Computer Graphics Term Project

장재혁, 이민서, 조병근, 심우석

TABLE OF CONTENTS

01

Contents

Red light, Green light
(무궁화 꽃이 피었습니다)

02

Key Features

Our game's feature

03

tech. details

Implementation details for technologies

04

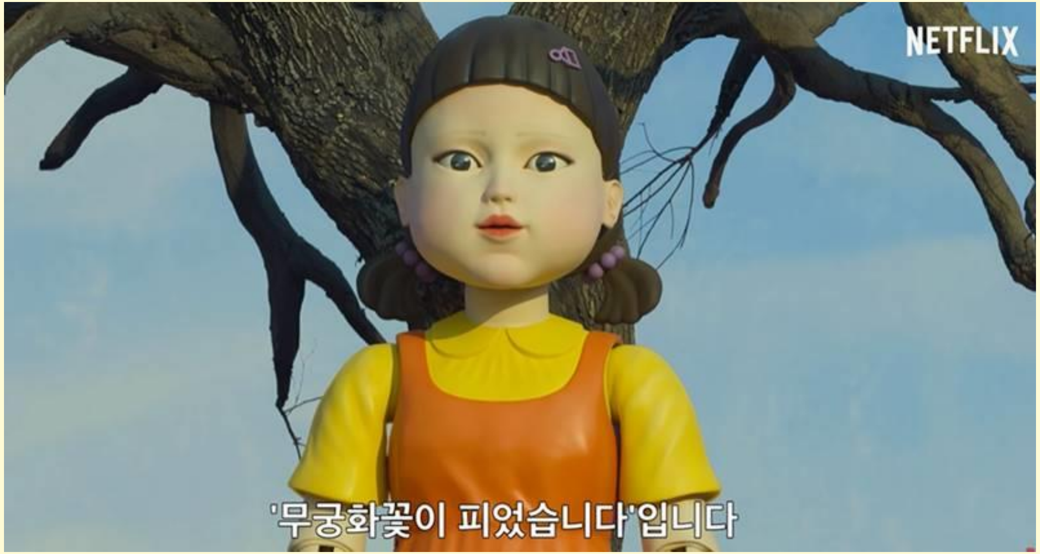
Member Role

Each Member's Role

01 - Contents

Title & Brief Description

Red Light, Green Light



Squid Game!!

Description

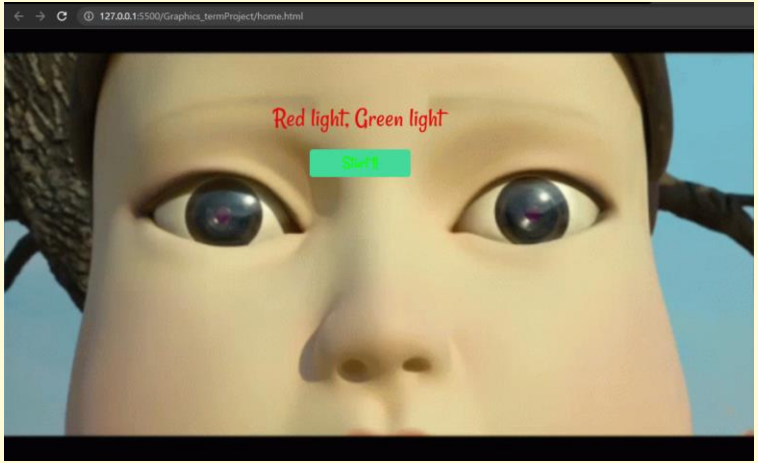
Name: Red light, Green light (무궁화 꽃이 피었습니다)

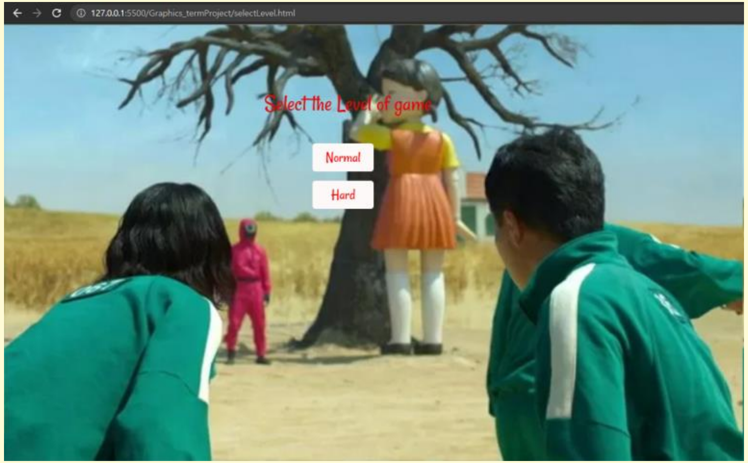
Game summary:

Before the tagger look back, the player can move to front or anywhere. But the player can't move when the tagger looks the player. The player wins when the player reaches to the tagger. The player also have to avoid obstacle from the tagger.

Contents

Contents





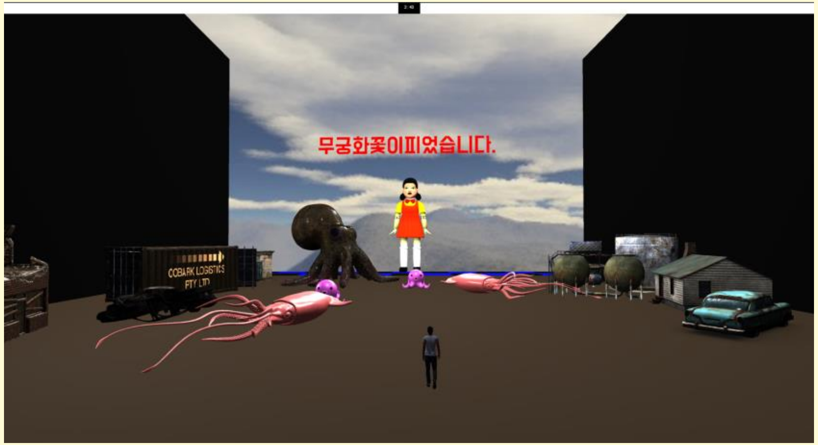
Home Page

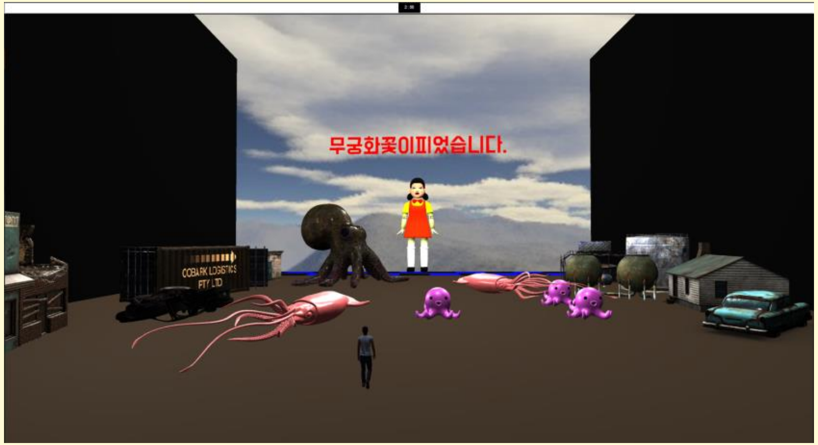
Press the start button to move on to the next page.

Select the level of game

You can choose the level of difficulty of the game.
If you choose the difficulty level, you'll move on to the next page.

Contents





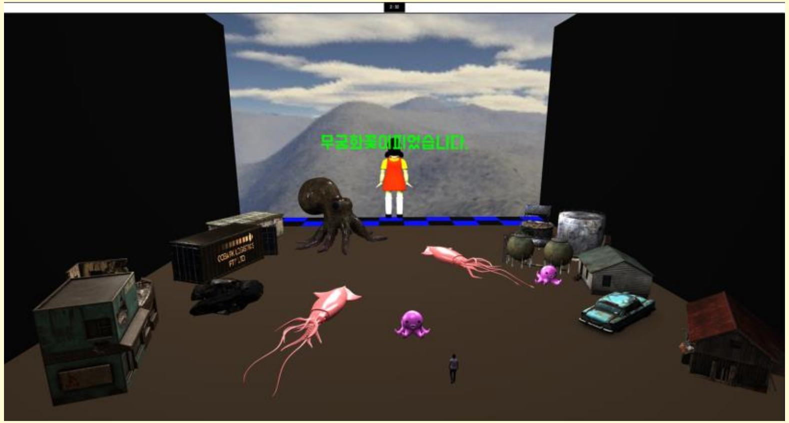
Normal Mode

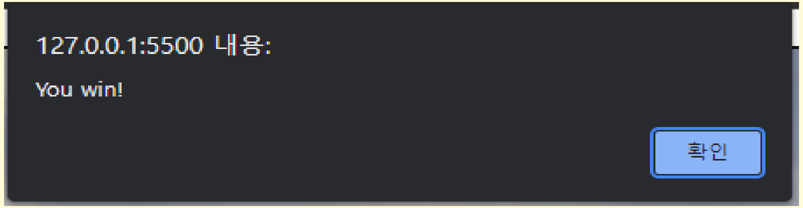
Freely moving obstacles(Small squid)
Big stable obstacles(Big Squid)
Spinning tagger(Girl)

Hard Mode

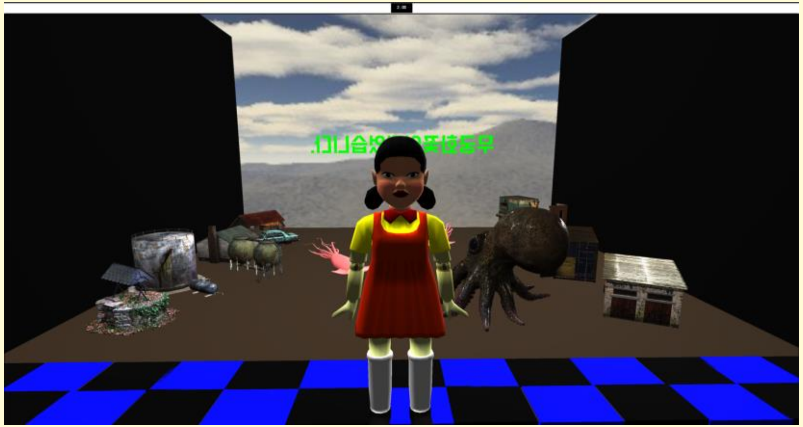
More obstacles
faster obstacles
faster tagger rotation
than normal

Contents



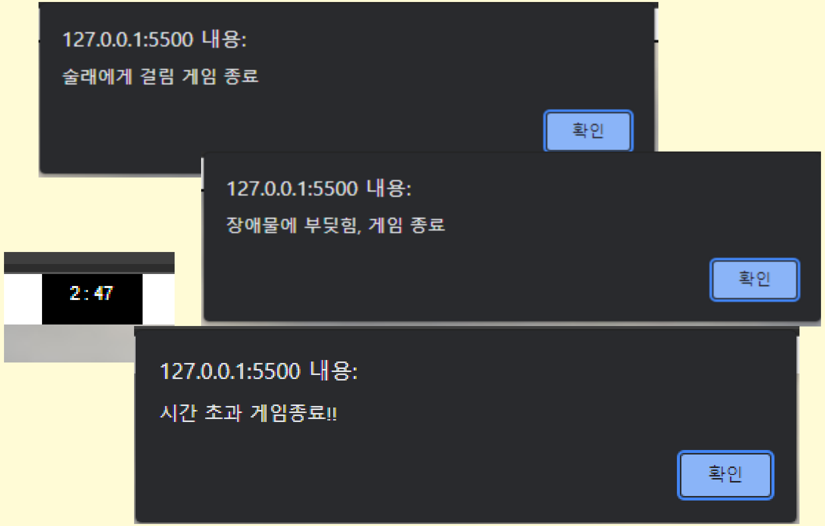

127.0.0.1:5500 내용:

You win!



You win when you reach the blue finish line

Contents



127.0.0.1:5500 내용:

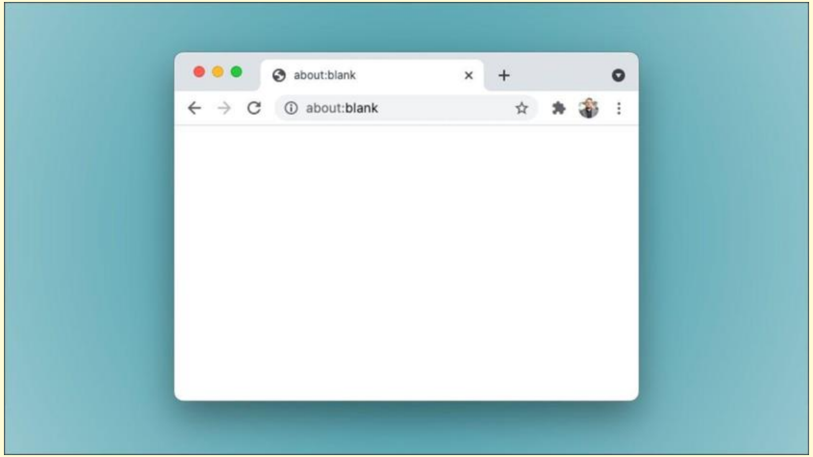
술래에게 걸림 게임 종료

127.0.0.1:5500 내용:

장애물에 부딪힘, 게임 종료

127.0.0.1:5500 내용:

시간 초과 게임종료!!



Game over

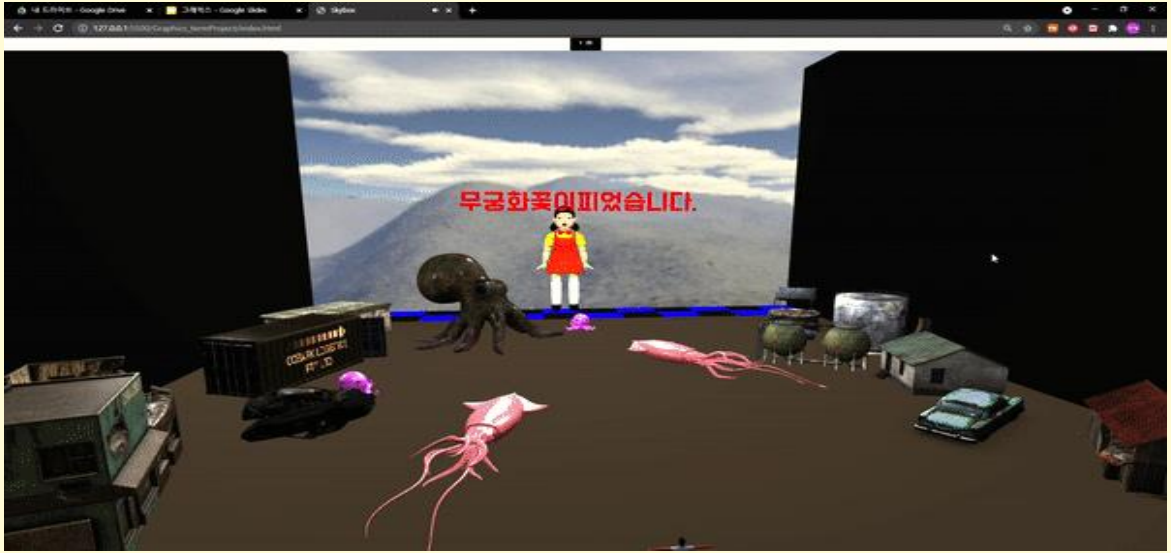
Move to blank page if game over

02 - Key features

Advantages

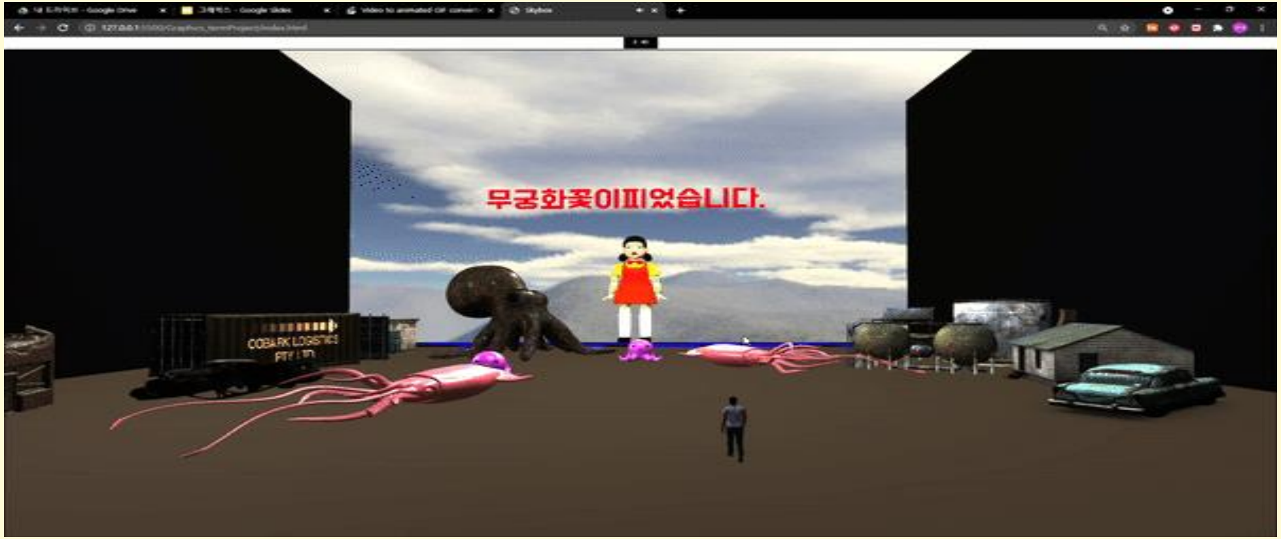
Red light, Green light

Key features

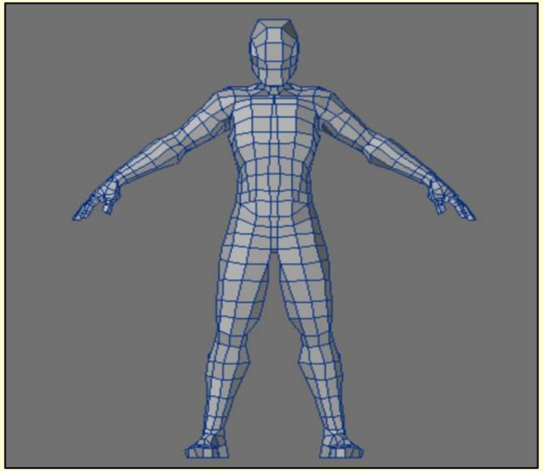


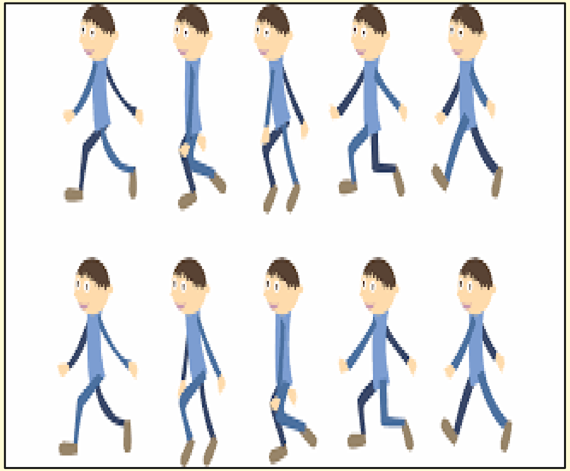
Red light, Green light

Key features



Key features





3D model Use

We use various model including GLTF model, FBX model and use it for background and game character

Animation

We use Animation in moving character, obstacles movement, tagger rotation

Key features

Event Handler

Movement through keyboard
Collision with obstacles
Movement when Tagger look back

Rotate Camera Position

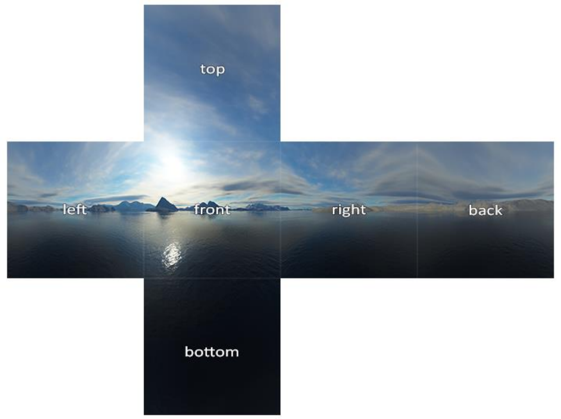
Change the viewpoint of
camera with the mouse

03 - Tech. details

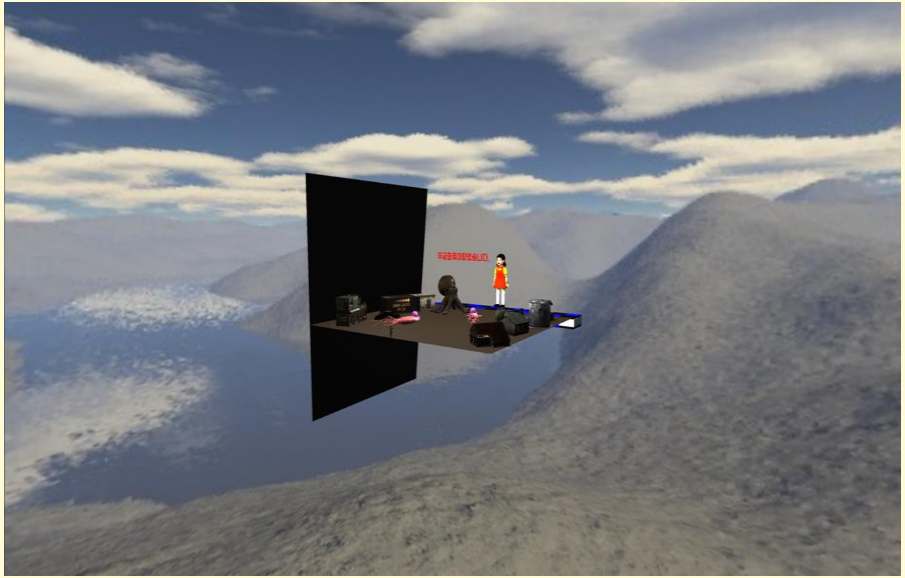
Implementation details
for technologies

Tech. details

Skybox – Cube



Skybox (Cube)

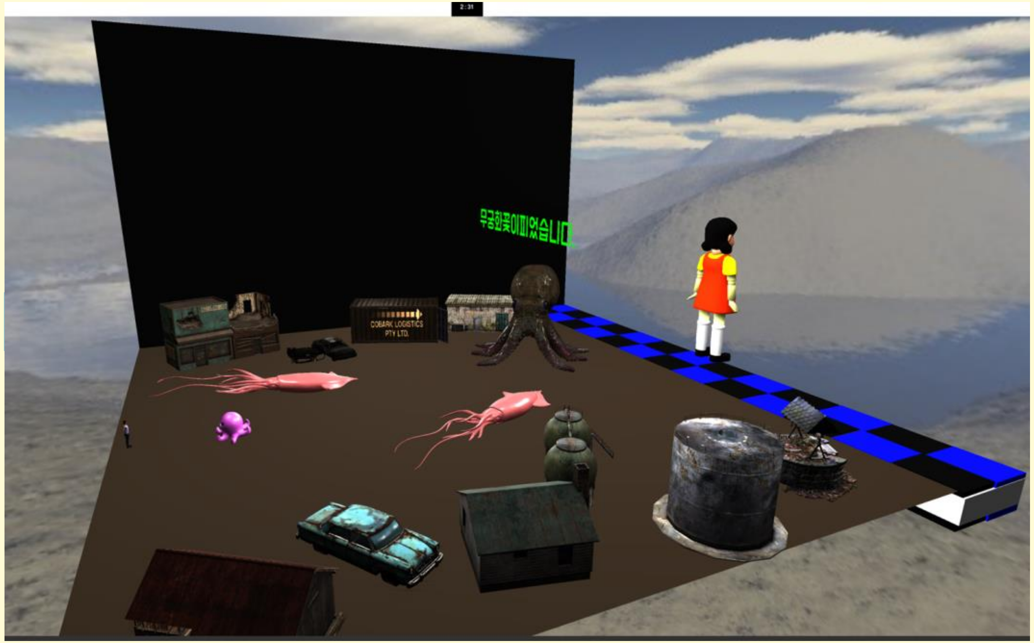


https://opengameart.org/node/28781

Tech. details

Model – Background

GLTF Model



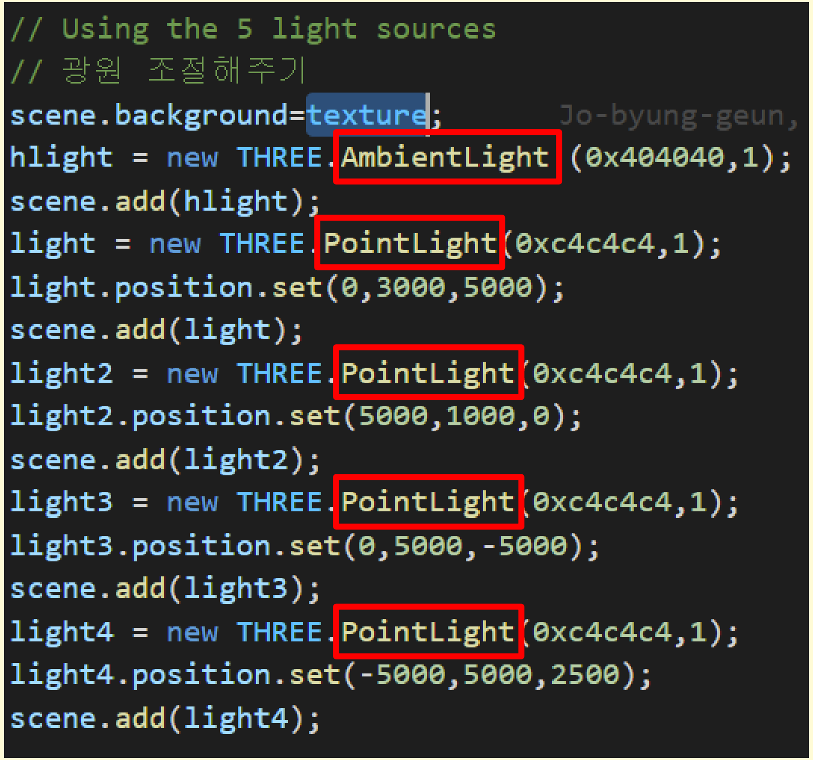
https://sketchfab.com/feed

Tech. details

Light source

Light

Ambient +
Point

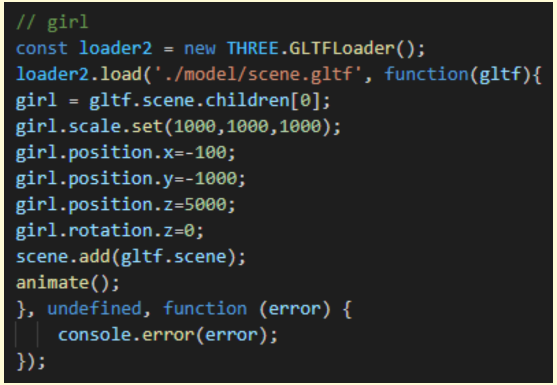


```
// Using the 5 light sources
// 광원 조절해주기
scene.background=texture;        Jo-byung-geun,
hlight = new THREE.AmbientLight (0x404040,1);
scene.add(hlight);
light = new THREE.PointLight(0xc4c4c4,1);
light.position.set(0,3000,5000);
scene.add(light);
light2 = new THREE.PointLight(0xc4c4c4,1);
light2.position.set(5000,1000,0);
scene.add(light2);
light3 = new THREE.PointLight(0xc4c4c4,1);
light3.position.set(0,5000,-5000);
scene.add(light3);
light4 = new THREE.PointLight(0xc4c4c4,1);
light4.position.set(-5000,5000,2500);
scene.add(light4);
```

Tech. details

Model

GLTF Model



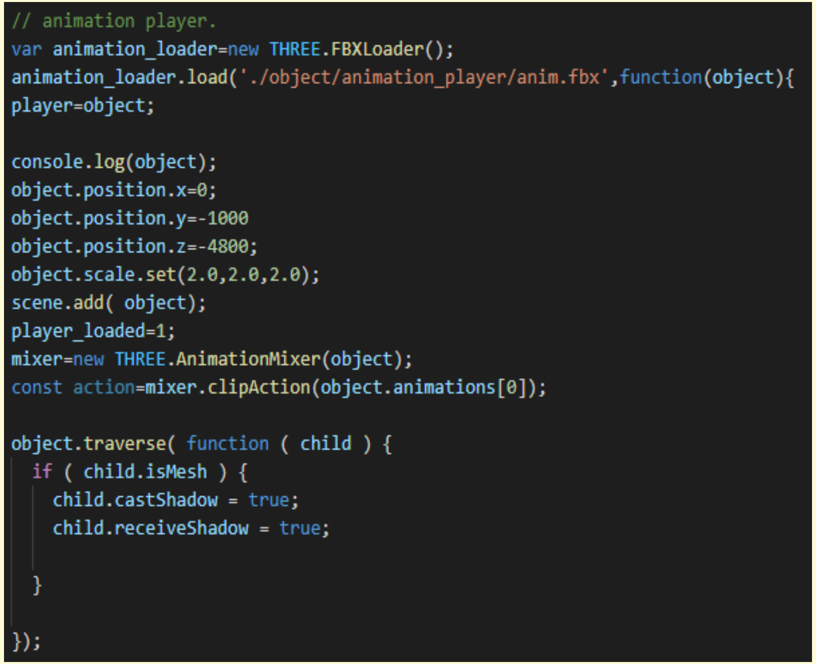
```
// girl
const loader2 = new THREE.GLTFLoader();
loader2.load('./model/scene.gltf', function(gltf){
girl = gltf.scene.children[0];
girl.scale.set(1000,1000,1000);
girl.position.x=-100;
girl.position.y=-1000;
girl.position.z=5000;
girl.rotation.z=0;
scene.add(gltf.scene);
animate();
}, undefined, function (error) {
    console.error(error);
});
```

https://sketchfab.com/feed

Tech. details

Model

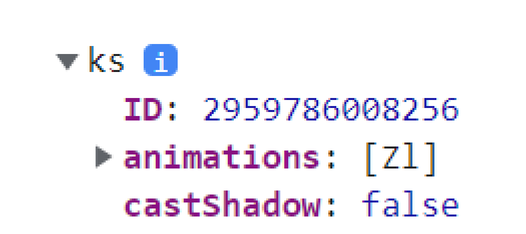
Animation Model



```
// animation player.
var animation_loader=new THREE.FBXLoader();
animation_loader.load('./object/animation_player/anim.fbx',function(object){
player=object;

console.log(object);
object.position.x=0;
object.position.y=-1000
object.position.z=-4800;
object.scale.set(2.0,2.0,2.0);
scene.add( object);
player_loaded=1;
mixer=new THREE.AnimationMixer(object);
const action=mixer.clipAction(object.animations[0]);

object.traverse( function ( child ) {
  if ( child.isMesh ) {
    child.castShadow = true;
    child.receiveShadow = true;

  }

});
```

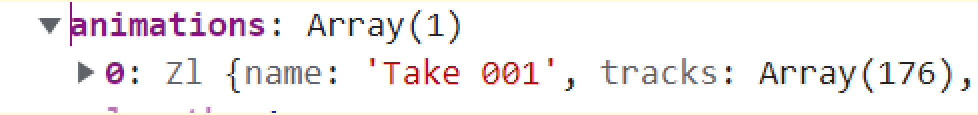
Tech. details

Animation model mixer

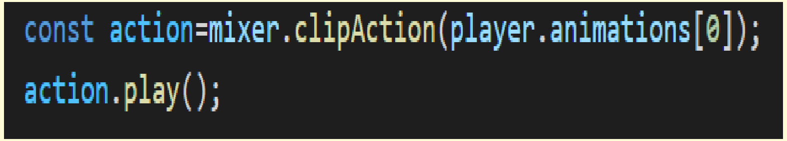


```
▼ ks i
    ID: 2959786008256
  ▶ animations: [Zl]
    castShadow: false
```

Model console

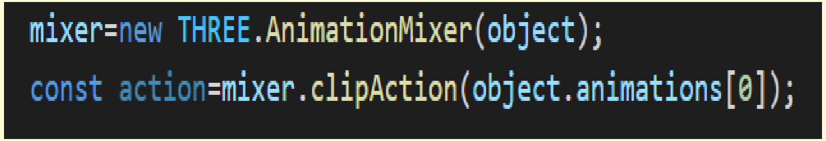


```
▼ animations: Array(1)
  ▶ 0: Zl {name: 'Take 001', tracks: Array(176),
```



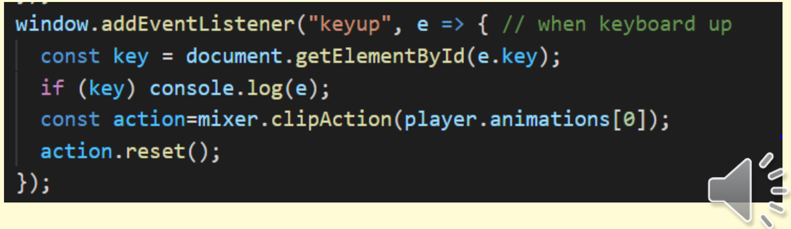
```
const action=mixer.clipAction(player.animations[0]);
action.play();
```

Action play



```
mixer=new THREE.AnimationMixer(object);
const action=mixer.clipAction(object.animations[0]);
```

Animation Mixer

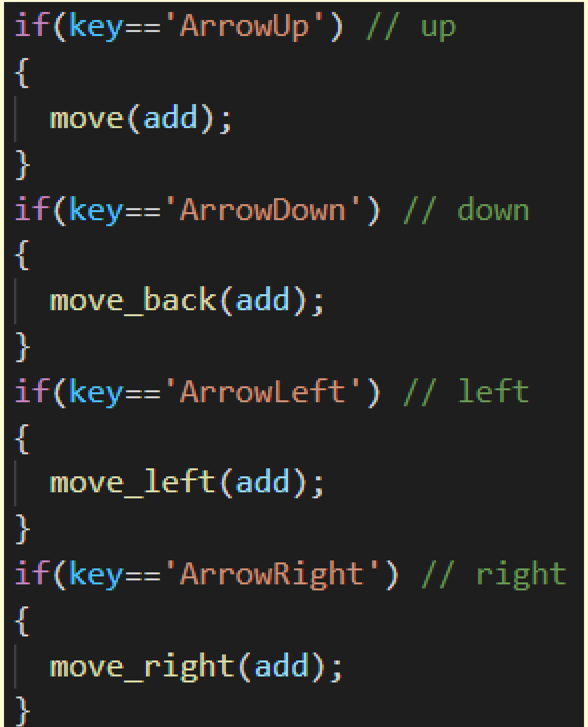


```
window.addEventListener("keyup", e => { // when keyboard up
  const key = document.getElementById(e.key);
  if (key) console.log(e);
  const action=mixer.clipAction(player.animations[0]);
  action.reset();
});
```

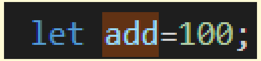
Action Rest

Tech. details

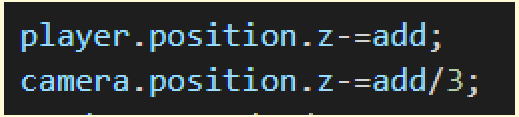
Keyboard



```
if(key=='ArrowUp') // up
{
  move(add);
}
if(key=='ArrowDown') // down
{
  move_back(add);
}
if(key=='ArrowLeft') // left
{
  move_left(add);
}
if(key=='ArrowRight') // right
{
  move_right(add);
}
```



```
 let add=100;
```

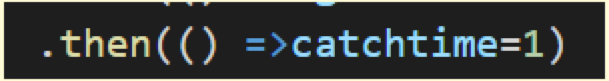


```
player.position.z-=add;
camera.position.z-=add/3;
```

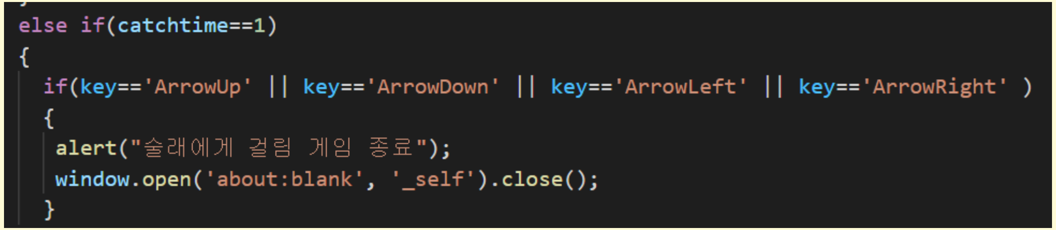
Tech. details

Keyboard

무궁화꽃이피었습니다.



```
.then(() =>catchtime=1)
```

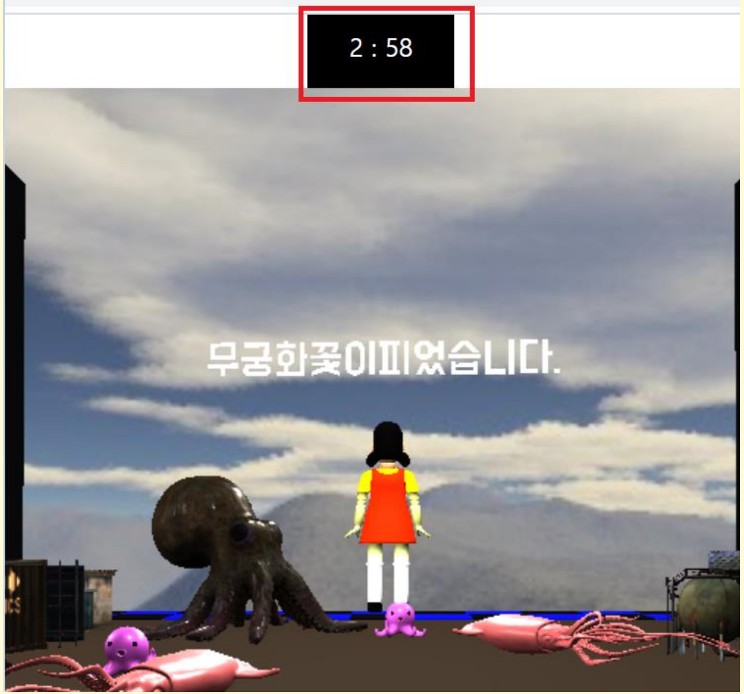


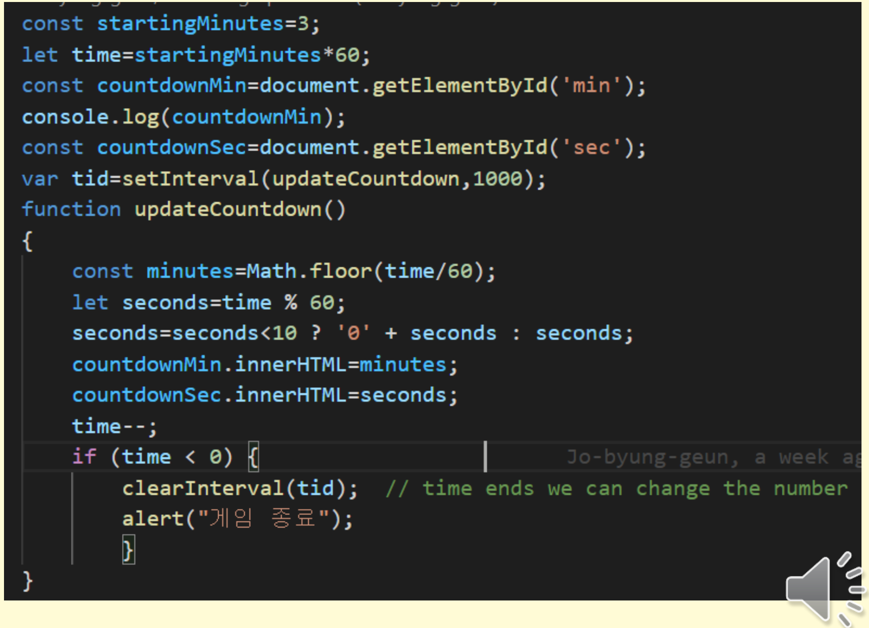
```
else if(catchtime==1)
{
  if(key=='ArrowUp' || key=='ArrowDown' || key=='ArrowLeft' || key=='ArrowRight' )
  {
   alert("술래에게 걸림 게임 종료");
   window.open('about:blank', '_self').close();
  }
```

When player moves

Tech. details

Timer





```
const startingMinutes=3;
let time=startingMinutes*60;
const countdownMin=document.getElementById('min');
console.log(countdownMin);
const countdownSec=document.getElementById('sec');
var tid=setInterval(updateCountdown,1000);
function updateCountdown()
{
    const minutes=Math.floor(time/60);
    let seconds=time % 60;
    seconds=seconds<10 ? '0' + seconds : seconds;
    countdownMin.innerHTML=minutes;
    countdownSec.innerHTML=seconds;
    time--;
    if (time < 0) {                    Jo-byung-geun, a week ag
        clearInterval(tid);  // time ends we can change the number
        alert("게임 종료");
        }
}
```

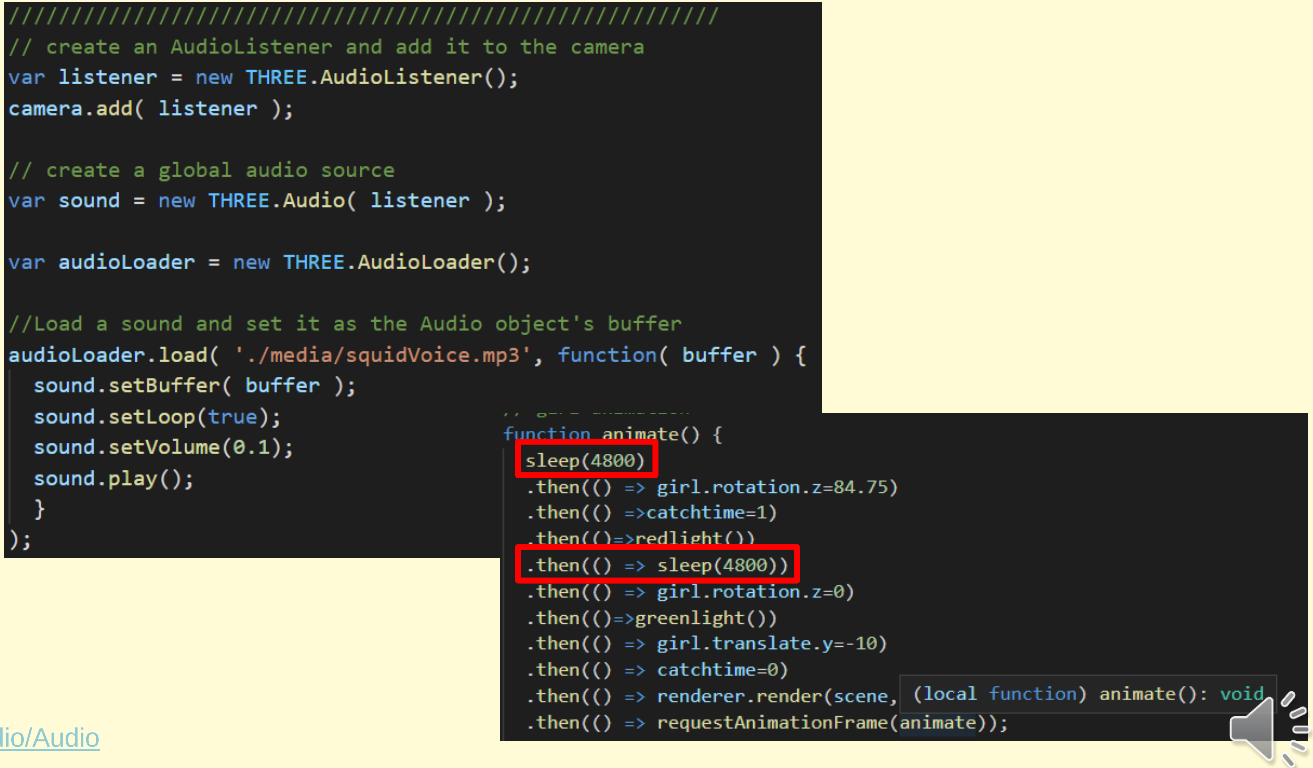
script.js

Tech. details

Audio

Normal Audio

Hard Audio



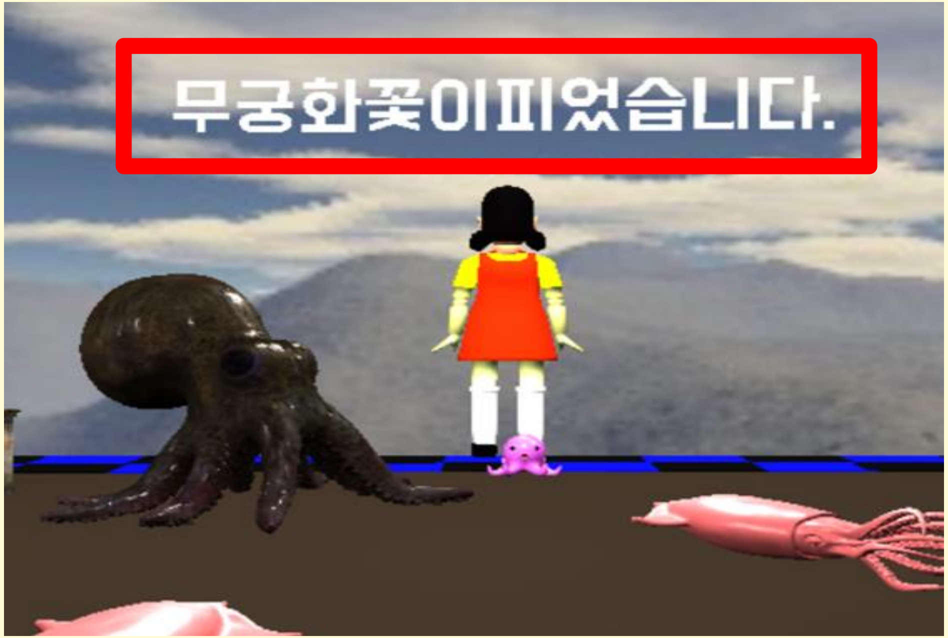
```
////////////////////////////////////////////////////////
// create an AudioListener and add it to the camera
var listener = new THREE.AudioListener();
camera.add( listener );

// create a global audio source
var sound = new THREE.Audio( listener );

var audioLoader = new THREE.AudioLoader();

//Load a sound and set it as the Audio object's buffer
audioLoader.load( './media/squidVoice.mp3', function( buffer ) {
  sound.setBuffer( buffer );
  sound.setLoop(true);
  sound.setVolume(0.1);
  sound.play();
  }
);
```

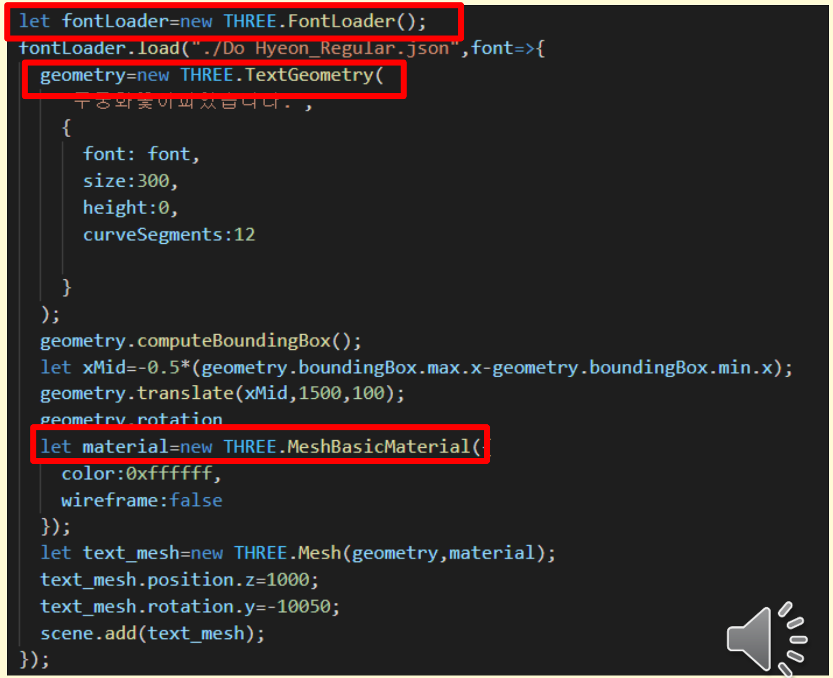
```
// girl animation
function animate() {
  sleep(4800)
  .then(() => girl.rotation.z=84.75)
  .then(() =>catchtime=1)
  .then(()=>redlight())
  .then(() => sleep(4800))
  .then(() => girl.rotation.z=0)
  .then(()=>greenlight())
  .then(() => girl.translate.y=-10)
  .then(() => catchtime=0)
  .then(() => renderer.render(scene, (local function) animate(): void
  .then(() => requestAnimationFrame(animate));
```

https://threejs.org/docs/#api/en/audio/Audio
https://audiotrimmer.com/kr/

Tech. details

3D text





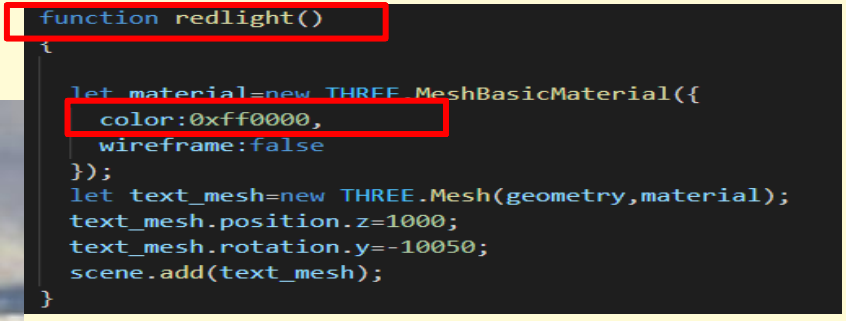
```
let fontLoader=new THREE.FontLoader();
fontLoader.load("./Do Hyeon_Regular.json",font=>{
  geometry=new THREE.TextGeometry(
    "무궁화꽃이피었습니다.",
    {
      font: font,
      size:300,
      height:0,
      curveSegments:12

    }
  );
  geometry.computeBoundingBox();
  let xMid=-0.5*(geometry.boundingBox.max.x-geometry.boundingBox.min.x);
  geometry.translate(xMid,1500,100);
  geometry.rotation
  let material=new THREE.MeshBasicMaterial({
    color:0xffffff,
    wireframe:false
  });
  let text_mesh=new THREE.Mesh(geometry,material);
  text_mesh.position.z=1000;
  text_mesh.rotation.y=-10050;
  scene.add(text_mesh);
});
```

http://www.gisdeveloper.co.kr/?p=10962

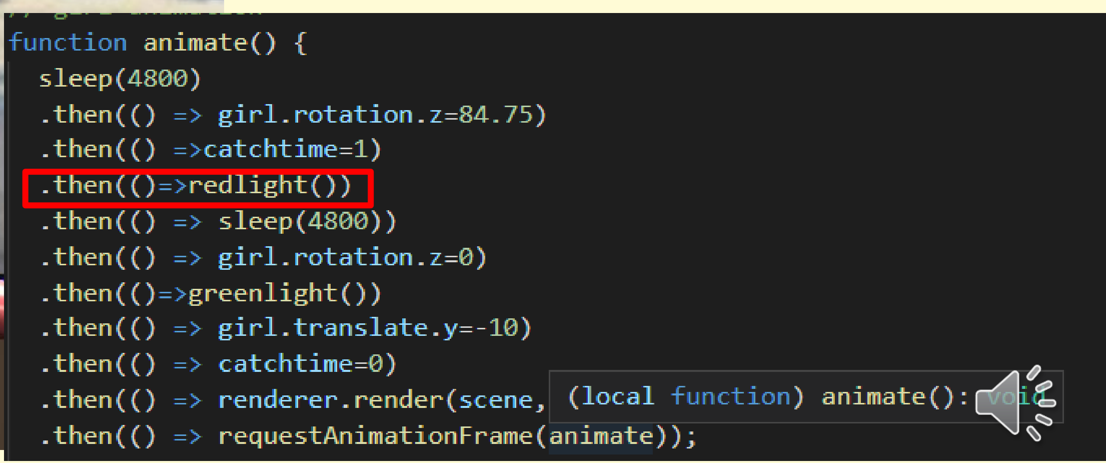
Tech. details

3D text

무궁화꽃이피었습니다.



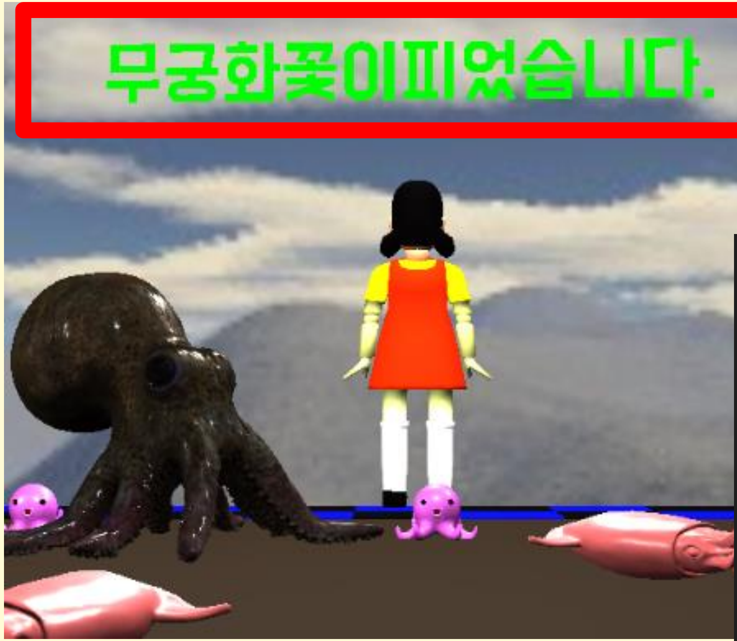
```
function redlight()
{

  let material=new THREE.MeshBasicMaterial({
    color:0xff0000,
    wireframe:false
  });
  let text_mesh=new THREE.Mesh(geometry,material);
  text_mesh.position.z=1000;
  text_mesh.rotation.y=-10050;
  scene.add(text_mesh);
}
```

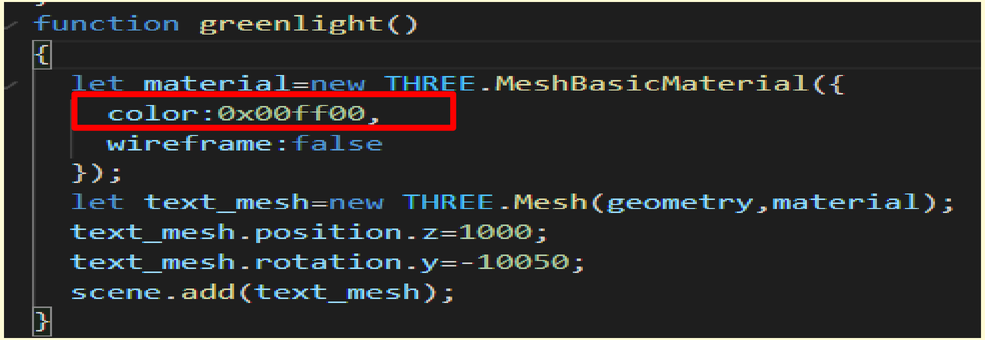


```
// girl animation
function animate() {
  sleep(4800)
  .then(() => girl.rotation.z=84.75)
  .then(() =>catchtime=1)
  .then(()=>redlight())
  .then(() => sleep(4800))
  .then(() => girl.rotation.z=0)
  .then(()=>greenlight())
  .then(() => girl.translate.y=-10)
  .then(() => catchtime=0)
  .then(() => renderer.render(scene, (local function) animate(): void
  .then(() => requestAnimationFrame(animate));
```

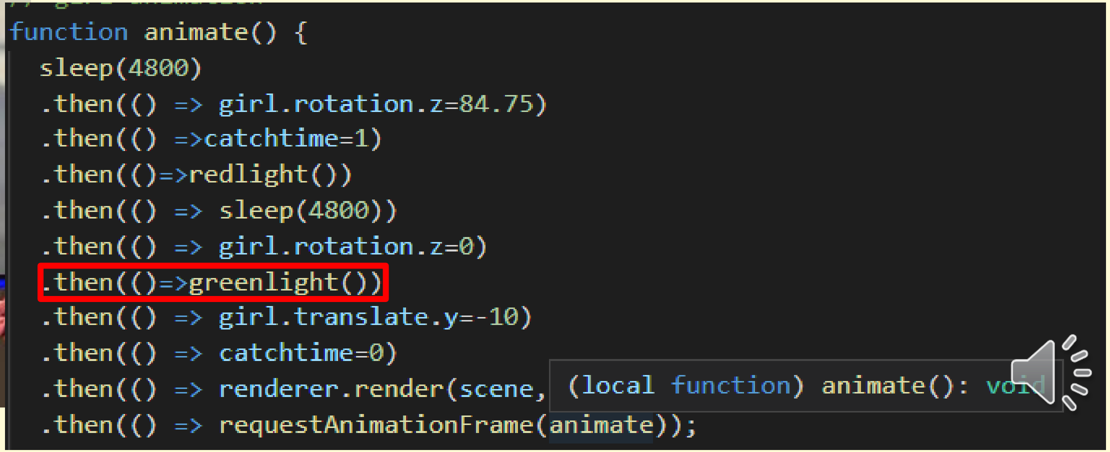
Tech. details

3D text





```
function greenlight()
{
    let material=new THREE.MeshBasicMaterial({
      color:0x00ff00,
      wireframe:false
    });
    let text_mesh=new THREE.Mesh(geometry,material);
    text_mesh.position.z=1000;
    text_mesh.rotation.y=-10050;
    scene.add(text_mesh);
}
```

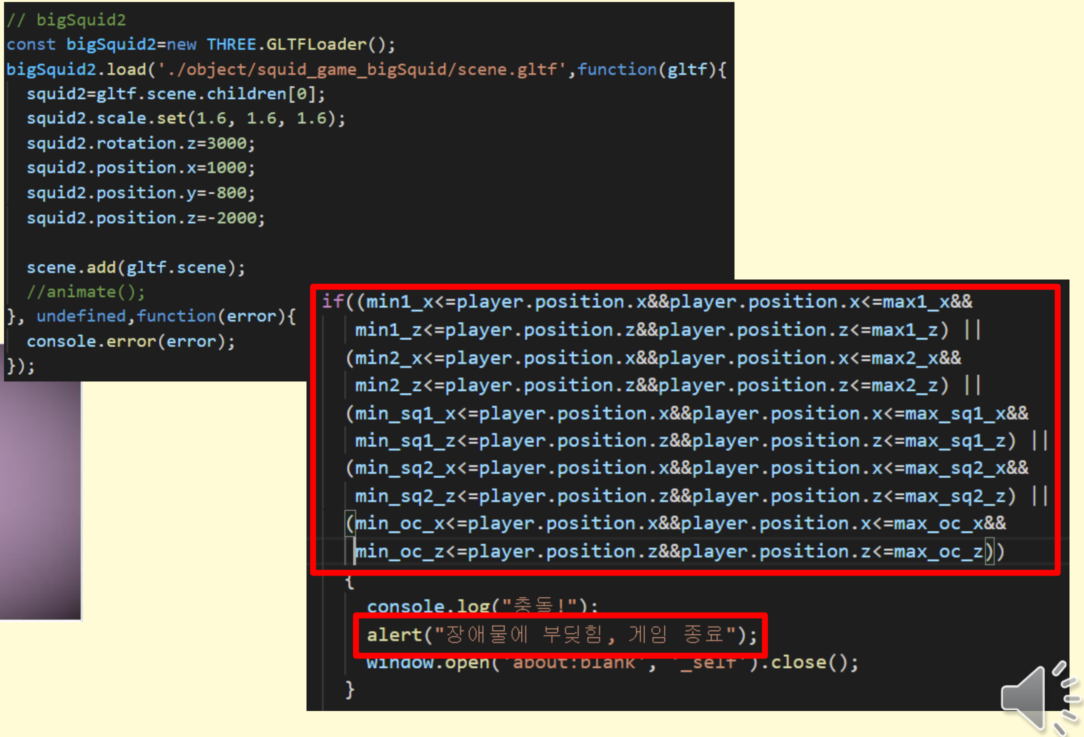


```
function animate() {
  sleep(4800)
  .then(() => girl.rotation.z=84.75)
  .then(() =>catchtime=1)
  .then(()=>redlight())
  .then(() => sleep(4800))
  .then(() => girl.rotation.z=0)
  .then(()=>greenlight())
  .then(() => girl.translate.y=-10)
  .then(() => catchtime=0)
  .then(() => renderer.render(scene, (local function) animate(): void
  .then(() => requestAnimationFrame(animate));
```

Tech. details

Obstacle
(BigSquid)

No move
Obstacle



```
// bigSquid2
const bigSquid2=new THREE.GLTFLoader();
bigSquid2.load('./object/squid_game_bigSquid/scene.gltf',function(gltf){
  squid2=gltf.scene.children[0];
  squid2.scale.set(1.6, 1.6, 1.6);
  squid2.rotation.z=3000;
  squid2.position.x=1000;
  squid2.position.y=-800;
  squid2.position.z=-2000;

  scene.add(gltf.scene);
  //animate();
}, undefined,function(error){
  console.error(error);
});
```

```
if((min1_x<=player.position.x&&player.position.x<=max1_x&&
   min1_z<=player.position.z&&player.position.z<=max1_z) ||
  (min2_x<=player.position.x&&player.position.x<=max2_x&&
   min2_z<=player.position.z&&player.position.z<=max2_z) ||
  (min_sq1_x<=player.position.x&&player.position.x<=max_sq1_x&&
   min_sq1_z<=player.position.z&&player.position.z<=max_sq1_z) ||
  (min_sq2_x<=player.position.x&&player.position.x<=max_sq2_x&&
   min_sq2_z<=player.position.z&&player.position.z<=max_sq2_z) ||
  (min_oc_x<=player.position.x&&player.position.x<=max_oc_x&&
   min_oc_z<=player.position.z&&player.position.z<=max_oc_z))
  {
    console.log("충돌!");
    alert("장애물에 부딪힘, 게임 종료");
    window.open('about:blank', '_self').close();
  }
```

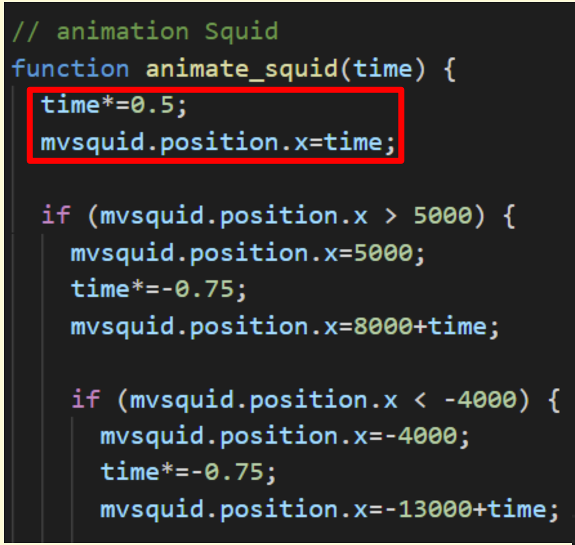
https://sketchfab.com/feed

Tech. details

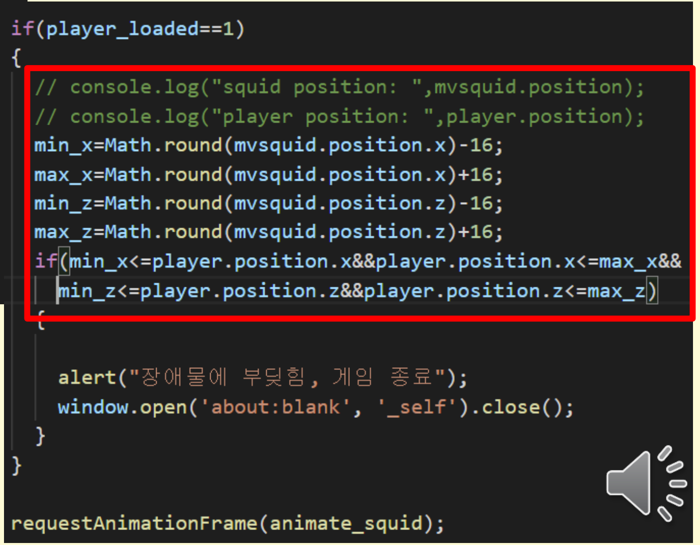
Obstacle
(Small Squid)

Move Obstacle

https://sketchfab.com/feed



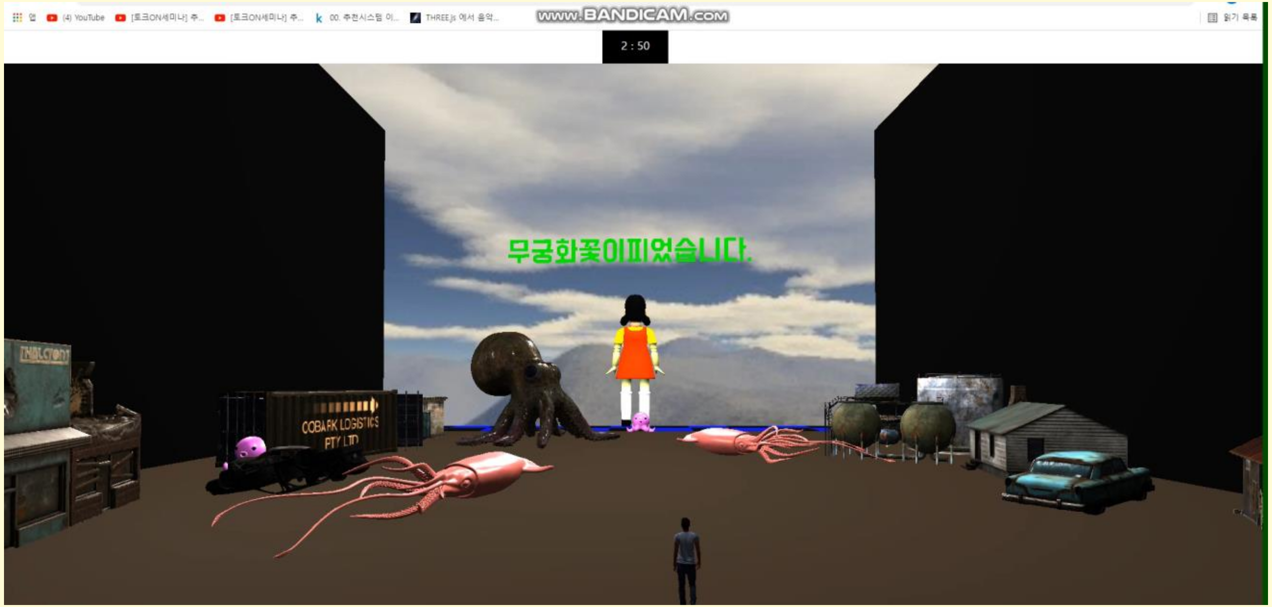
```
// animation Squid
function animate_squid(time) {
  time*=0.5;
  mvsquid.position.x=time;

  if (mvsquid.position.x > 5000) {
    mvsquid.position.x=5000;
    time*=-0.75;
    mvsquid.position.x=8000+time;

    if (mvsquid.position.x < -4000) {
      mvsquid.position.x=-4000;
      time*=-0.75;
      mvsquid.position.x=-13000+time;
```



```
if(player_loaded==1)
{
  // console.log("squid position: ",mvsquid.position);
  // console.log("player position: ",player.position);
  min_x=Math.round(mvsquid.position.x)-16;
  max_x=Math.round(mvsquid.position.x)+16;
  min_z=Math.round(mvsquid.position.z)-16;
  max_z=Math.round(mvsquid.position.z)+16;
  if(min_x<=player.position.x&&player.position.x<=max_x&&
    min_z<=player.position.z&&player.position.z<=max_z)
  {

    alert("장애물에 부딪힘, 게임 종료");
    window.open('about:blank', '_self').close();
  }
}

requestAnimationFrame(animate_squid);
```

Demo

Win

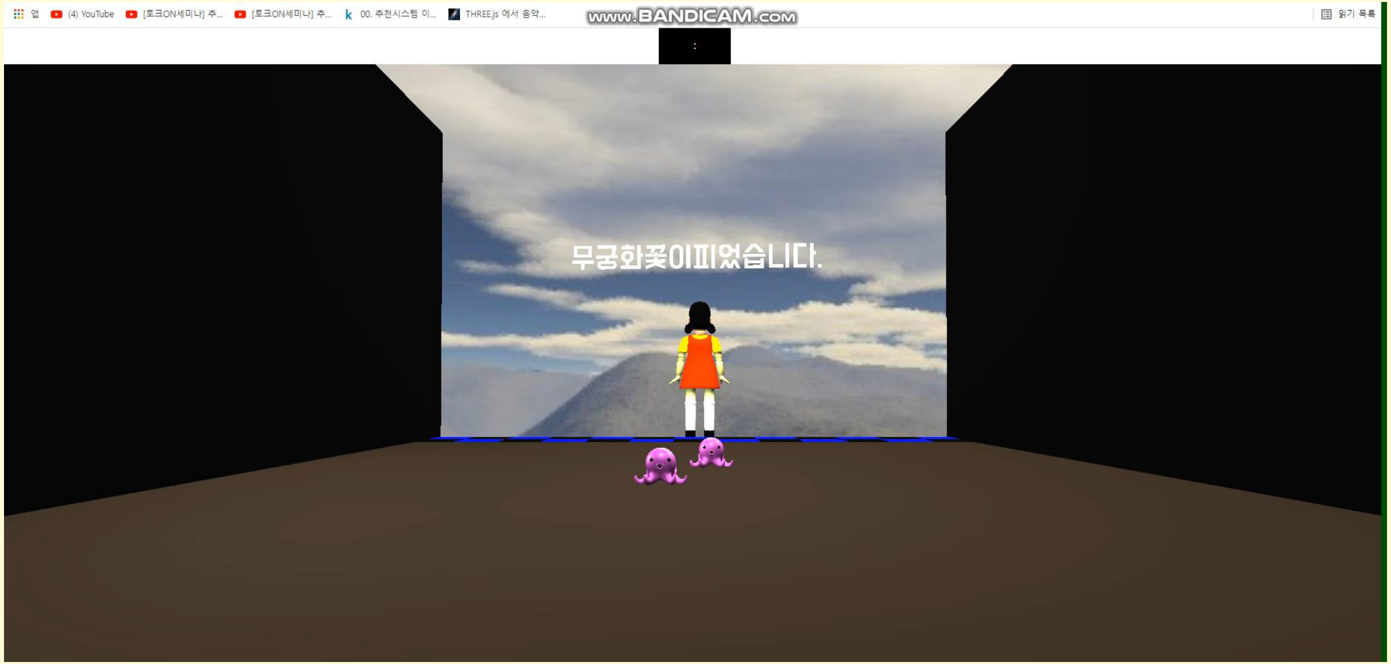


2 : 50

무궁화꽃이피었습니다.

Demo

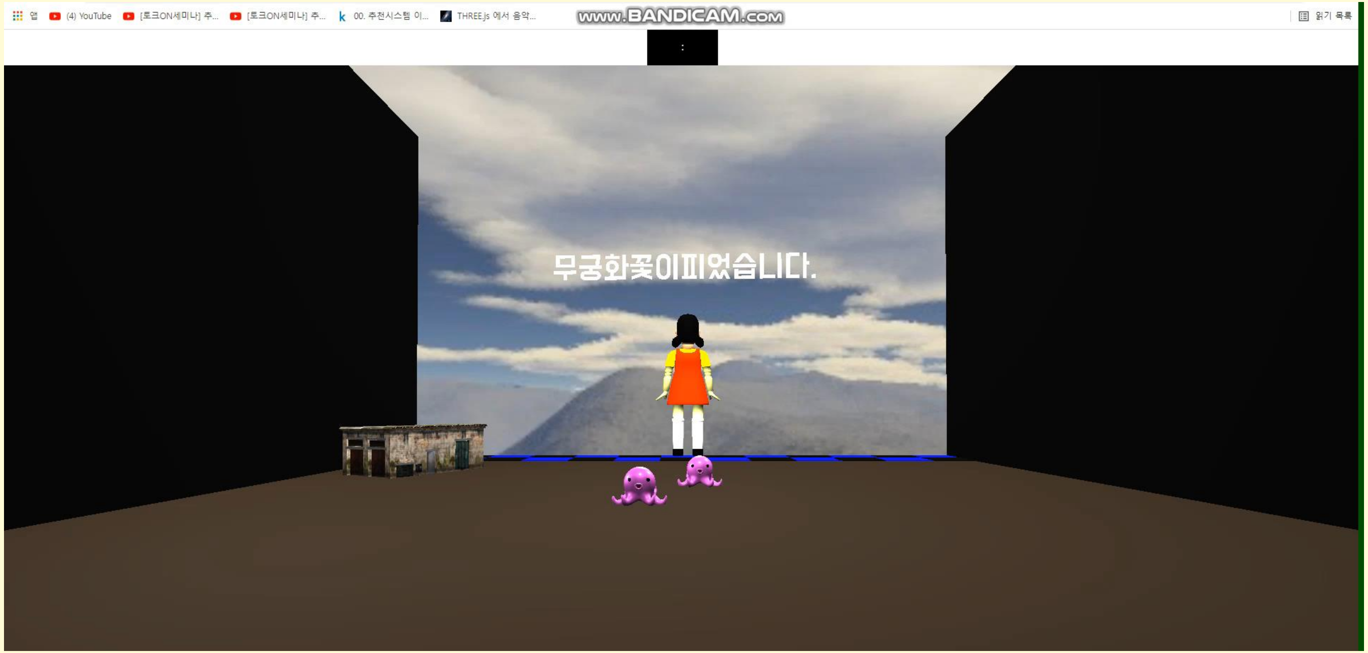
Caught by tagger



무궁화꽃이피었습니다.

Demo

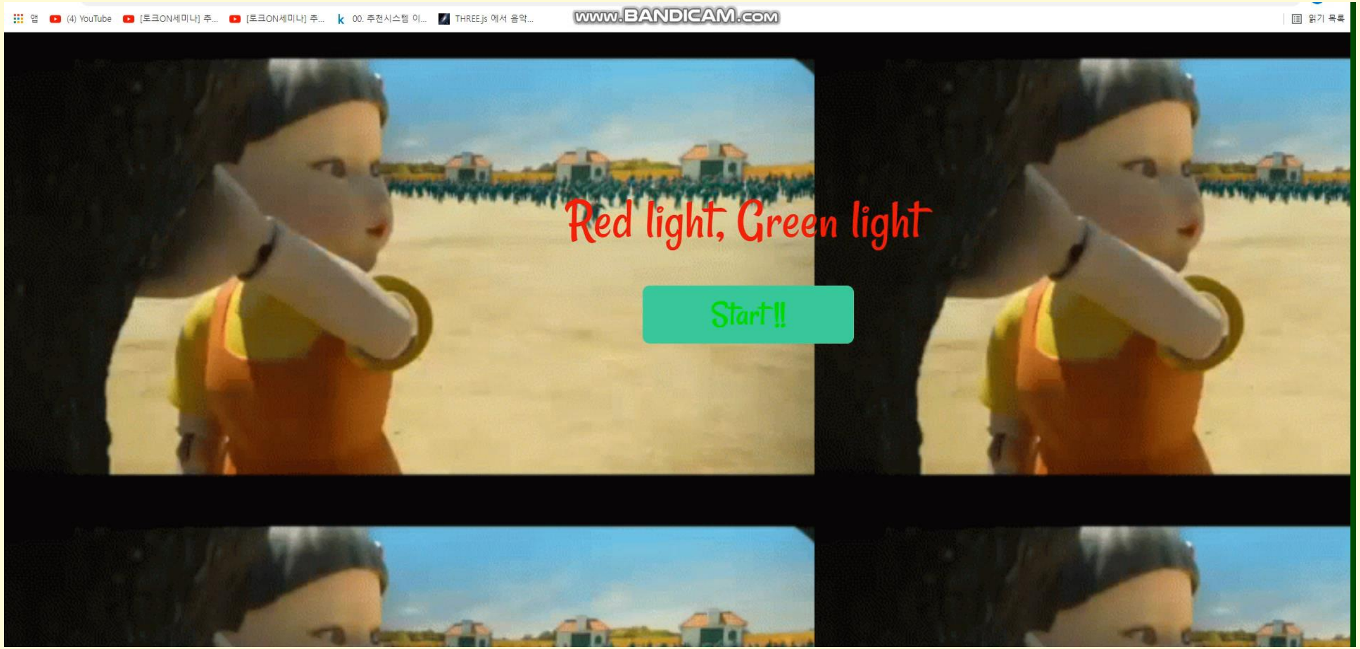
Obstacle Collision



무궁화꽃이피었습니다.

Demo

Hard mode



04 - Member role

Team members information
& Duties division

Member Role

장재혁
201533667
jjwjjw215@gmail.com

Camera Rotation,
Event Handler(Keyboard),
Animation model
Insertion

심우석
201636417
qkqh8639@gmail.com

Skybox + landscape,
Set light source,
Sound effect,
obstacle animation

이민서
201835486
lmslmsms0616@gmail.com

obstacle collision,
Event Handler,
Animation

조병근
201835528
jbg0528@gmail.com

Make movement
restricted area,
Event Handler,
Animation

Github

Github: https://github.com/dntjr41/Graphics_termProject/wiki

Youtube: https://www.youtube.com/watch?v=leVCNvBxHCw

THANKS

Do you have any questions?