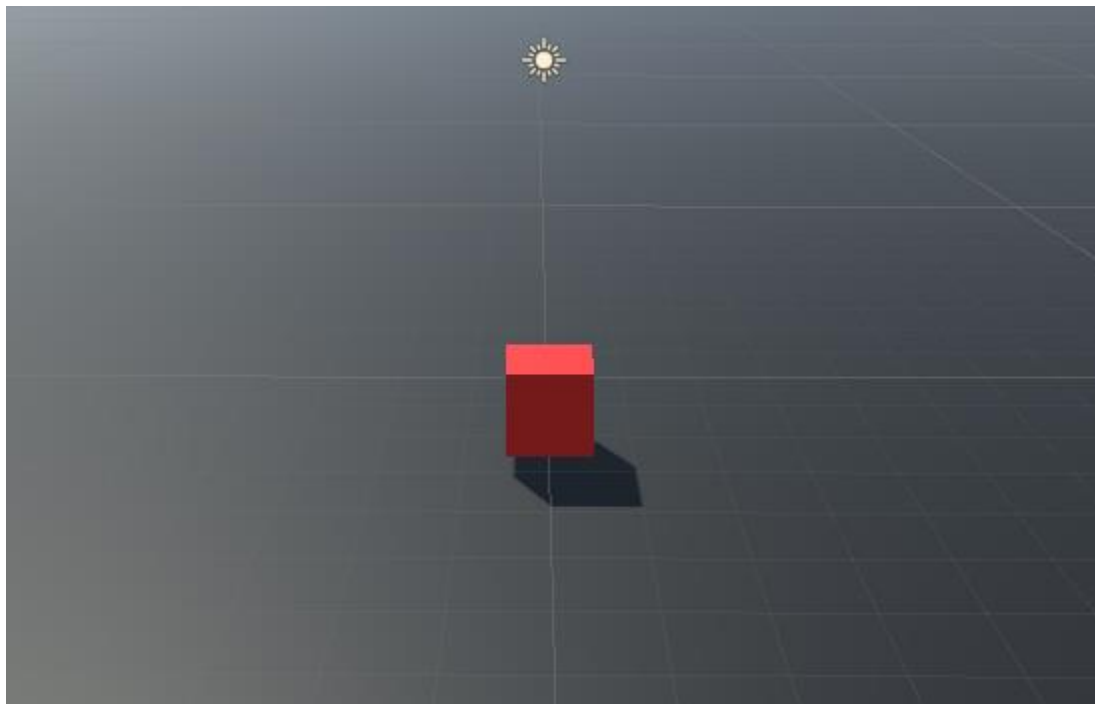
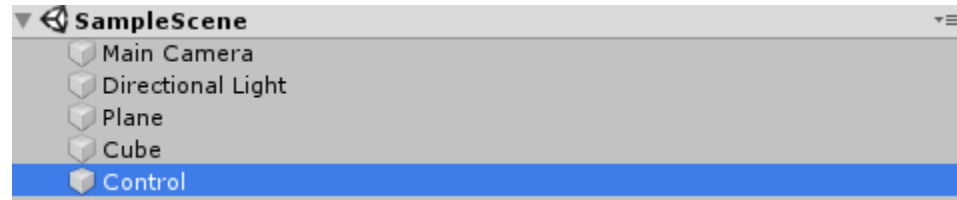


접근 지정자 보충 설명2

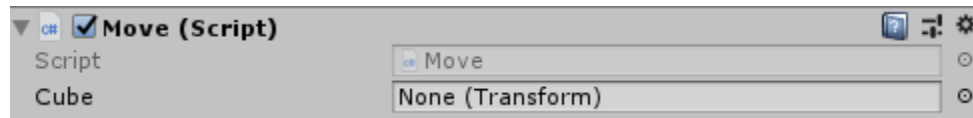
Scene에는 큐브가 올려져 있습니다



또한 Hierarchy에는 Control 이라는 무형 오브젝트가 있구요



Move 라는 스크립트가 달려있습니다



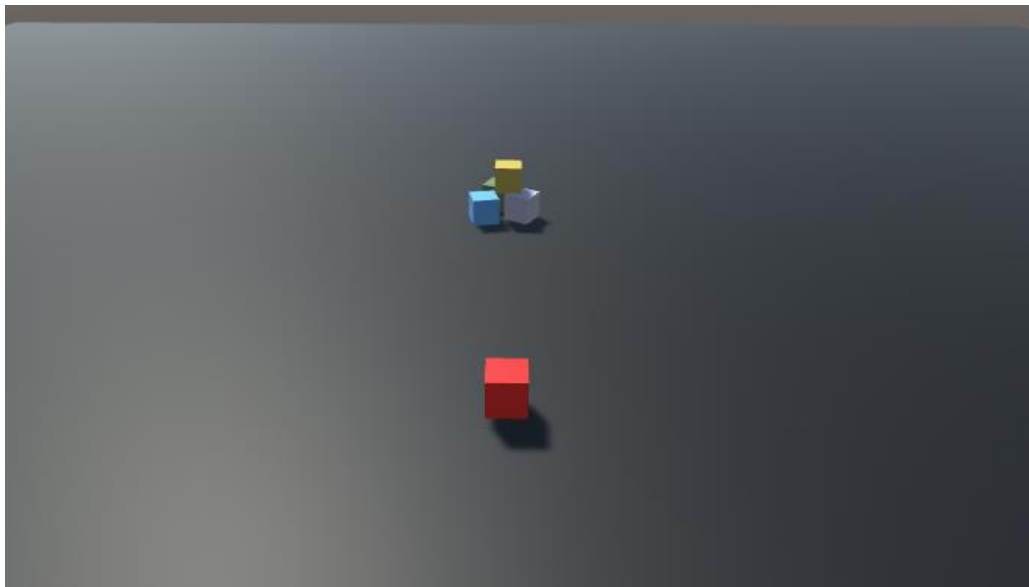
Control 스크립트를 살펴보겠습니다

```
public class Move : MonoBehaviour
{
    public Transform Cube;

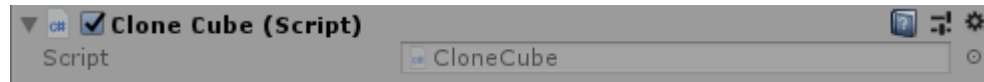
    void Update()
    {
        if(Cube != null) // Cube를 초기화 하지 않기에 적어두는 조건 문
        {
            if (Input.GetKey(KeyCode.UpArrow)) //위 방향키를 누르면 이동
            {
                Cube.Translate(10f * Time.deltaTime, 0f, 0f);
            }
        }
    }
}
```

Cube변수는 public 권한이기 때문에 값을 외부에서 넣어준다면 움직일 수 있겠네요!

계속해서 큐브를 클릭하면 OnMouseDown 함수를 통해 큐브가 생성이 됩니다



생성되는 큐브 Prefab에는 CloneCube라는 스크립트가 달려있네요

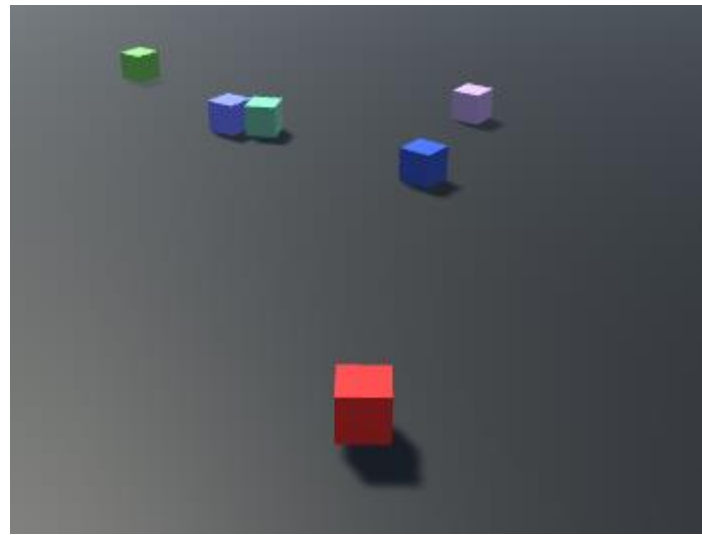
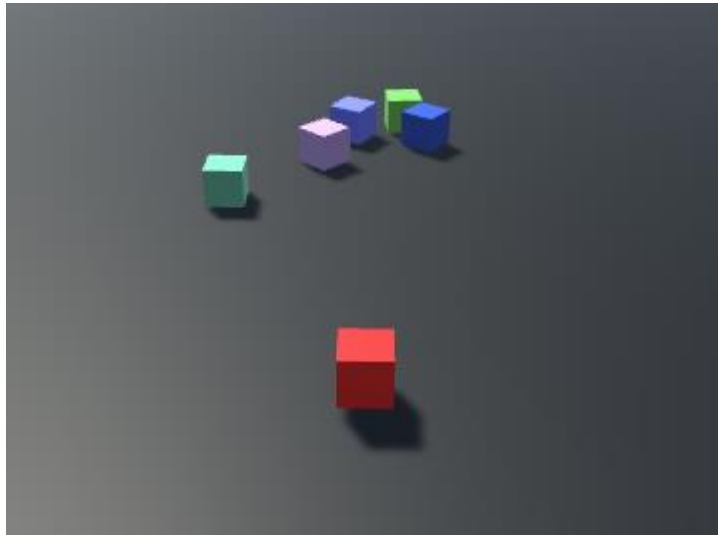


CloneCube 스크립트를 살펴보겠습니다

```
public class CloneCube : MonoBehaviour
{
    Move m; //스크립트를 자료형으로 선언
    private void Start()
    {
        m = GameObject.Find("Control").GetComponent<Move>(); //Control을 찾아서 Move를 담아줌
    }
    private void OnMouseDown()
    {
        m.Cube = this.gameObject.GetComponent<Transform>(); //클릭될 경우, 현재 큐브의 Transform 정보를 Move 스크립트에 삽입한다!
    }
}
```

이러면, 복사된 큐브가 클릭될 경우에 Move 스크립트에 해당 Transform 정보를 넘겨줄 수 있겠네요!

따라서, 클릭된 오브젝트를 움직일 수 있게 되는 것이죠.



그림으로 표현하면 이렇습니다

Control(GameObject)

Move.cs

```
public Transform Cube;
```

⋮

[큐브에 값이 있으면 움직인다]

⋮

삽입

클론

Cube0(GameObject)

CloneCube.cs

```
Move m;  
⋮  
private void OnMouseDown()  
{  
    m.Cube = this.gameObject.GetComponent<Transform>();  
}
```

Cube1(GameObject)

CloneCube.cs

[위와 동일한 내용]

Cube2(GameObject)

CloneCube.cs

[위와 동일한 내용]

⋮

더 궁금한 점이 있으시면 연락주세요~