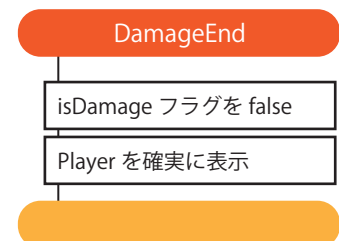
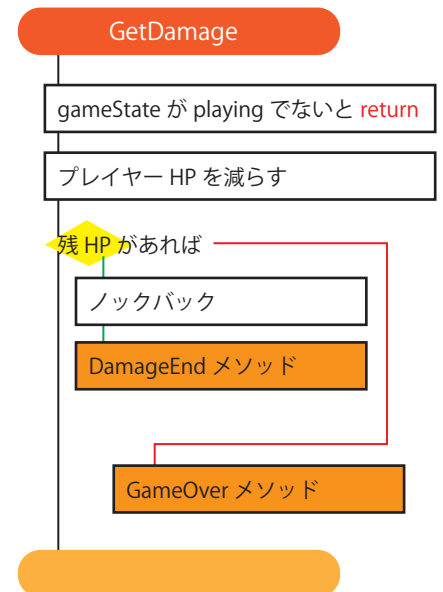
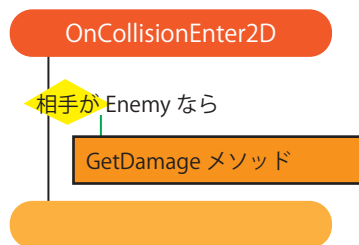
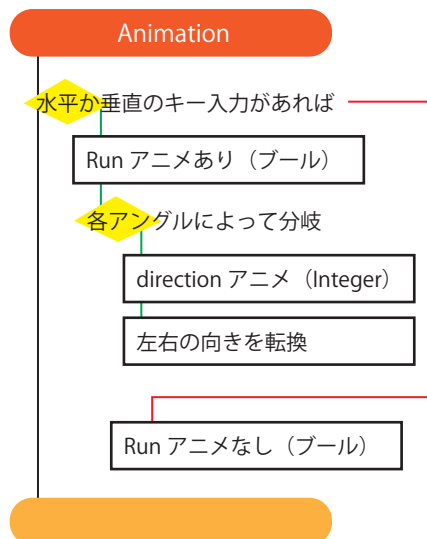
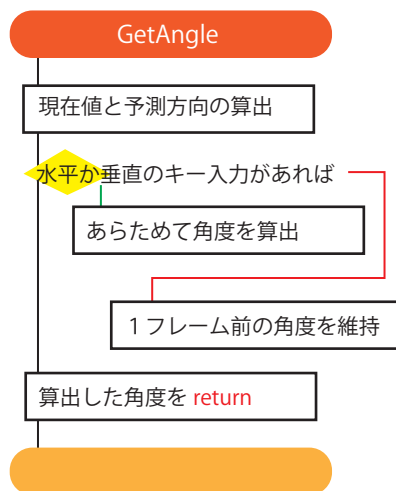


# PlayerController

```
public float PlayerSpeed //Player 移動スピード
private float axisH // 水平方向の入力値
private float axisV // 垂直方向の入力値
public float angleZ // 向いている角度
private Rigidbody2D rbody //Rigidbody2D コンポーネント
private Animator anime //Animator コンポーネント
public GameObject spotLight // ライトの子オブジェクト
private bool isDamage // ダメージ中フラグ
```



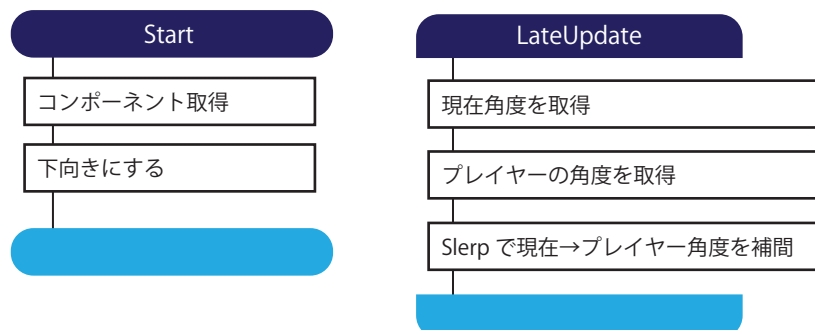
## Shooter

```
private PlayerController playerCnt //PlayerController コンポーネント
public GameObject billPrefab // 生成されるプレハブ
public float shootSpeed // 投擲物のスピード
public float shootDelay // 発射間隔
private bool inAttack // 攻撃中フラグ
```



## SpotLight

```
private PlayerController playerCnt //PlayerController コンポーネント
public float rotationSpeed // 回転速度
```



# GameManager

```
public static GameState gameState // ゲームのステータス

public static bool[] doorsOenedState // ドアの開閉状況

public static int key1 // 鍵 1 の持ち数
public static int key2 // 鍵 2 の持ち数
public static int key3 // 鍵 3 の持ち数
public static bool[] keysPickedState // 鍵の取得状況

public static int bill // お札の持ち数
public static bool[] itemsPickedState // アイテムの取得況

public static bool hasSpotLight // スポットライトを持っているか
public static int playerHP // プレイヤーの HP
```

public enum GameState

- playing
- talk
- gameover
- gameclear
- stop
- ending

## 【Player】

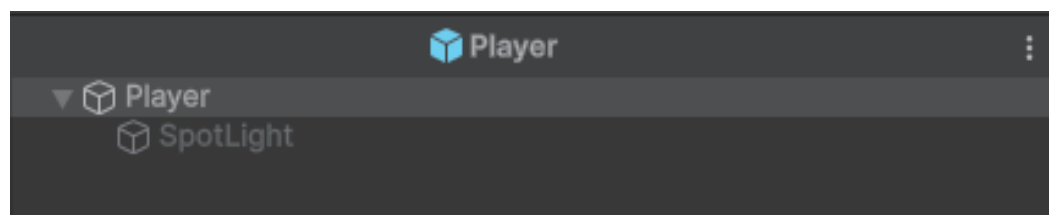
(特徴)

Player は上下左右でアニメを切り替えながら移動ができます。

スペースキーでお札を投擲できます (Shooter.cs)

子オブジェクトに SpotLight がいて、プレイヤーにあわせて方向転換します (SpotLight.cs)

Player の HP は GameManager が管理し、ダメージをもらうと点減しながらノックバックします



angleZ

- PlayerController.cs
- Shooter.cs

angleZ を使って方向を決める

SpotLight

- SpotLight