

오픈소스 SW 15조
팀프로젝트 발표

오픈소스 SW
the building 0

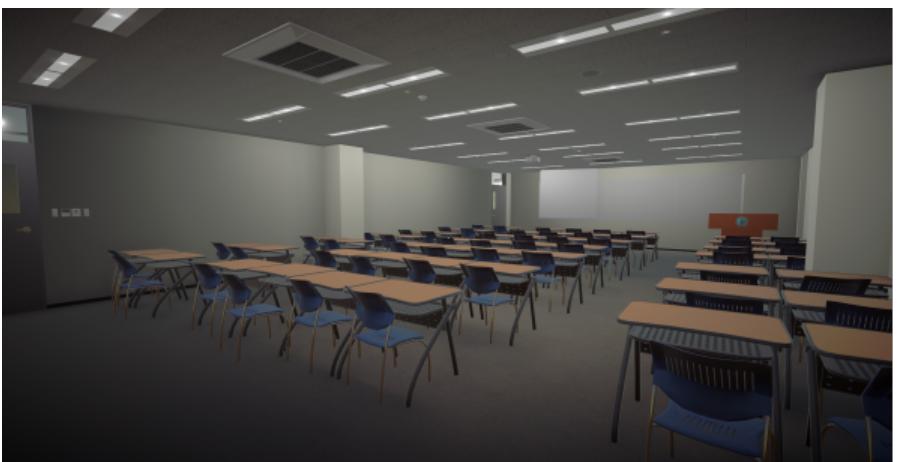
오픈소스
the building 0

0번 건물

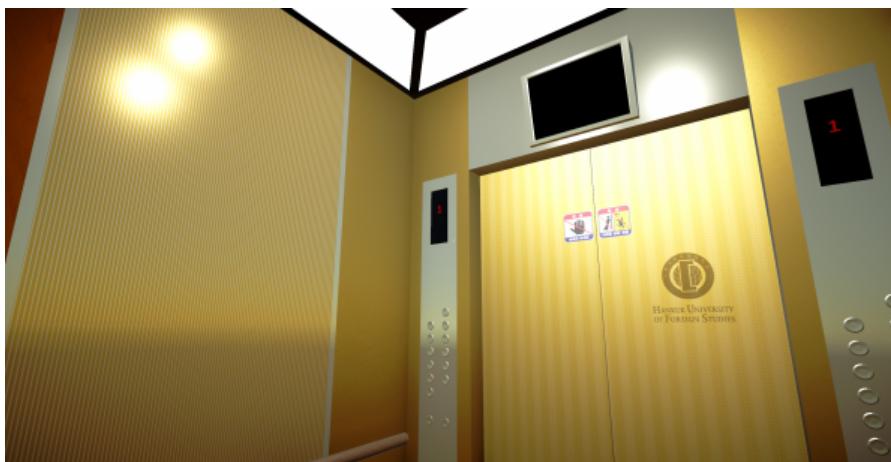
the building 0

- ▶ 게임 '8번 출구'를 모티브로 한 공포 탈출 게임
- ▶ 끝없이 반복되는 복도에서 놓고 온 물건을 찾아 탈출해야 하는 스토리
- ▶ 복도의 평상시 모습일 때 '이변'을 발견하지 못하고 지나치면 무한히 복도에서 헤매게 됨

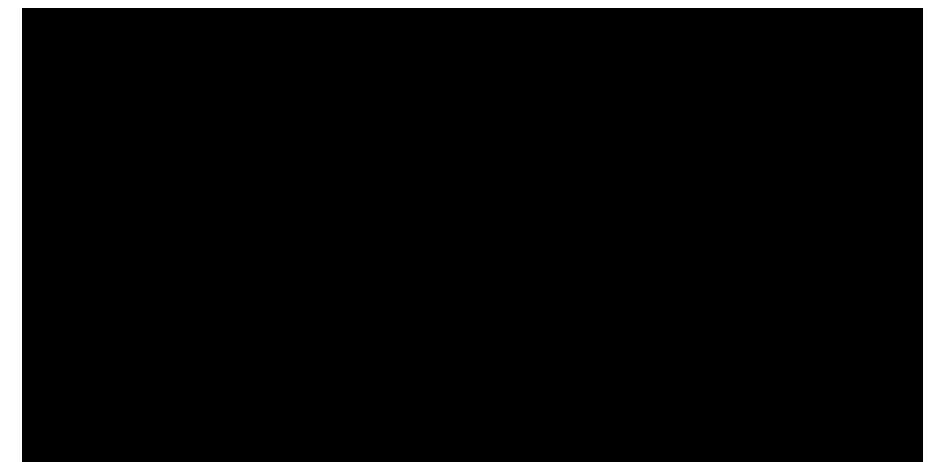
게임 맵



<교실 내부>



<엘리베이터>



<복도>

0번 건물 개인 방법

1인칭 시점의 공포 탈출 게임



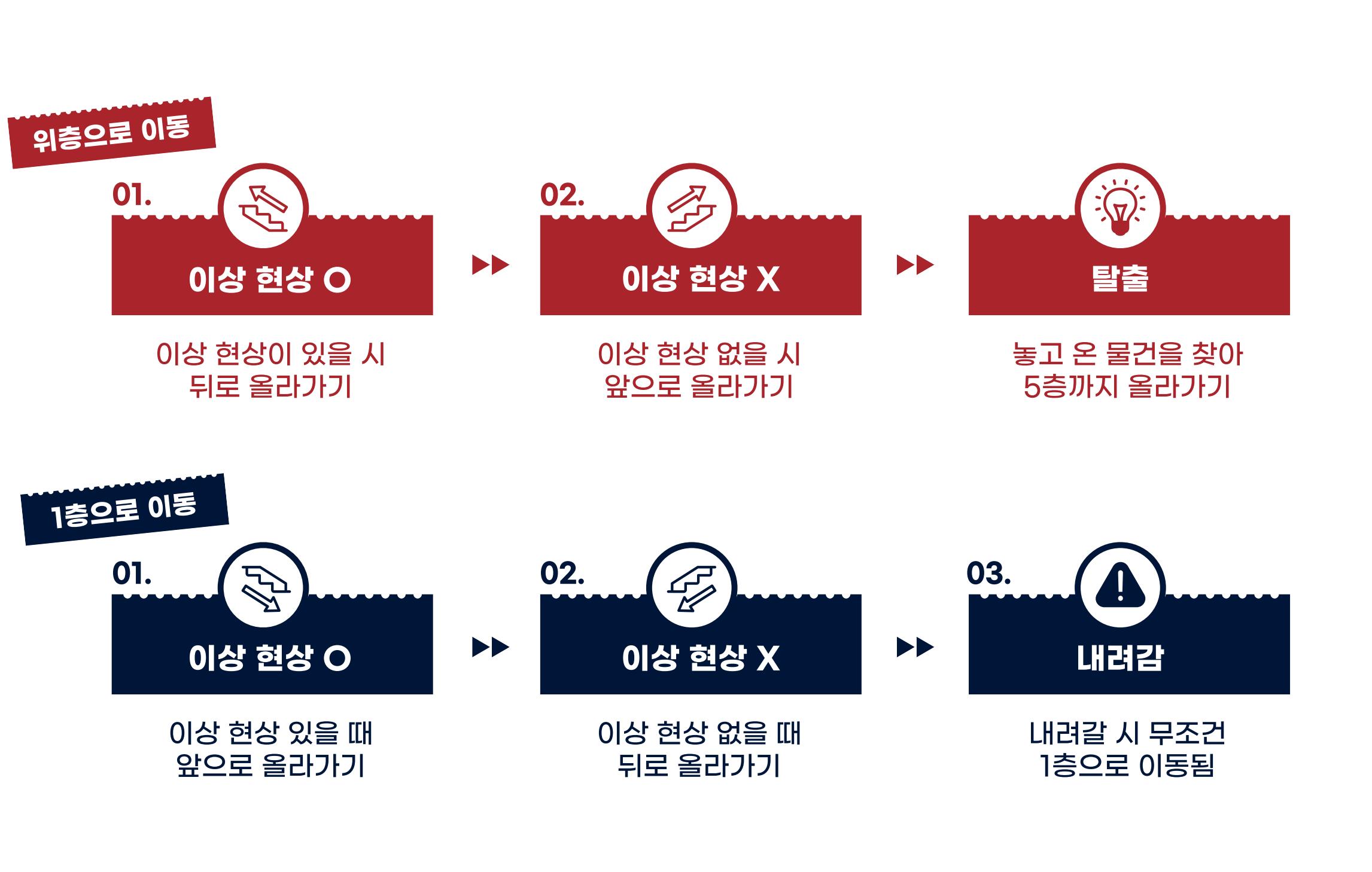
- 01**  **플레이어의 이동**
WASD 또는 방향키 이용
- 02**  **플레이어 시점**
1인칭 시점
마우스를 사용해 시점 이동
- 03**  **달리기**
SHIFT 키를 사용해 대쉬
- 04**  **점프**
스페이스바를 사용해 점프
- 05**  **상호작용**
상호작용 가능 시 빨간 점으로
변경됨
- 06**  **놓고 온 물건**
놓고 온 물건을 찾아서 탈출

0번 건물

클리어 방법

시작 지점: 1층
탈출 지점: 5층

놓고 온 물건을 찾아
5층으로 이동하면 탈출 성공

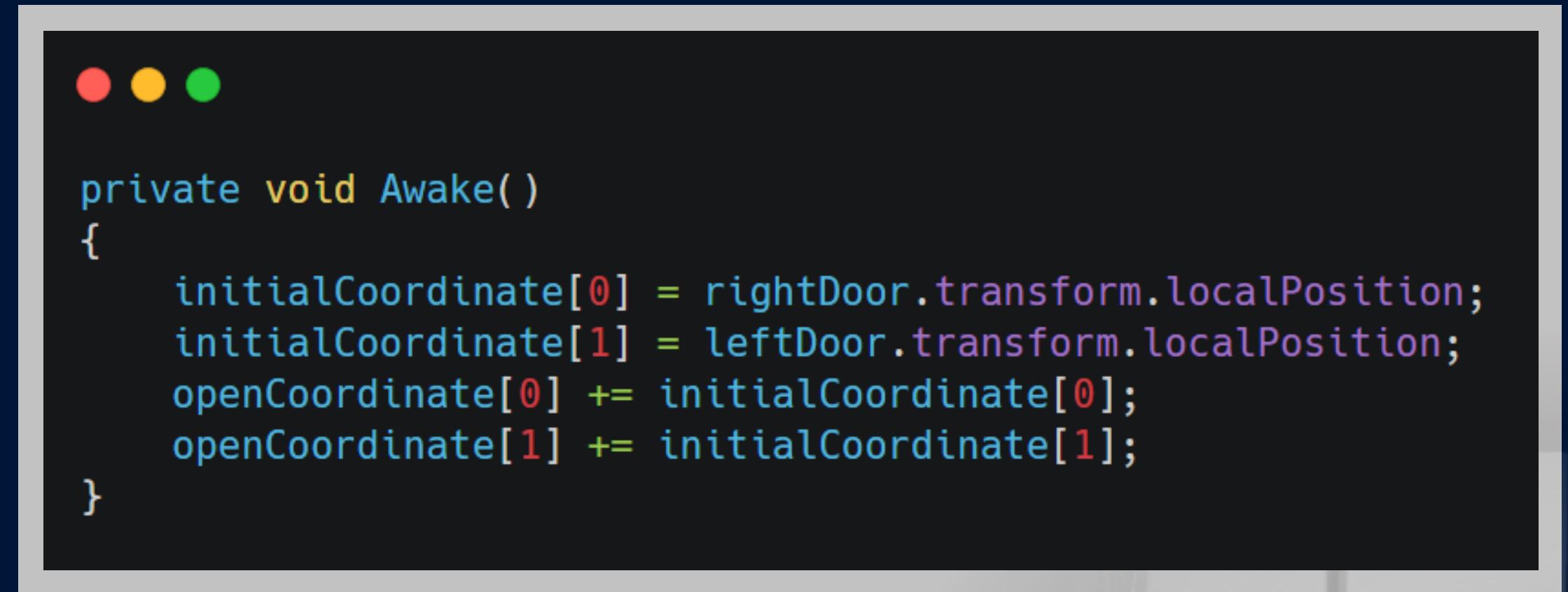




```
public GameObject rightDoor, leftDoor;  
public bool isOpen = false;  
private Vector3[] initialCoordinate = new Vector3[2];  
private Vector3[] openCoordinate = { new(-0.55f, 0, 0), new(0.55f, 0, 0) };  
private Vector3 rightVelocity = Vector3.zero;  
private Vector3 leftVelocity = Vector3.zero;
```

- ▶ **rightDoor, leftDoor**: 엘리베이터의 오른쪽과 왼쪽 문을 나타내는 GameObject
- ▶ **isOpen**: 문이 열려 있는지 여부를 나타내는 불리언 변수
- ▶ **initialCoordinate**: 문이 처음 시작할 때의 위치를 저장하는 Vector3 배열
- ▶ **openCoordinate**: 문이 열렸을 때의 목표 위치를 저장하는 Vector3 배열
- ▶ **rightVelocity, leftVelocity**: 문이 이동할 때의 속도를 저장하는 Vector3

ElevatorController Script 소개



```
● ● ●

private void Awake()
{
    initialCoordinate[0] = rightDoor.transform.localPosition;
    initialCoordinate[1] = leftDoor.transform.localPosition;
    openCoordinate[0] += initialCoordinate[0];
    openCoordinate[1] += initialCoordinate[1];
}
```

- ▶ `Awake()`: 객체가 활성화될 때 호출
- ▶ `initialCoordinate`: 문이 처음 위치한 좌표 저장
- ▶ `openCoordinate`: 초기 좌표에 열릴 때의 이동 값을 더하여 문이 열릴 때의 최종 위치 설정

ElevatorController Script 소개

```
private void Update()
{
    if (isOpen)
    {
        rightDoor.transform.localPosition = Vector3.SmoothDamp(rightDoor.transform.localPosition,
openCoordinate[0], ref rightVelocity, 0.5f);
        leftDoor.transform.localPosition = Vector3.SmoothDamp(leftDoor.transform.localPosition,
openCoordinate[1], ref leftVelocity, 0.5f);
    }
    else
    {
        rightDoor.transform.localPosition = Vector3.SmoothDamp(rightDoor.transform.localPosition,
initialCoordinate[0], ref rightVelocity, 0.5f);
        leftDoor.transform.localPosition = Vector3.SmoothDamp(leftDoor.transform.localPosition,
initialCoordinate[1], ref leftVelocity, 0.5f);
    }
}
```

Update(): 매 프레임마다 호출

isOpen이 true인 경우:

문을 부드럽게 열기 위해 Vector3.SmoothDamp 사용

현재 위치에서 목표 위치(openCoordinate)로 일정 시간(0.5f) 동안 이동

isOpen이 false인 경우:

문을 부드럽게 닫기 위해 Vector3.SmoothDamp 사용

현재 위치에서 초기 위치(initialCoordinate)로 일정 시간(0.5f) 동안 이동

감사합니다.

오픈소스 SW 15조

202102385 이동윤

202302255 이서진

202303436 이서희

202302963 이승혁