```
// Player.cs

using UnityEngine;
using UnityEngine.AI;
using System.Collections;

public enum PlayerState
{
    InSunShine,
    InShadow,
}

public class Player : MonoBehaviour
{
    //GameObjectの定義
    public GameObject PlayerModel;        // プレイヤーモデル
    public GameObject LookPoint;          // エネミーに見られる位置
    public float walkSpeed = 4.0f;       //歩くスピード
    public float rotateSpeed = 120.0f;    //回転速度(度/秒)
    public float HP;                      // プレイヤーのＨＰ

    private SunLightChecker m_SunLightChecker;              // 太陽（ライト）が何個当たっているか調べる
    private PlayerState m_State = PlayerState.InShadow;     // ステートから影かどうか
    private Animator animator;                              // プレイヤーのアニメター
    public Vector3 m_velocity;                              // プレイヤー座標
    private CharacterController controller;                 // キャラクターコントローラー
    private GameObject m_possesion;                         // 所持しているオブジェクト
    public bool death = false;                              // 死亡フラグ
    private NavMeshAgent agent;                             // 経路探索を行うナビゲーションシステム
    public bool isWallHit;

    /// <summary>
    /// プレイヤーの日照状態
    /// </summary>
    public PlayerState PlayerState
    {
        get { return m_State; }
        set { m_State = value; }
    }

    void Start()
    {
```

```
        m_SunLightChecker = GetComponent<SunLightChecker>();          // SunSunLightCheckerの成分取得
        controller = GetComponent<CharacterController>();             // キャラクターコントローラーの成分の
取得
        animator = GetComponent<Animator>();
        agent = GetComponent<NavMeshAgent>();                         // NavMeshAgentの成分の取得

        isWallHit = false;
    }

    //void Awake()
    //{
    //      DontDestroyOnLoad (this.gameObject);
    //}

    void Move()
    {
        transform.Rotate(0, Input.GetAxis("Horizontal") * rotateSpeed * Time.deltaTime, 0);
        m_velocity = transform.forward * -1 * Input.GetAxis("Vertical") * walkSpeed;
        controller.Move(m_velocity * Time.deltaTime);

        //if (GameObject.Find("Main Camera").GetComponent<ContorollerCamera>() != null)
        //{
        //      GameObject.Find("Main Camera").GetComponent<ContorollerCamera>().targetTransform =
this.transform;
        //}
    }

    void Update()
    {
        transform.position = new Vector3(transform.position.x, 0.23f, transform.position.z);

        if (m_possesion == null)
        {
            Move();
            controller.enabled = true;
            //this.GetComponent<BoxCollider>().enabled = false;
        }
        else
        {
            controller.enabled = false;
            RaycastHit hit;
            this.GetComponent<BoxCollider>().enabled = true;
```

```
        if (Physics.Raycast(transform.position, Vector3.forward, out hit, Mathf.Infinity))
        {
            print("find" + hit.collider.gameObject.name);
        }

        //エラー1
        if (!m_possesion.GetComponent<ObjectState>().IsState())
        {

            m_possesion = null;
            controller.enabled = true;
        }
    }



    // プレイヤーステートより日向か日陰かを調べる
    // もし影の中だった場合
    if (m_State == PlayerState.InShadow)
    {
        // もし 1 個以上の太陽（ライト）に当てられていた場合
        if (m_SunLightChecker.IlluminatedCount >= 1)
        {
            // 日向
            ChangeState(PlayerState.InSunShine);
        }
        else
        {
            // 影
            print("回復");
            HP += 1.0f * Time.deltaTime;
        }
    }
    else    // もし日向だった場合
    {
        // もし太陽（ライト）に当てられていなかった場合 == 太陽（ライト）が 0 個
        if (m_SunLightChecker.IlluminatedCount < 1)
        {
            // 影
            ChangeState(PlayerState.InShadow);
        }
        else
        {
```

```
                // 日向
                print("ダメージ");
                HP += -3.0f * Time.deltaTime;
            }
        }
                 HP = Mathf.Clamp (HP, 0, 300);

}

private void ChangeState(PlayerState newState)
{
    // プレイヤーステート
    m_State = newState;

    // ステートが影の場合
    if (m_State == PlayerState.InShadow)
    {
        // 半透明にする
        print("影です");
        Color color = PlayerModel.GetComponent<Renderer>().material.color;
        color.a = 0.3f;
        PlayerModel.GetComponent<Renderer>().material.color = color;
    }
    else
    {
        // 不透明にする
        print("影じゃないです");
        Color color = PlayerModel.GetComponent<Renderer>().material.color;
        color.a = 1.0f;
        PlayerModel.GetComponent<Renderer>().material.color = color;
    }
}

private void OnControllerColliderHit(ControllerColliderHit hit)
{
    if (hit.gameObject.tag == "Enemy") { HP -= 0.1f; }

    if (hit.gameObject.tag == "Glass" && m_possesion == null)
    {
        hit.gameObject.GetComponent<BoxCollider>().isTrigger = true;
    }
}
```

```
    void OnTriggerExit(Collider hit)
    {
        if (hit.gameObject.tag == "Wall" && m_possesion != null)
        {
            isWallHit = false;
        }
    }

     private void OnTriggerStay(Collider hit)
    {
        if (hit.gameObject.tag == "Wall" && m_possesion != null)
        {
            Debug.Log("当たる");
            isWallHit = true;
        }
    }
}
```