

# README\_時間管理クラス\_TIME\_MANAGER

## このクラスでは時間の正進行と逆行を管理している

- クラス内に時間の正進行と逆行を総括する変数を設けた

### is\_revtime

初期値はfalse

```
public static bool is_revtime = false;
```

型	trueの時想定される挙動	falseの時想定される挙動
public static bool	時間が逆行している	通常の進行

現状はそれぞれのギミックで時間の正進行と逆行を管理していると思われるが  
今後は効率の良い開発のために **is\_revtime** を参照しての時間進行の管理を行ってほしい

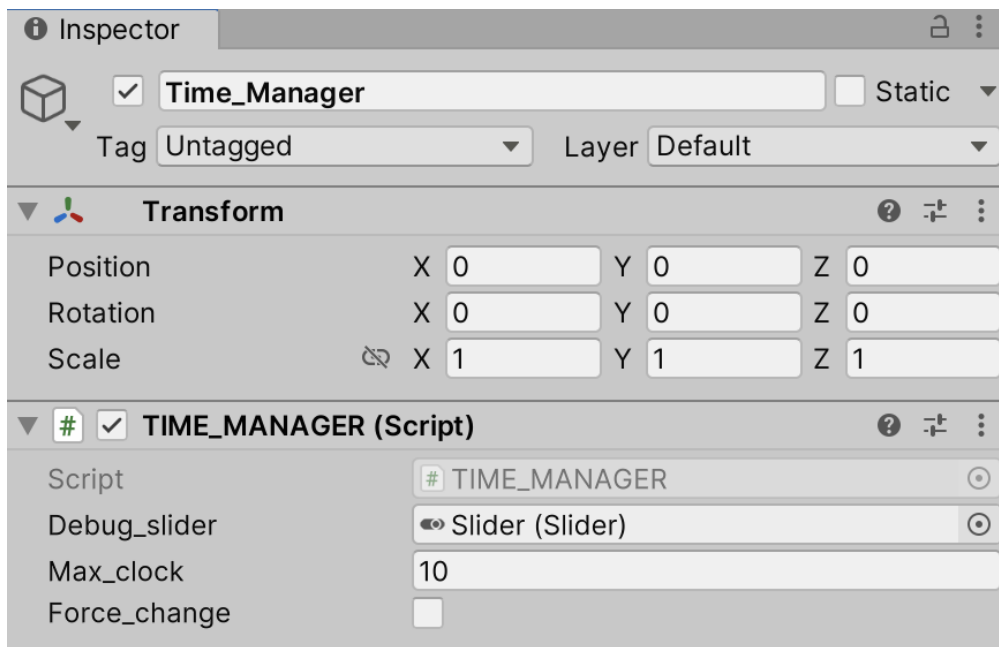
赤字部分→緑字部分のように判定部分を変更するとTIME\_MANAGERを利用できる

```
- if(各自で考えた時間管理フラグ)
- {
    それぞれのギミックの挙動
- }

+ if(TIME_MANAGER.isrevtime)
+ {
    それぞれのギミックの挙動
+ }
```

## 逆行を行える時間の制限

- 時間の逆行が可能な量をコントロールできるようにした  
(これはゲーム性の向上を狙ってのことである)



画像：

量の調節はTIME\_MANAGERの **max\_clock** の値をインスペクターから変更することによって行う

```
public int max_clock = 20;
```

残量は画面上部のゲージにて試験的に確認できるが最終的なUIデザインは要相談  
ゲージの挙動の様子は **stage\_add\_rock** シーンにて確認することができる

その他オプションとして

```
public bool Force_change = false;
```

を用意した  
インスペクター上の

チェックボックスのon / offで切り替えることができる

trueの時想定される挙動	falseの時想定される挙動
制限量を超えた時強制的に時間逆行が解除される	次にスペースキーを押すまでは逆行は解除されない