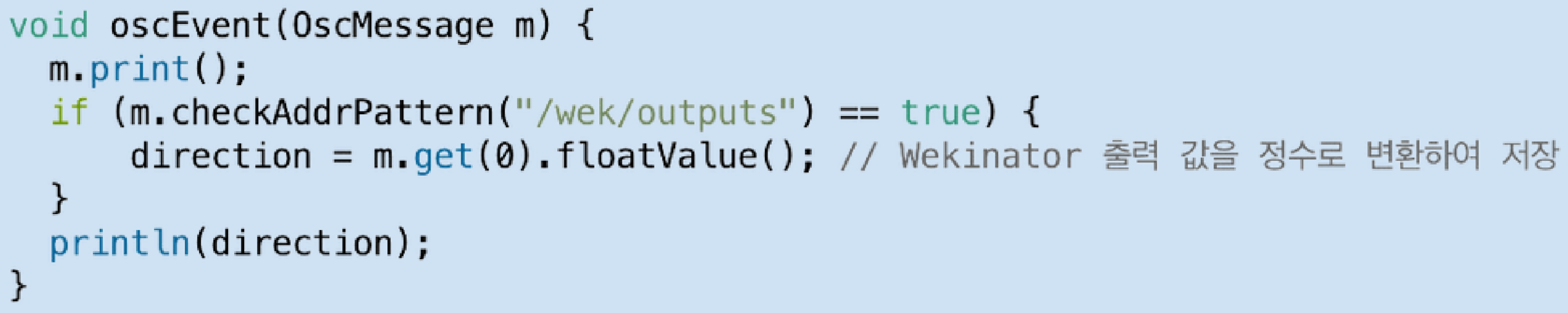


```
//making turning roadcircuit sphere
translate(width / 2, height / 2 + 600, 0);
rotateX(angleX+acc);
rotateZ(30);

roadCircuit.setTexture(roadImg);
noStroke();
shape(roadCircuit);
//angleX -= 0.02;

//if theta(first parameter) bigger, it will be placed down
//if pi(second parameter) value bigger, it will be placed to left
CreateCar(radians(130) + PI / 2,radians(80)+PI/2,"red");
CreateCar(radians(80)+PI/2,radians(30)+PI/2,"taxi");
CreateCar(radians(40) + PI / 2,radians(100)+PI/2,"red");
CreateCar(radians(20) + PI / 2,radians(90)+PI/2,"taxi");
CreateCar(radians(0) + PI / 2,radians(80)+PI/2,"red");
CreateCar(radians(170) + PI / 2,radians(30)+PI/2,"red");
```

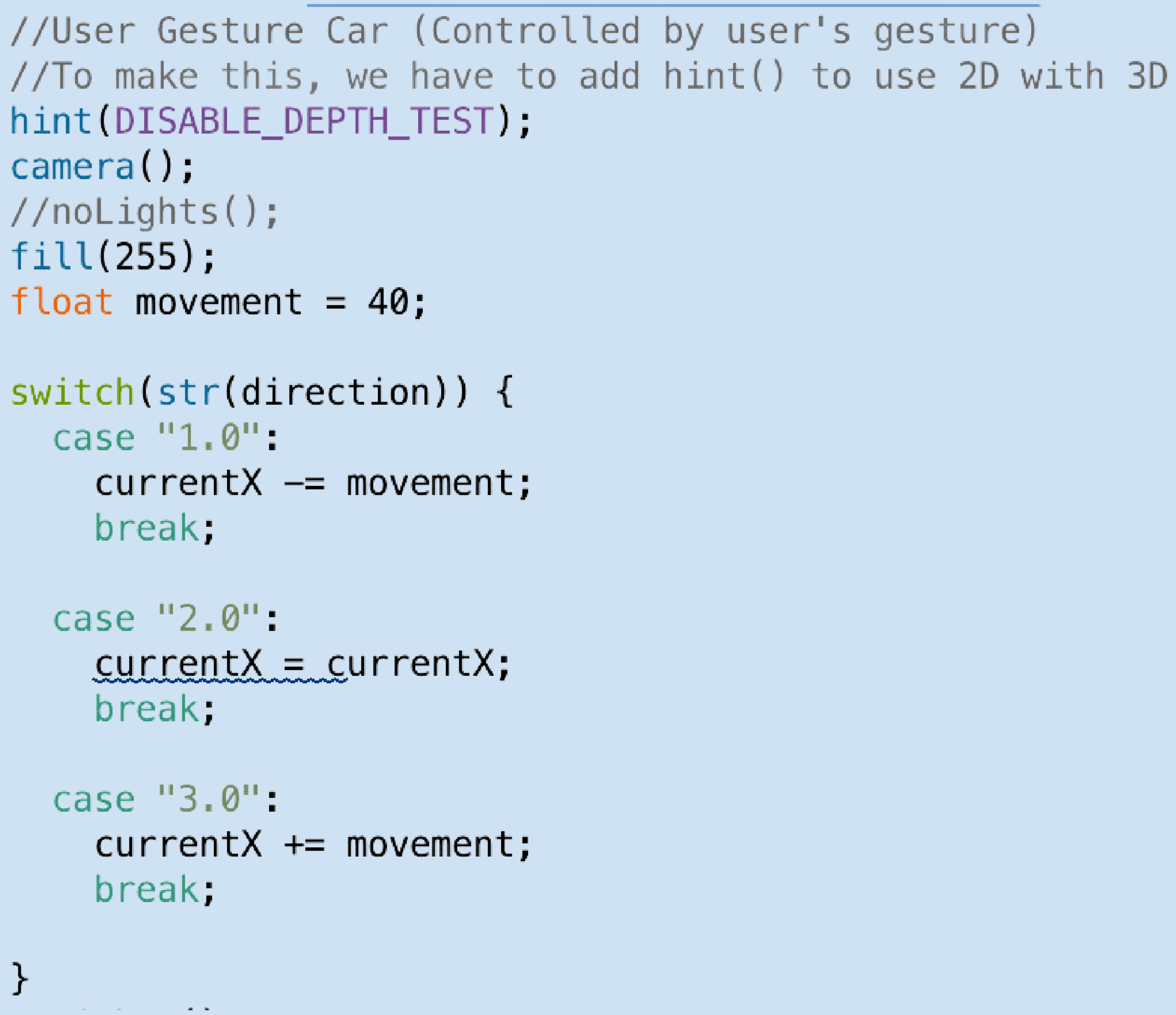


A spherical coordinate system was used to place 3d models on each spherical surface.
The 3D model was downloaded in obj format and placed in processing using PShape.

Code



```
void oscEvent(OscMessage m) {
  m.print();
  if (m.checkAddrPattern("/wek/outputs") == true) {
      direction = m.get(0).floatValue(); // Wekinator 출력 값을 정수로 변환하여 저장
  }
  println(direction);
}
```



```
//User Gesture Car (Controlled by user's gesture)
//To make this, we have to add hint() to use 2D with 3D
hint(DISABLE_DEPTH_TEST);
camera();
//noLights();
fill(255);
float movement = 40;

switch(str(direction)) {
  case "1.0":
    currentX -= movement;
    break;

  case "2.0":
    currentX = currentX;
    break;

  case "3.0":
    currentX += movement;
    break;

}
```

After learning the wekinator with class (type: 1,2), direction was judged to be left if it was 1, and right if it was 2.

taxi
taxi
TAXI