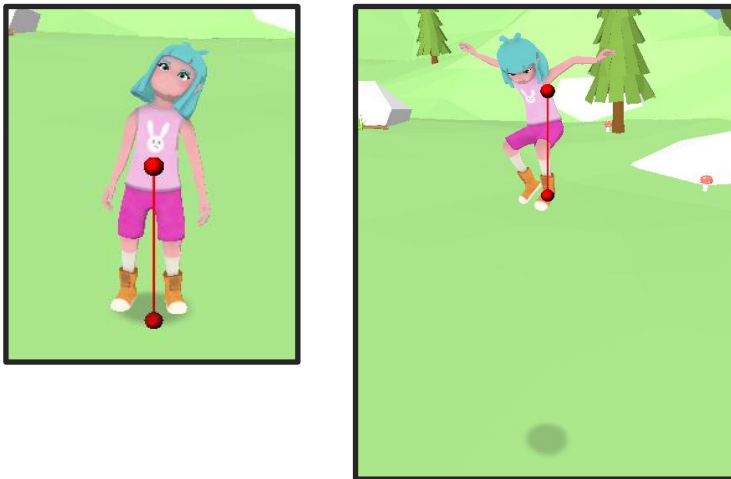


丸影描画

ジャンプを実装したら、合わせて影を実装して欲しいです。
ユーザから見て、キャラクターの着地ポイントが常に視認できることにより、
上下も含めたキャラクターの位置を把握しやすくなり、ゲームが快適になります。

影の描画には、いくつか方法がありますが、
DxLibの公式に紹介されている「丸影」と「リアルシャドウ」を紹介します。

【丸影】



丸影の実装

https://dxlib.xsrv.jp/program/dxprogram_3DAction_CollObj.html

(Player_ShadowRender関数)

【リアルシャドウ】



リアルシャドウの実装

https://dxlib.xsrv.jp/function/dxfunc_3d_shadow.html

DxLibのサイトには、他にも3Dのサンプルプログラムがありますので、
時間がある時に見ておきましょう。

■ 3Dのサンプルプログラム

https://dxlib.xsrv.jp/program/dxprogram_3D.html

丸影の方が、実装方法が簡単ですので、
今回は、丸影を描画していきたいと思います。

```
ResourceManager.h
```

```
class ResourceManager  
{
```

```
public:
```

```
    // リソース名  
    enum class SRC  
    {  
        ~ 省略 ~  
        PLAYER_SHADOW,  
    };
```

```
ResourceManager.cpp
```

```
void ResourceManager::Init(void)  
{
```

```
    ~ 省略 ~
```

```
    // プレイヤー影  
    res = new RES(RES_T::IMG, PATH_IMG + "Shadow.png");  
    resourcesMap_.emplace(SRC::PLAYER_SHADOW, res);
```

```
    ~ 省略 ~
```

```
}
```

```
CharactorBase.h
```

```
#pragma once
#include "../ActorBase.h"
class AnimationController;

class CharactorBase : public ActorBase
{

public:

    ~ 省略 ~

    // 描画
    virtual void Draw(void) override;

    // 解放
    virtual void Release(void) override;

protected:

    ~ 省略 ~

    // 丸影画像
    int imgShadow_;

    // リソースロード
    virtual void InitLoad(void) override;

    ~ 省略 ~

    // 丸影描画
    void DrawShadow(void);
```

```
CharactorBase.cpp
```

```
void CharactorBase::Draw(void)
{

    // 基底クラスの描画処理
```

```

ActorBase::Draw();

// 丸影の描画
DrawShadow();

}

void CharactorBase::InitLoad(void)
{
    // 丸影画像
    imgShadow_ = resMng_.Load(ResourceManager::SRC::PLAYER_SHADOW).handleId_;
}

void CharactorBase::DrawShadow(void)
{
    float PLAYER_SHADOW_HEIGHT = 800.0f;
    float PLAYER_SHADOW_SIZE = 30.0f;

    DxLibのサイトを見ながら実装すること
https://dxlib.xsrv.jp/program/dxprogram\_3DAction\_CollObj.html
    ( Player_ShadowRender関数 )
}

```

Player.cpp

```

void Player::InitLoad(void)
{
    // 基底クラスのリソースロード
    CharactorBase::InitLoad();

    // モデルのロード
    transform_.SetModel(
        resMng_.Load(ResourceManager::SRC::PLAYER).handleId_);
}

```

【要件】

Playerの丸影を描画させること。

【目標】

