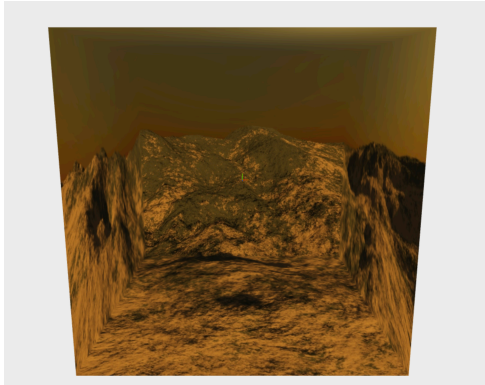


배경넣기

배경은 정육면체 안에 내가 들어와있다고 생각하면 이해하기 쉽다.

내가 보는 시야보다 정육면체가 작으면 이렇게 밖에서 볼 수 있다. 그러니 내가 볼 수 있는 시야각 보다 정육면체를 크게 만들어주면 배경으로 사용할 수 있다.



과정)

정육면체 모양의 배경을 제공하는 사이트가 있으니 참고하여 다운받고 static폴더안에 넣어준다.

1.BoxGeometry 설정해준다.(배경이 될 정육면체)

```
const skyGeometry = new THREE.BoxGeometry(600,600,600)
```

2.박스에 적용시킬 texture을 설정한다.

```
const texture = new THREE.TextureLoader().load('../static/파일명.jpg')
```

3.material에 적용

```
const skyMaterial= new THREE.MeshStandardMaterial({  
    map:texture,})
```

4.scene에 추가

```
const sky= new THREE.Mesh(skyGeometry,skyMaterial)
```

5.정육면체 밖이 아닌 안쪽에 이미지들을 넣어주기 위해서 설정 (설정 안하면 디폴트로 바깥)

```
skyMaterial.side = THREE.Backside
```

위와 같은 과정으로 하면 배경이 설정된다.

그러나 똑같은 배경이 6면을 이루고 있으므로 map을 사용하여 다른 이미지를 사용하여 실제 배경같은 느낌을 주도록한다.

과정)

1.이미지가 들어갈 배열하나를 만들어준다.

```
const skyMaterialArray = []
```

2.texture 변수를 만들었던것 처럼 6개의 변수를 만들어준다.
그리곤 만들어놓은 배열에 넣어준다.

```
const skyMaterialArray = []
const texture_ft = new THREE.TextureLoader().load(
  '../static/background/arid_ft.jpg'
)
const texture_bk = new THREE.TextureLoader().load(
  '../static/background/arid_bk.jpg'
)
const texture_dn = new THREE.TextureLoader().load(
  '../static/background/arid_dn.jpg'
)
const texture_up = new THREE.TextureLoader().load(
  '../static/background/arid_up.jpg'
)
const texture_rt = new THREE.TextureLoader().load(
  '../static/background/arid_rt.jpg'
)
const texture_lf = new THREE.TextureLoader().load(
  '../static/background/arid_lf.jpg'
)
skyMaterialArray.push(
  new THREE.MeshStandardMaterial({
    map: texture_ft,
  }),
  new THREE.MeshStandardMaterial({
    map: texture_bk,
  }),
  new THREE.MeshStandardMaterial({
    map: texture_up,
  }),
  new THREE.MeshStandardMaterial({
    map: texture_dn,
  }),
  new THREE.MeshStandardMaterial({
    map: texture_rt,
  }),
  new THREE.MeshStandardMaterial({
    map: texture_lf,
  })
)
```

3. 배열을 반복문을 통해 설정을 시켜준다.

```
for(let i =0; i<6 ;i++){
  skyMaterialArray[i].side = THREE.BackSide }
```

4. mesh적용해주기

```
const sky = new THREE.Mesh(skyGeometry ,skyMaterialArray)
scene.add(sky)
```