

Code Conventie

Variabelen

- De variabelen die je aanmaakt moeten allemaal in het Engels geschreven zijn.
- Alle variabelen die geschreven worden moeten volledig worden geschreven, zodat het een betere readability geeft.
- Alle variabelen moet standaard private zijn. Op een paar uitzonderingen na als het niet anders kan, zoals wanneer je een variabelen in een ander script nodig hebt of in de inspector iets wilt aanpassen.

```
private PlayerGravity _gravity;  
private Vector3 _currentRotation;  
private IEnumerator _routine;
```

- Alle variabelen die private zijn beginnen met een underscore.
- Wanneer een private variabelen een [SerializeField] heeft dan begint die niet met een underscore, want zo kan je zien dat die variabelen aanpasbaar is in de inspector.
- Alle variabelen schrijven we met camelCase. Dat houdt in dat alle variabelen beginnen met een kleine letter. Elk nieuw woord dat eraan vast zit moet met een hoofletter beginnen.

```
public FPControl.PlayerControlsActions inputControls;  
public InputParse input;  
public bool isTurning;  
[SerializeField] private GameObject world;
```

Funcities

- Alle functies zijn in het Engels geschreven en beginnen met een hoofdletter.
- Als de naam van een functie uit meerder woorden bestaat gebruik je PascalCasing.
- Alle functie namen zijn werkenwoorden, zodat het duidelijk is wat het doet.

```
IEnumerator RotateLevel(Vector3 dir)
{
    isTurning = true;
    _gravity.SwitchGravity();
    var targetAngle = _currentRotation + directions[dir]; //Kijkt naar de angle waar die naartoe moet.
    float time = 0;
    float duration = 0.5f;
    while (time < duration) //Wanneer de angle dat die is niet gelijk is aan de angle waar die naartoe moet.
    {
        time += Time.deltaTime;
        var newPosition = Vector3.Lerp(_currentRotation, targetAngle, time / duration);
        newPosition.x %= 360;
        newPosition.y %= 360;
        newPosition.z %= 360;
        world.transform.eulerAngles = newPosition;
        yield return null;
    }
    _currentRotation += directions[dir];
    isTurning = false;
    _gravity.SwitchGravity();
}
```

Classes

- Alle classes zijn in het Engels geschreven.
- De naam van een class begint altijd met een hoofdletter.
- Wanneer de class naam uit meerdere woorden bestaat dan gebruiken we PascalCasing, dus we starten met een hoofdletter bij elk nieuw woord dat eraan vastzit.
- Een class is altijd een 'object'. Bijvoorbeeld een Generator, PlayerMovement, Settings, Library.

```
public class MovementPlayer : MonoBehaviour
public class LevelRotater : MonoBehaviour
```