２Ｄアクションゲーム

角田研究室

奥﨑　弥

Unity概要

Unityを使っているゲーム


ヒューマン フォール フラット
HUMAN
Fall Flat
新ステージ「サーマル」追加!


白猫
プロジェクト
――まだ見ぬ世界へ――


DRAGON QUEST
VIII
ドラゴンクエストVIII
空と海と大地と呪われし姫君

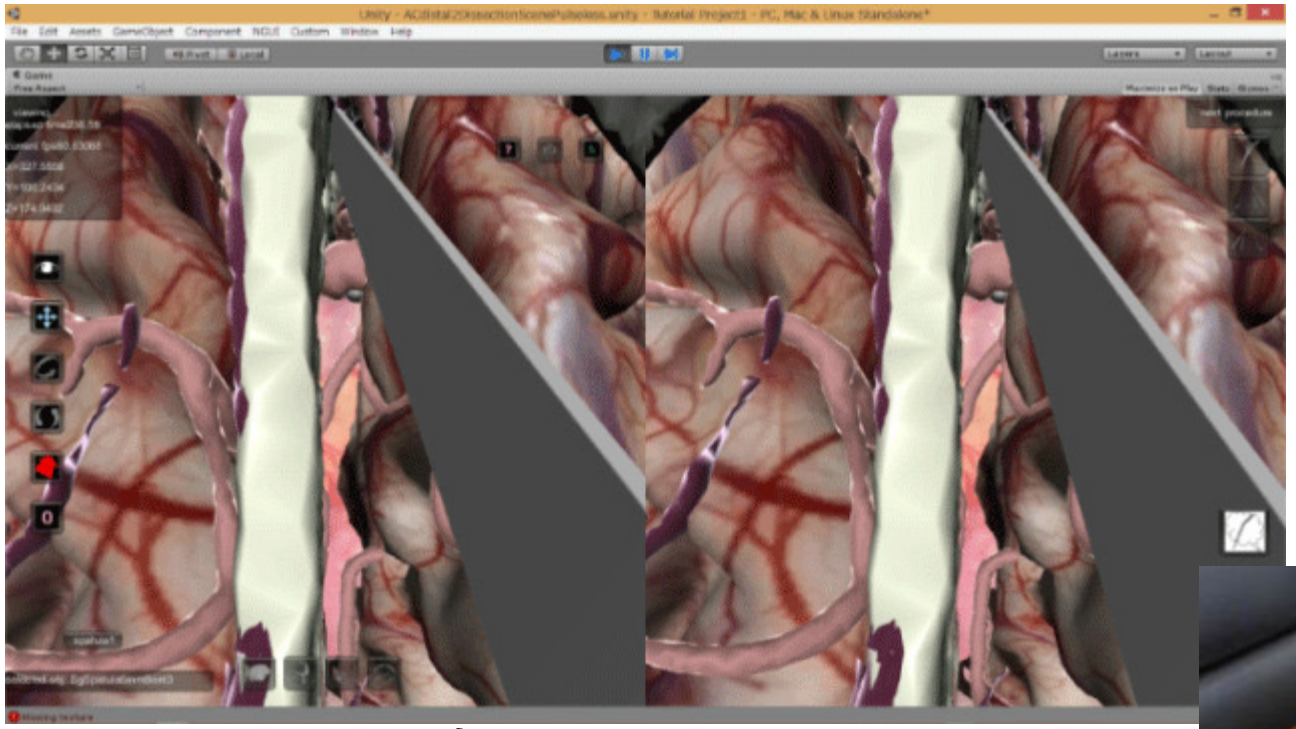
活用事例

・メタバース/VRサービス：
世界最大のソーシャルメタバースVRChat

・医療業界：東京大学のVRでの外科手術のシミュレーション

・建設業界：BIMデータを活用した3Dモデルの構築

メタバース/VRサービス



医療業界

建設業界

Unreal Engine（アンリアル・エンジン）

- ゲームエンジントップクラスのリアルなグラフィック表現

比較

モバイルARゲームや2Dゲームやモバイル・スマホゲームアプリ

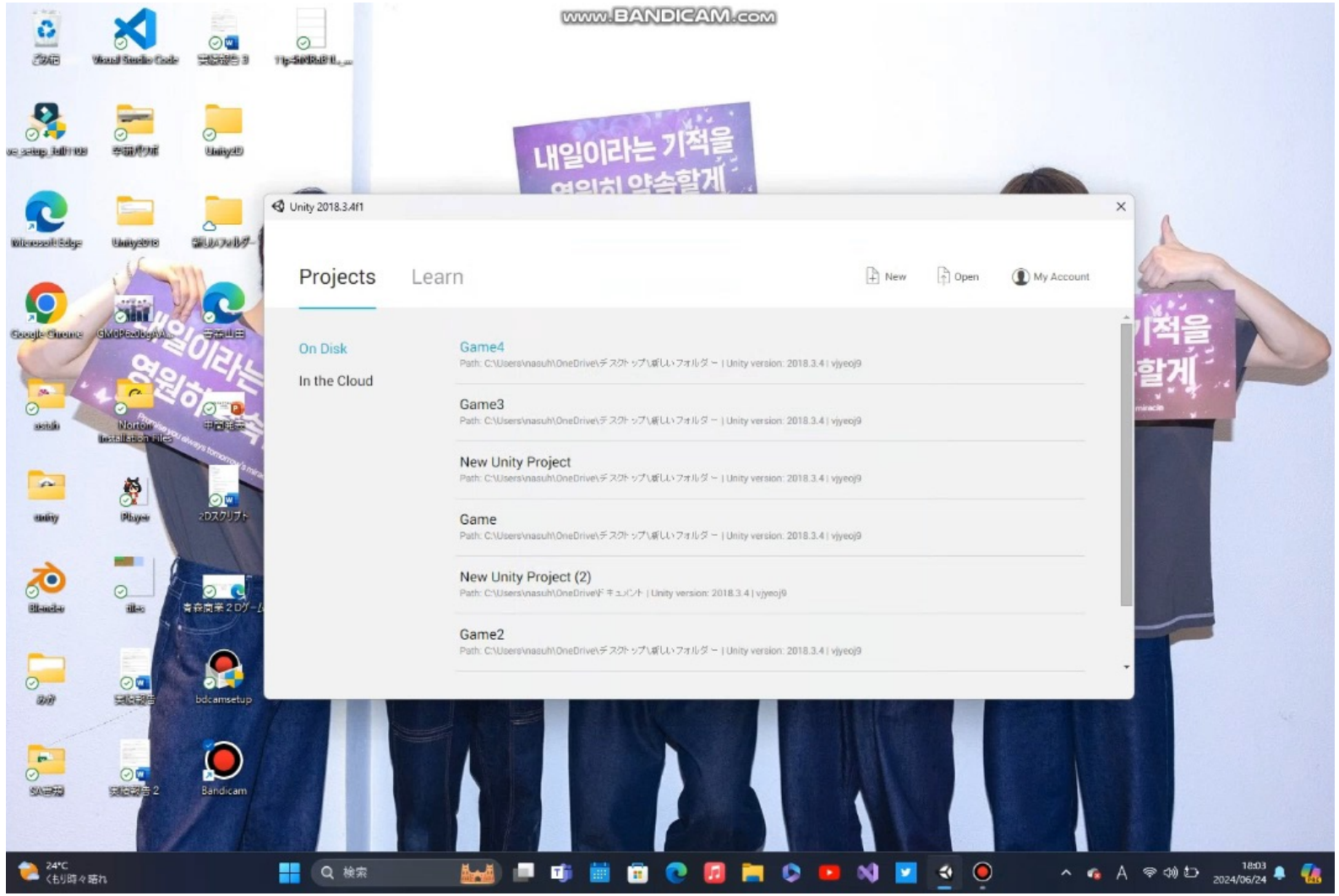
→「Unity」

PCやコンシューマ向けゲーム

→「Unreal Engine」

ゲーム作成

マップの作成



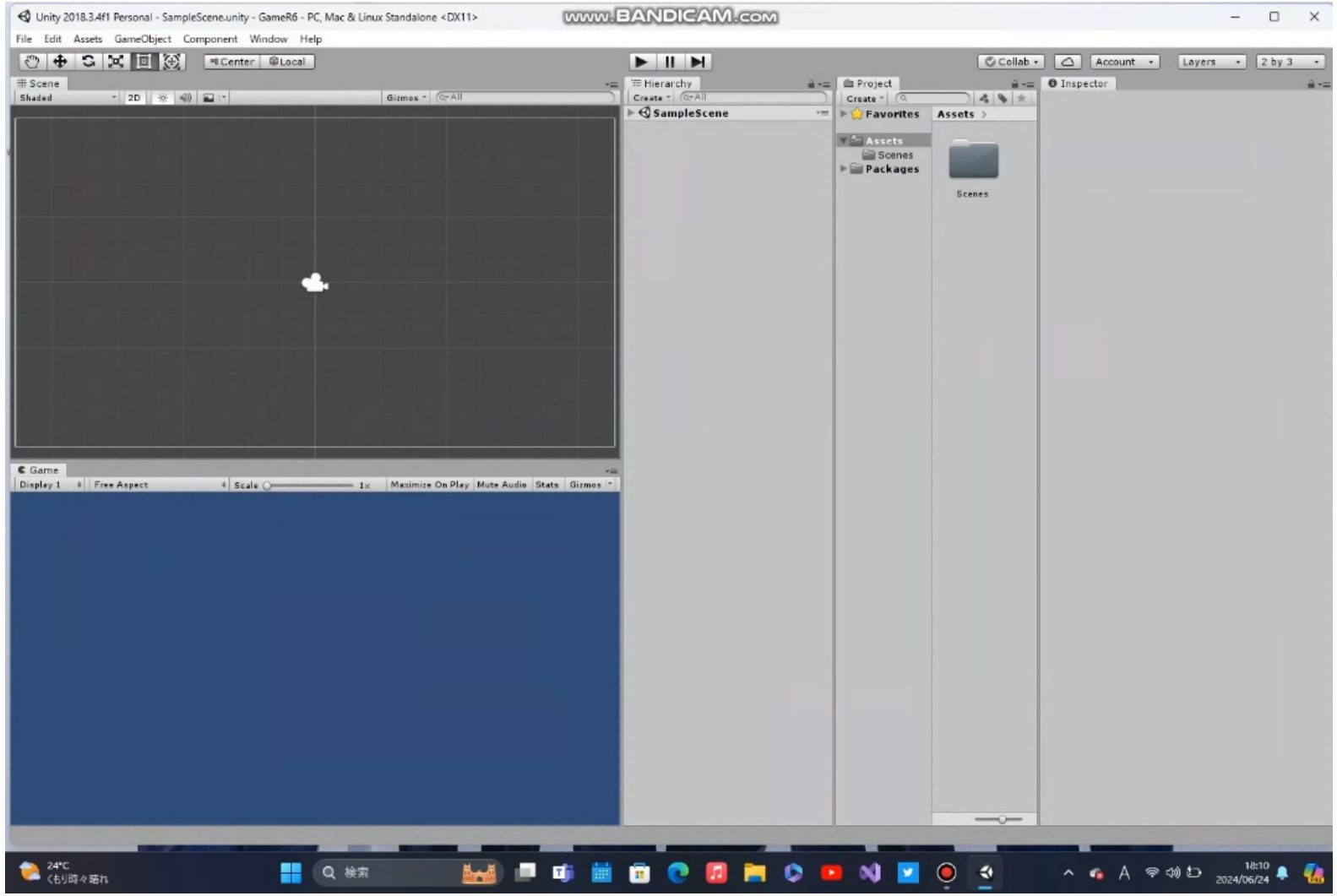
①Newクリック

②Project name
「GameR6」

③Location（保存場所）
「Local Disk」
Cドライブ
→Unityフォルダ

④Template
「２D」

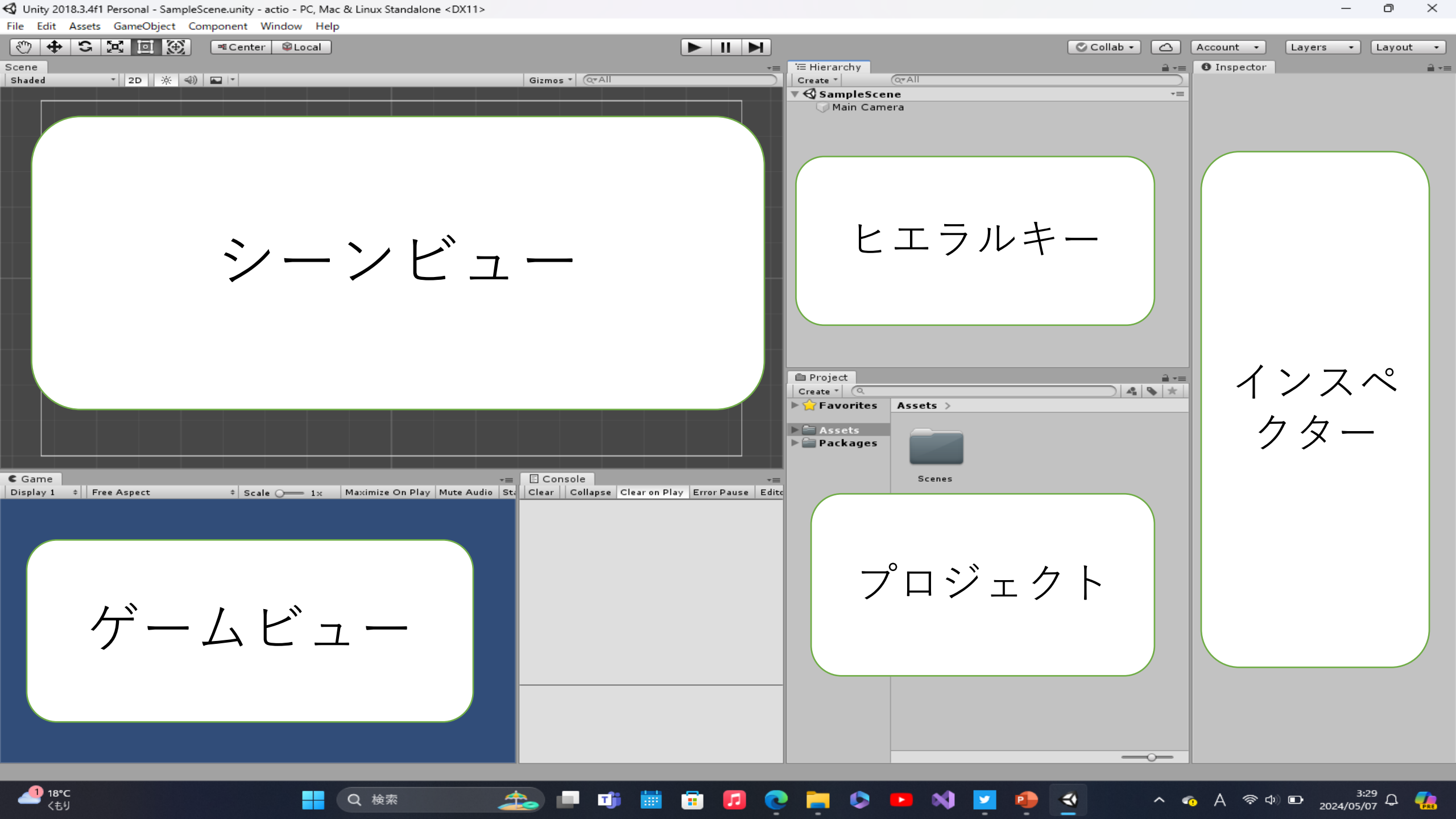
⑤Create Project

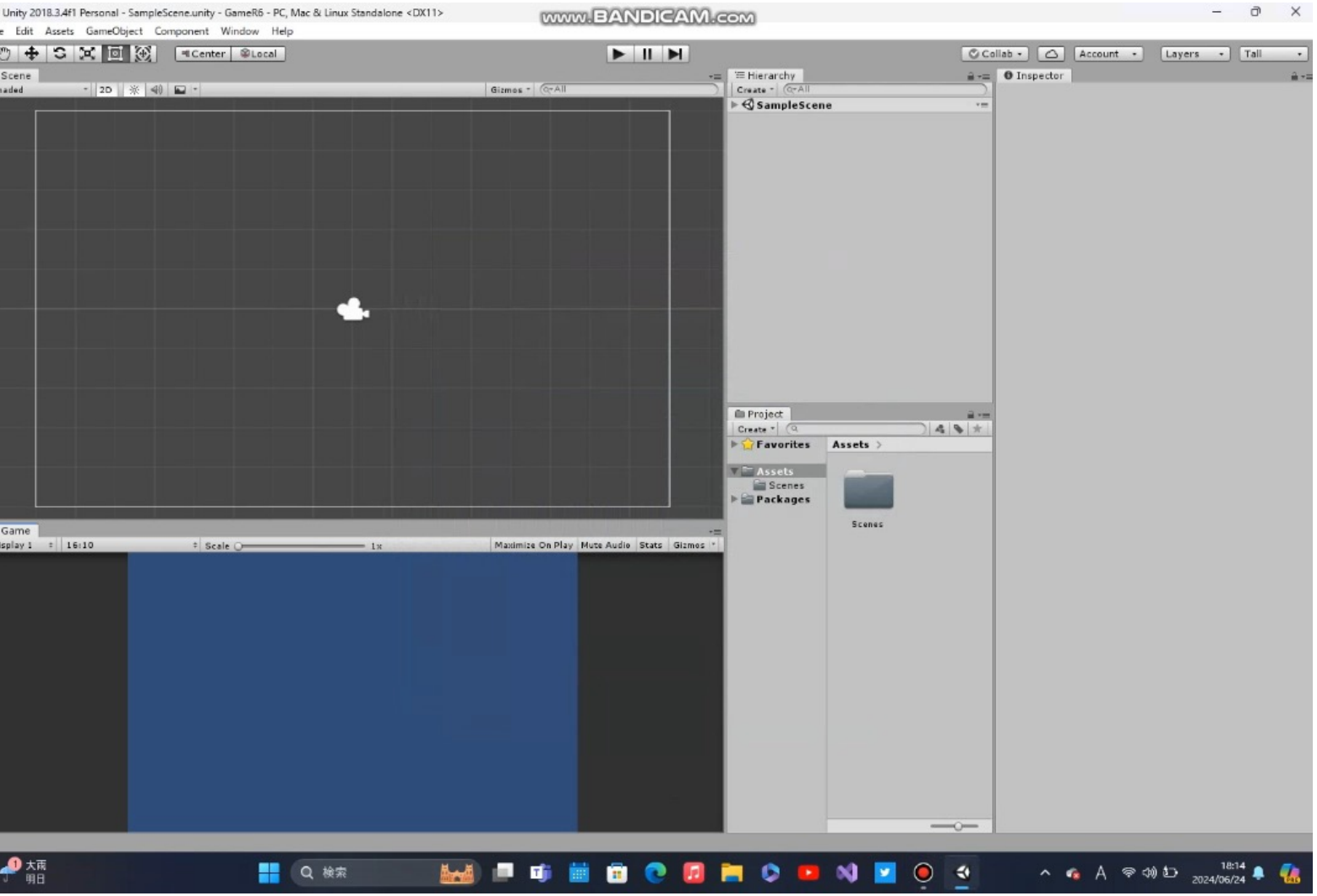


①Layout
「Tall」

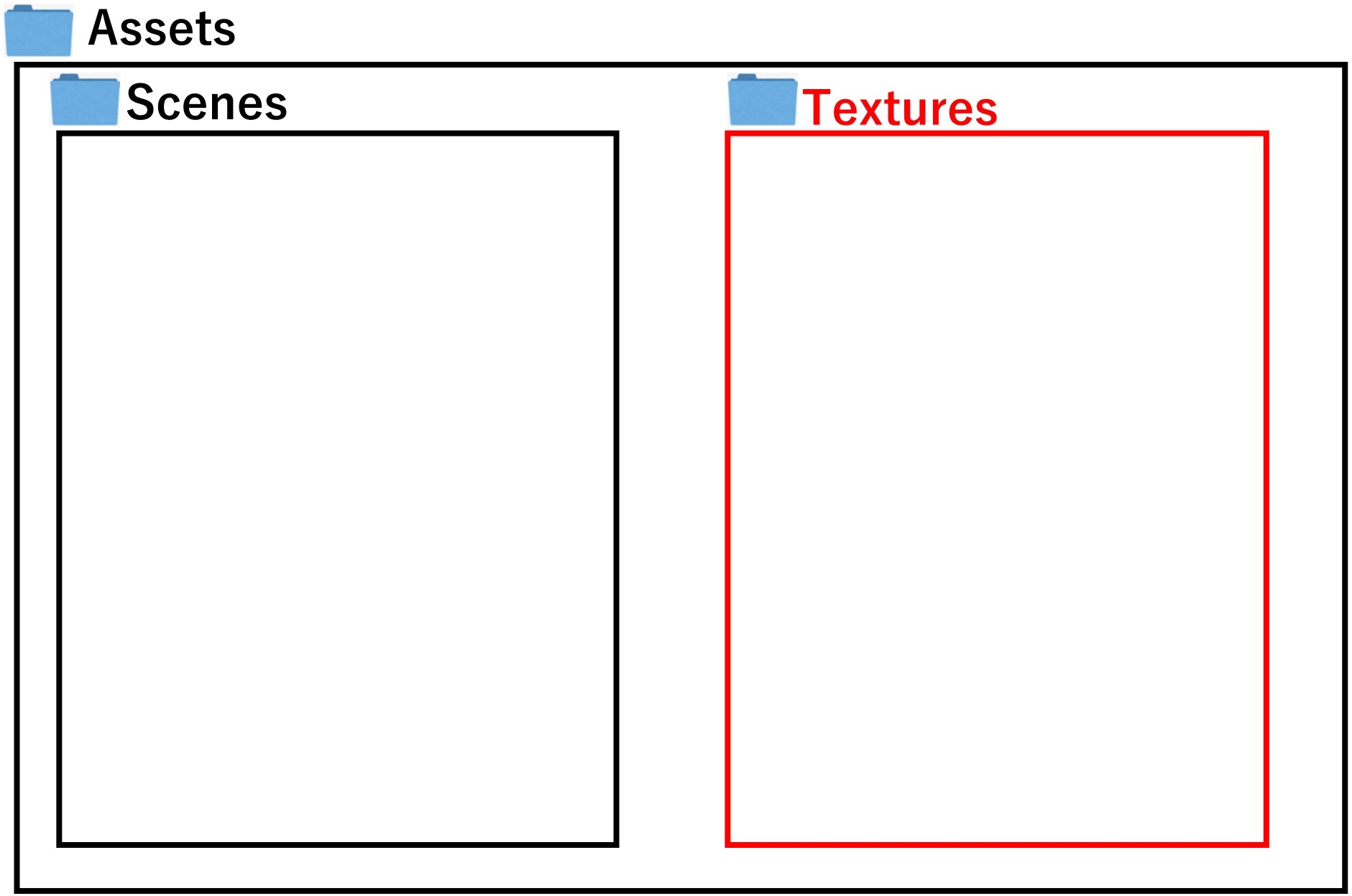
②Gameビューを下に

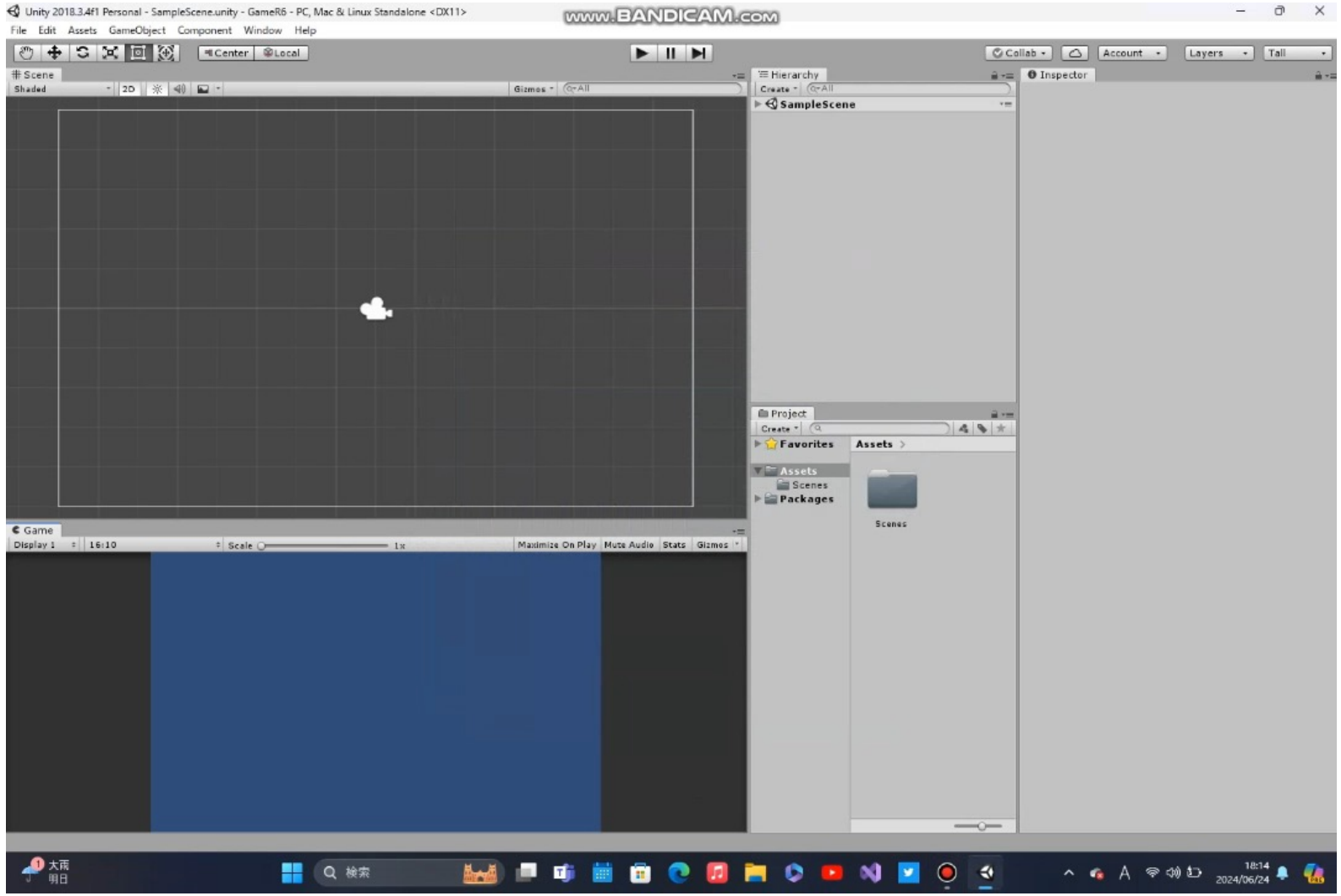
③Free Asepect
「16：10」


Unity 2018.3.4f1 Personal - SampleScene.unity - actio - PC, Mac & Linux Standalone <DX11>
File Edit Assets GameObject Component Window Help
Center Local
Collab Account Layers Layout
Scene
Shaded 2D Gizmos All
シーンビュー
Hierarchy
Create All
SampleScene
Main Camera
ヒエラルキー
Inspector
インスペクター
Project
Create
Favorites
Assets
Packages
Assets
Scenes
プロジェクト
Game
Display 1 Free Aspect Scale 1x Maximize On Play Mute Audio
ゲームビュー
Console
Clear Collapse Clear on Play Error Pause
18°C くもり
検索
3:29
2024/05/07



【Project】
①texturesフォルダ作成
↓
・Assetsクリック
→Create→folder
名前：Textures


Assets
Scenes
Textures



【Project】

②Texturesフォルダの中にTilesフォルダ作成

↓

・Texturesクリック

→Create→folder

名前：Tiles

③Tilesフォルダに画像をドラッグアンドドロップ

Assets

Scenes

Textures

Tiles

Assets

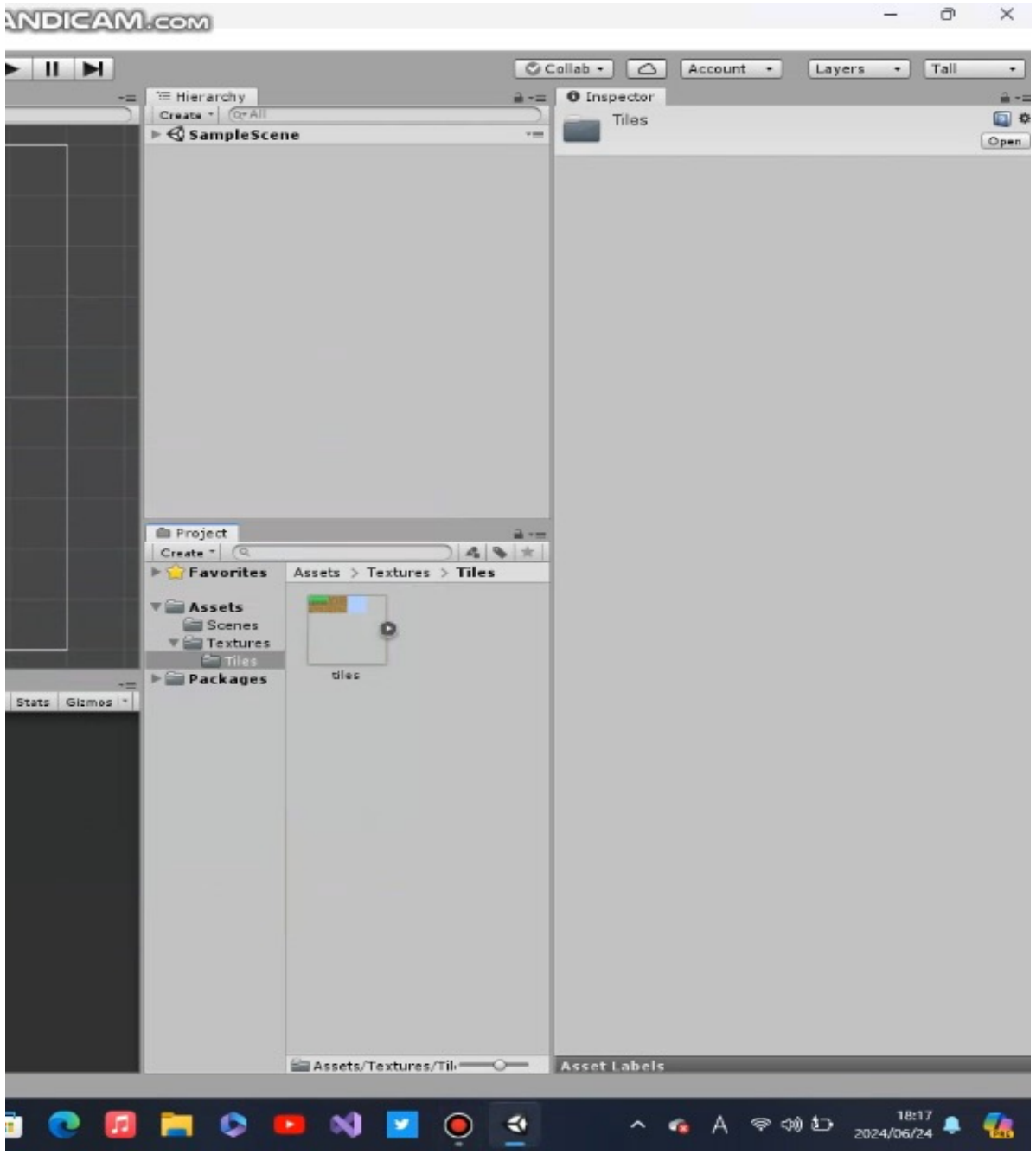
Scenes

Textures

Tiles

Player

テクスチャの設定



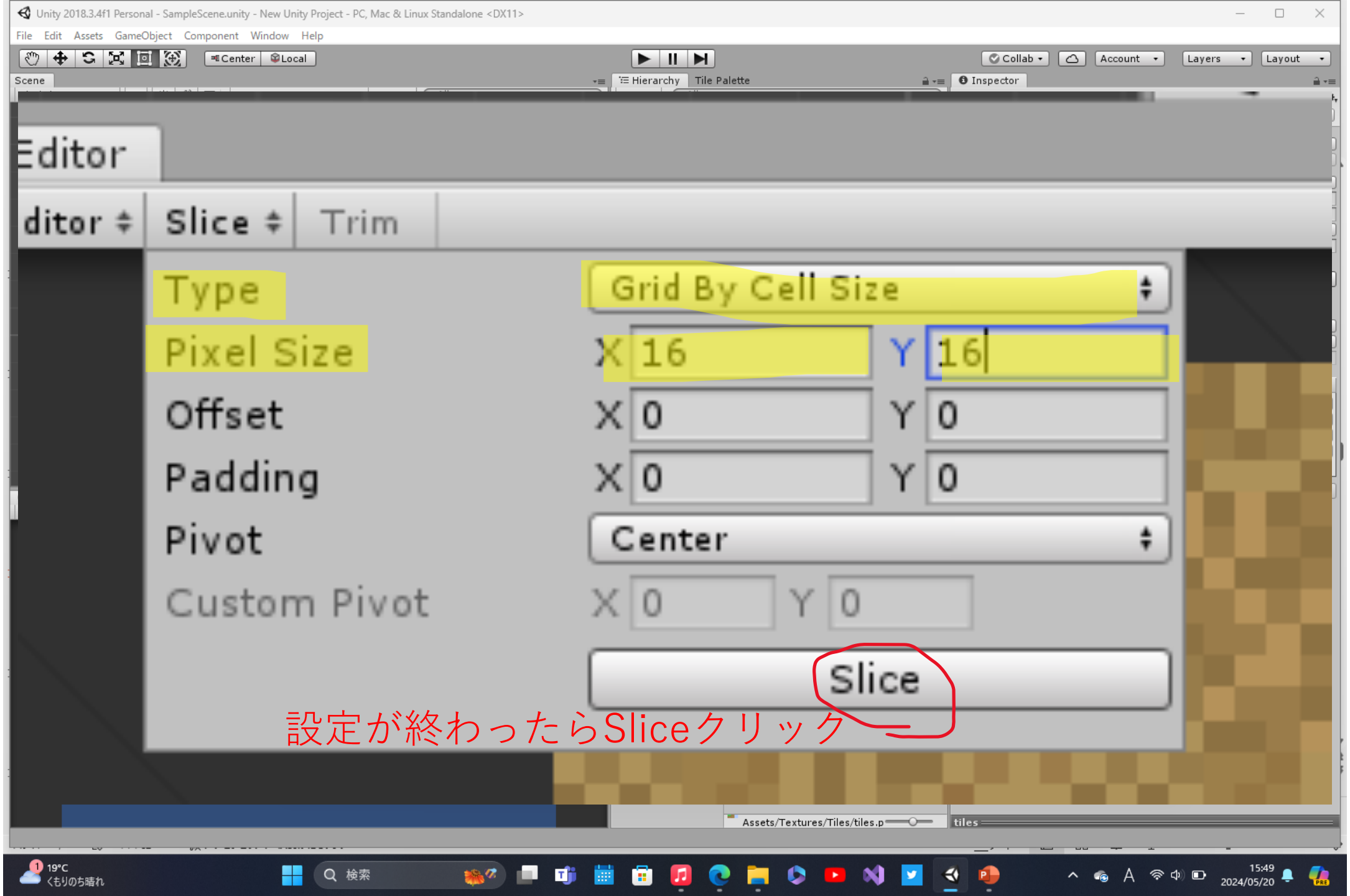
・１枚の画像から３枚の画像を切り取りする

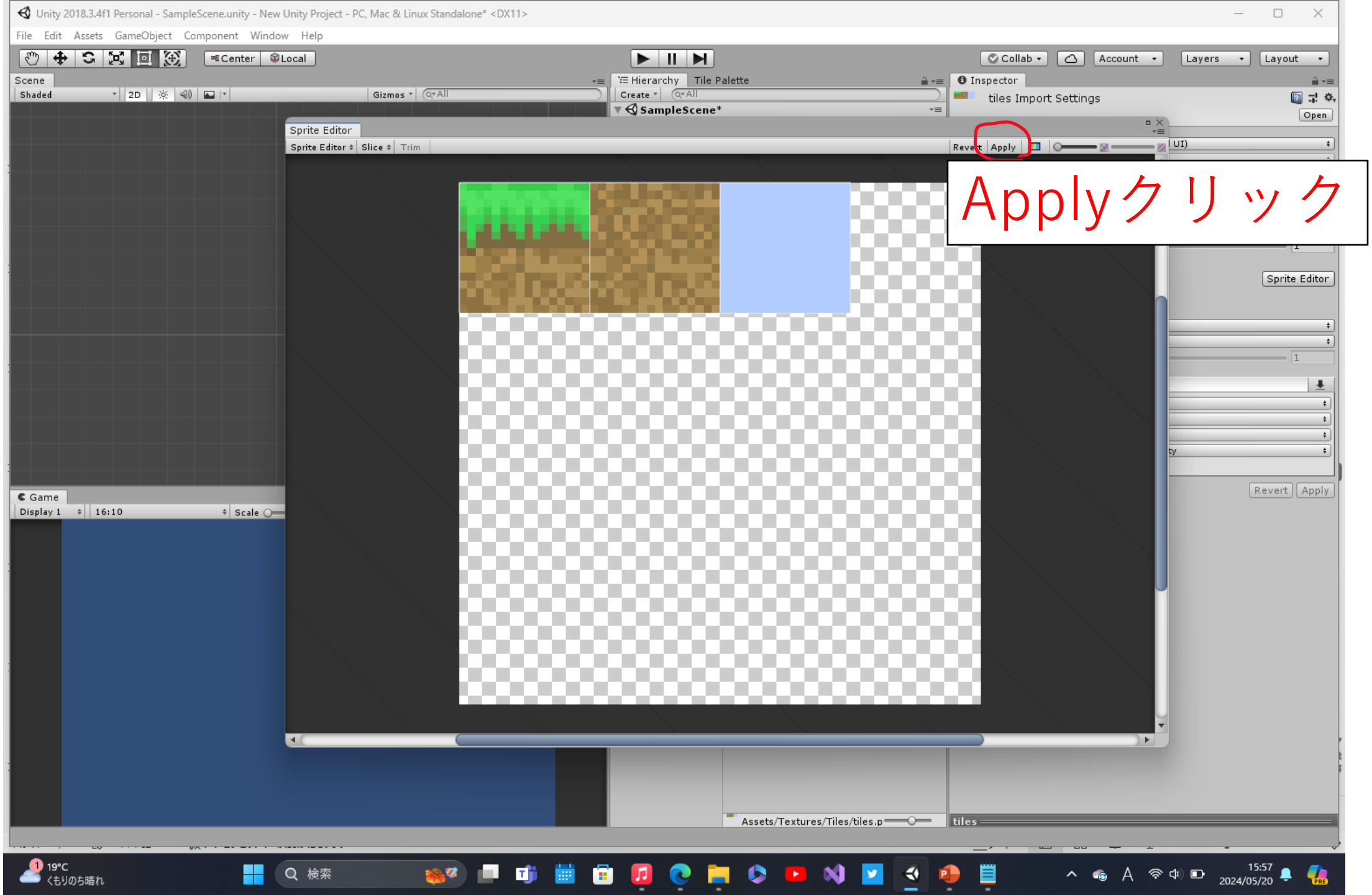
①Sprite Mode→Multiple(複数)
　：１つの画像から複数の
　　スプライトを作成

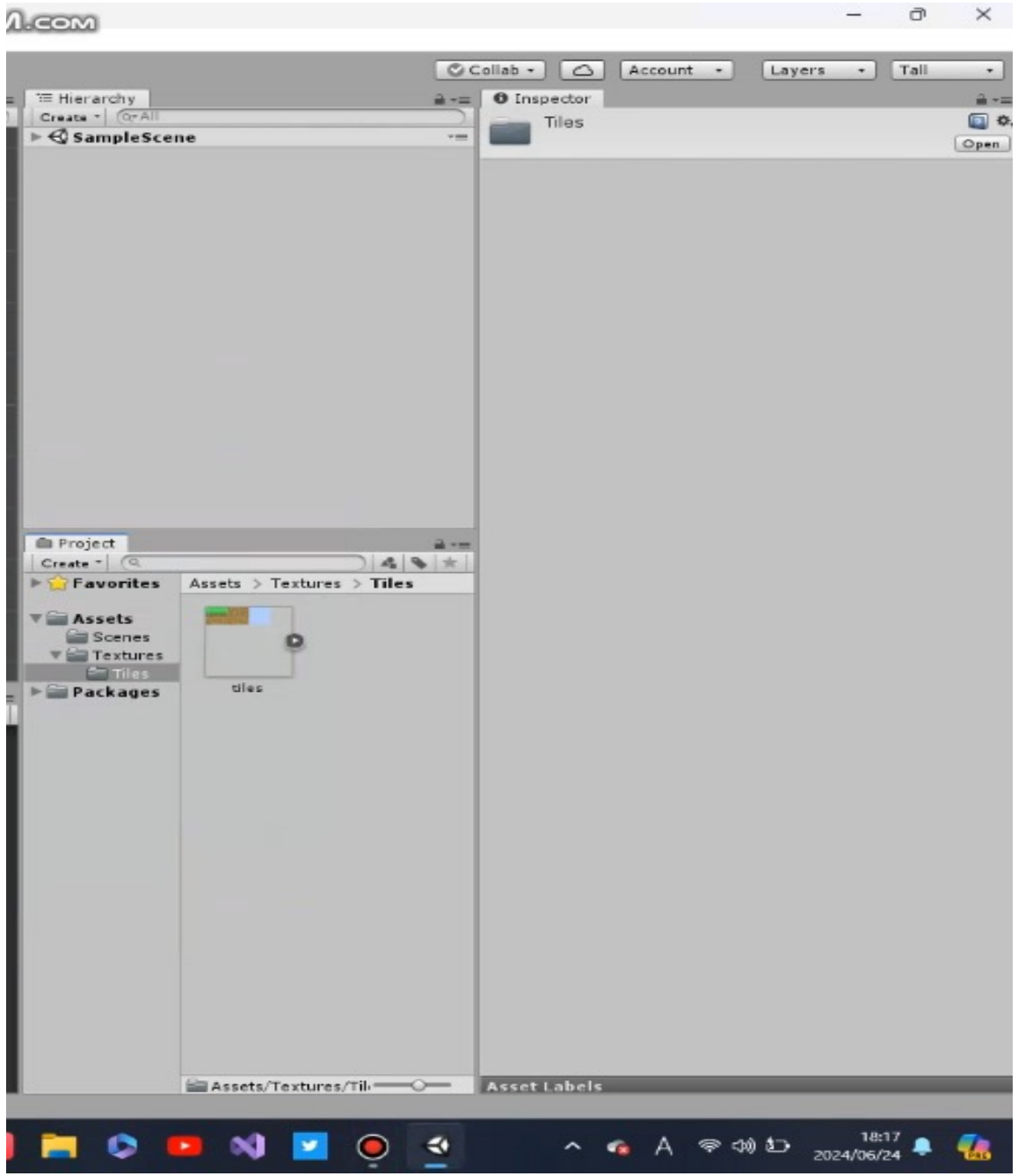
②Pixels Per Unit→16
　：スプライトのサイズ調整

③Sprite Editorクリック

④Applyクリック


Unity 2018.3.4f1 Personal - SampleScene.unity - New Unity Project - PC, Mac & Linux Standalone <DX11>
File Edit Assets GameObject Component Window Help
Center Local
Collab Account Layers Layout
Scene
Hierarchy Tile Palette
Inspector
Editor
ditor Slice Trim
Type
Grid By Cell Size
Pixel Size
X 16 Y 16
Offset
X 0 Y 0
Padding
X 0 Y 0
Pivot
Center
Custom Pivot
X 0 Y 0
Slice
設定が終わったらSliceクリック
Assets/Textures/Tiles/tiles.p
tiles
19°C
くもりのち晴れ
検索
15:49
2024/05/20

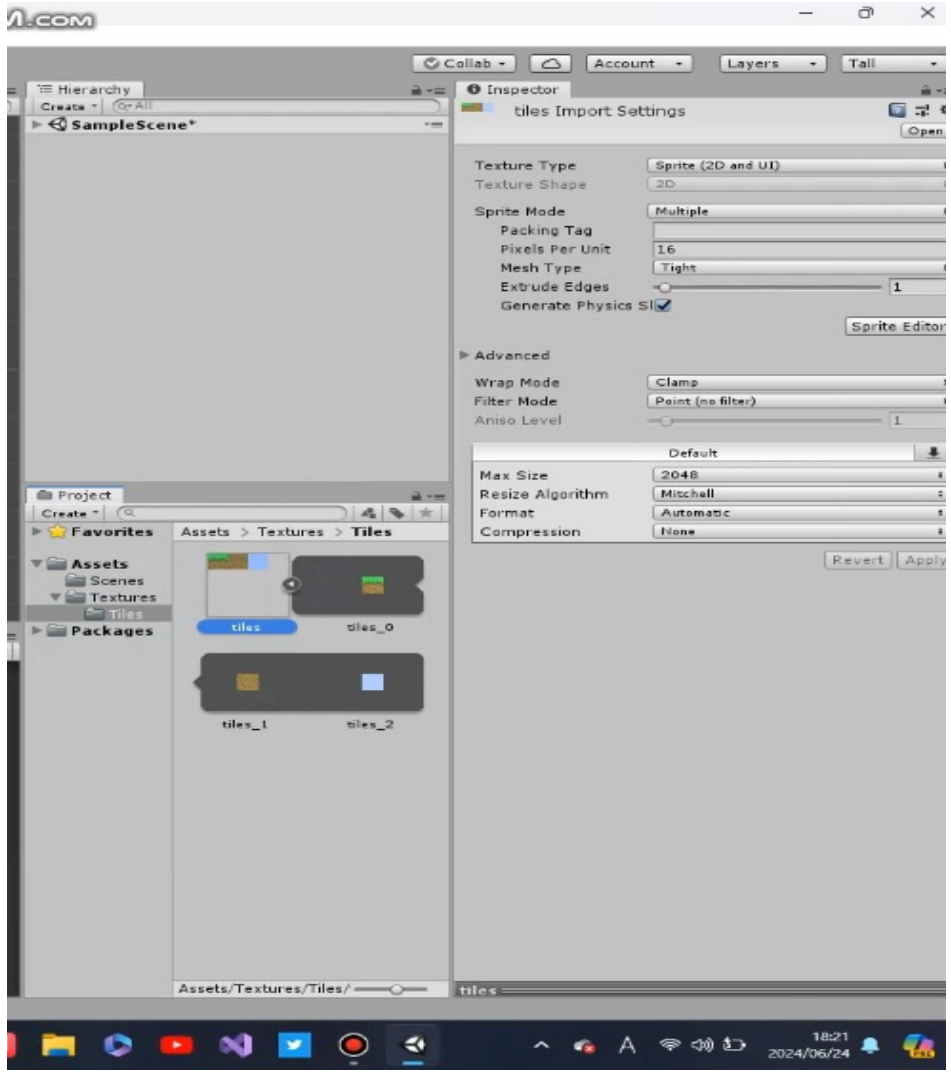

Unity 2018.3.4f1 Personal - SampleScene.unity - New Unity Project - PC, Mac & Linux Standalone* <DX11>
File Edit Assets GameObject Component Window Help
Center Local
Collab Account Layers Layout
Scene
Shaded 2D Gizmos
Hierarchy Tile Palette
Create
SampleScene*
Inspector
tiles Import Settings
Open
Sprite Editor
Sprite Editor Slice Trim
Revert Apply
Applyクリック
Game
Display 1 16:10 Scale
Assets/Textures/Tiles/tiles.p
tiles
Revert Apply
19°C くもりのち晴れ
検索
15:57
2024/05/20



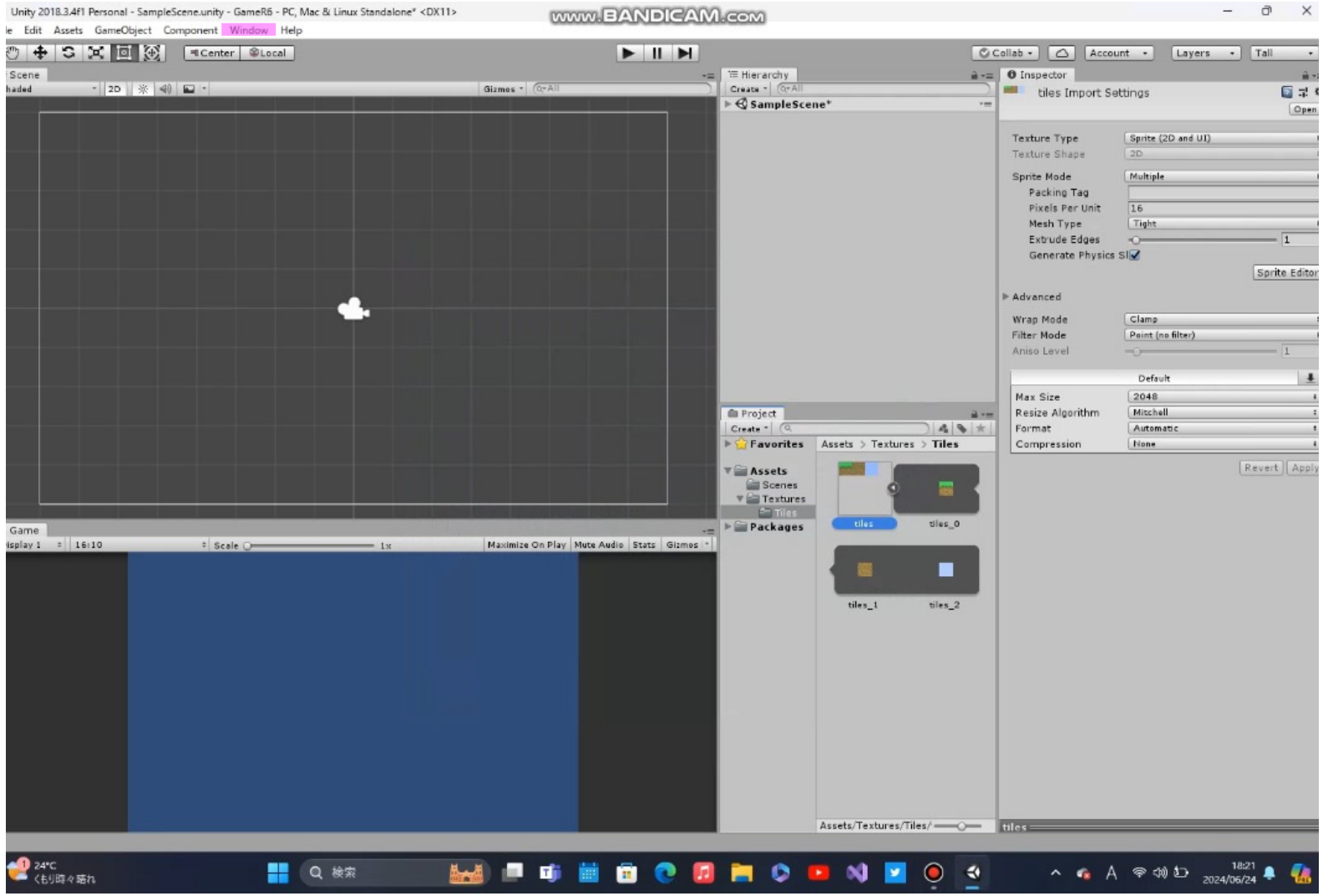
画質が悪い状態を直す
①Filter Mode→point（no filter）

②compression→None

③Applyクリック



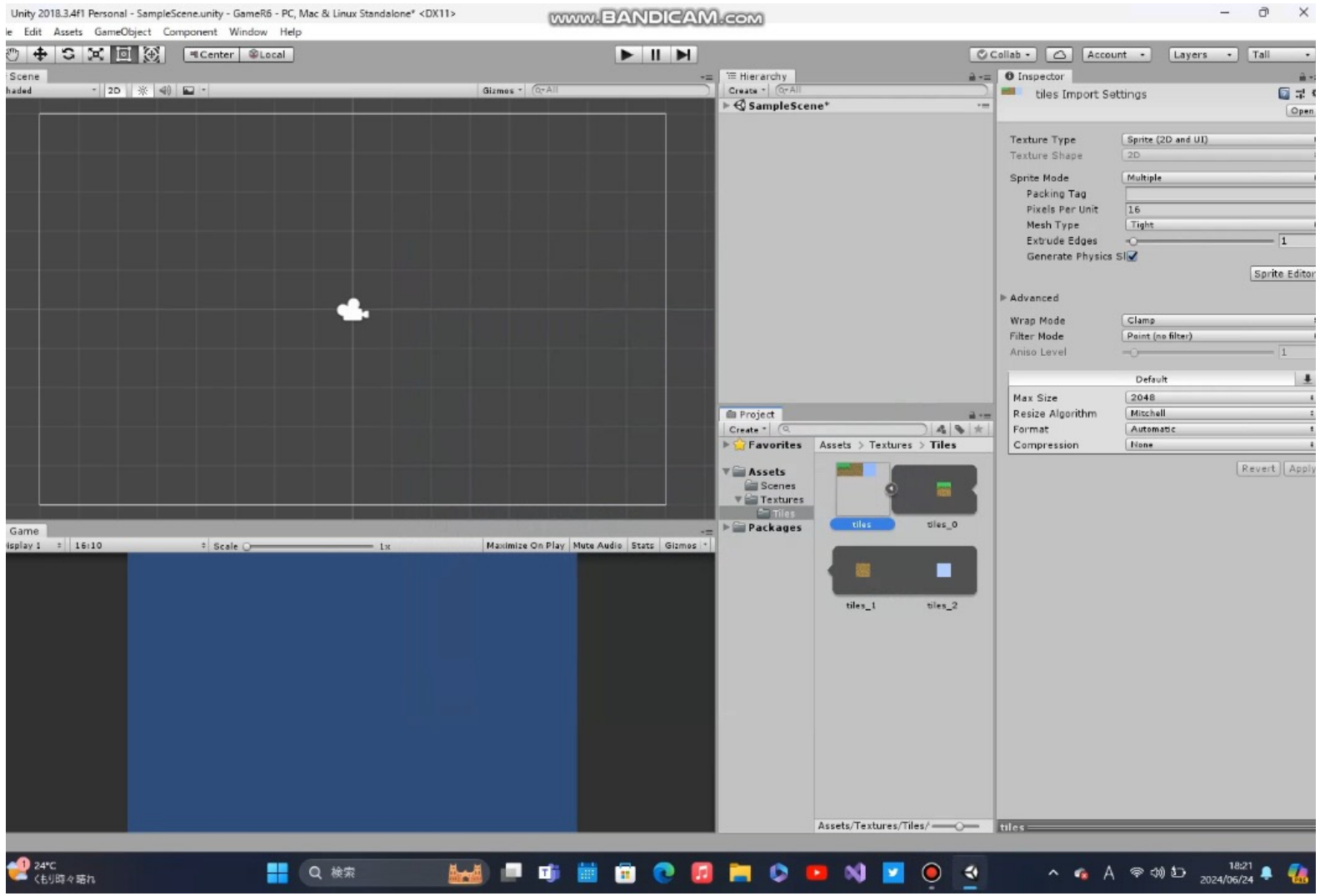
【Hierarchy】
Create→2DObject→Tilemap
名前：Terrain（地形）



タイルパレットの
ウィンドウを開く

・Window→２D
→Tile Parette

タイルパレットをインスペクターの横にドラッグ



【Tile Palette】

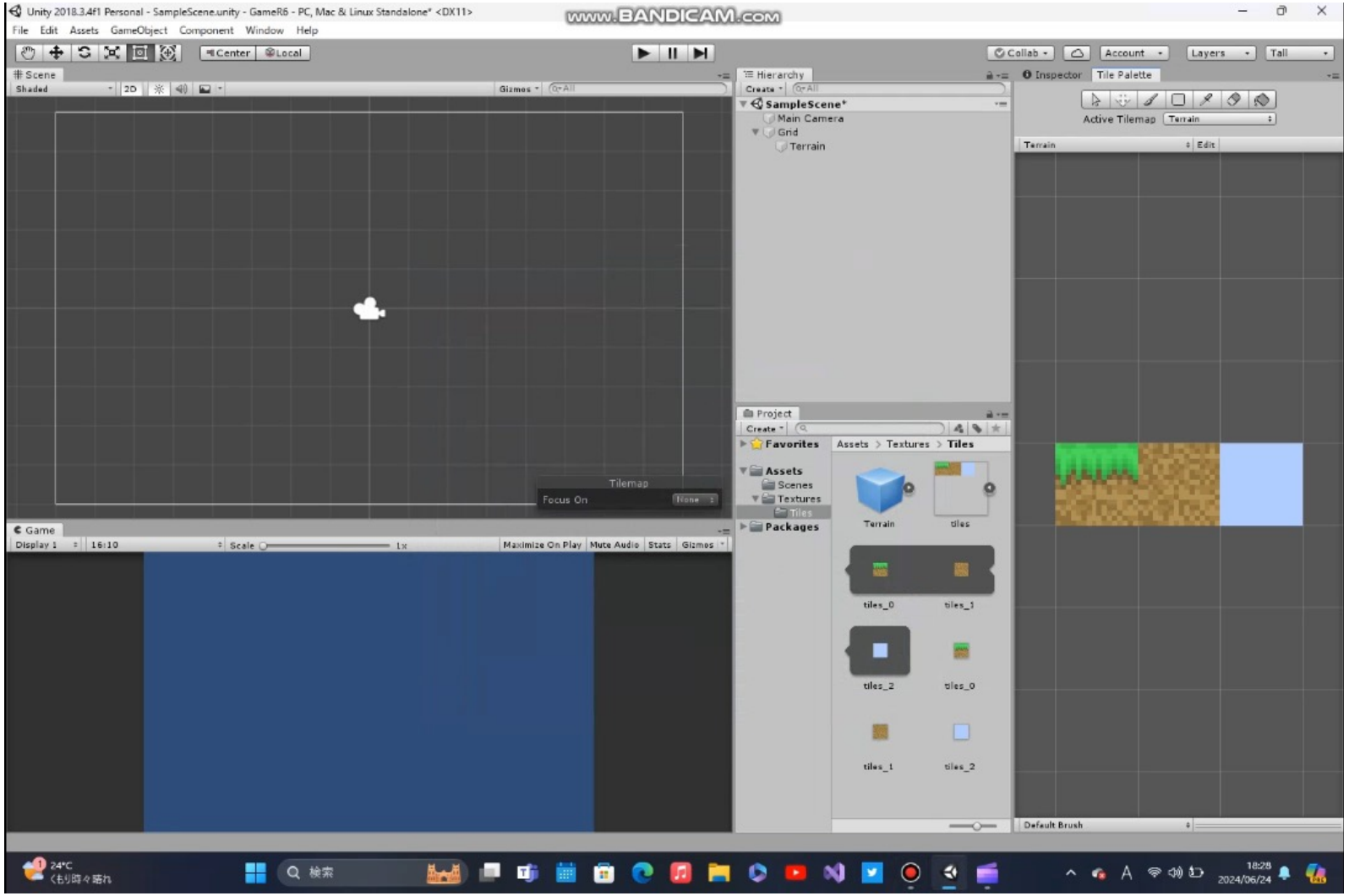
①Name→Terrain

②Create

③フォルダの選択

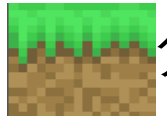
④画像をドラッグアンドドロップ

⑤フォルダの選択

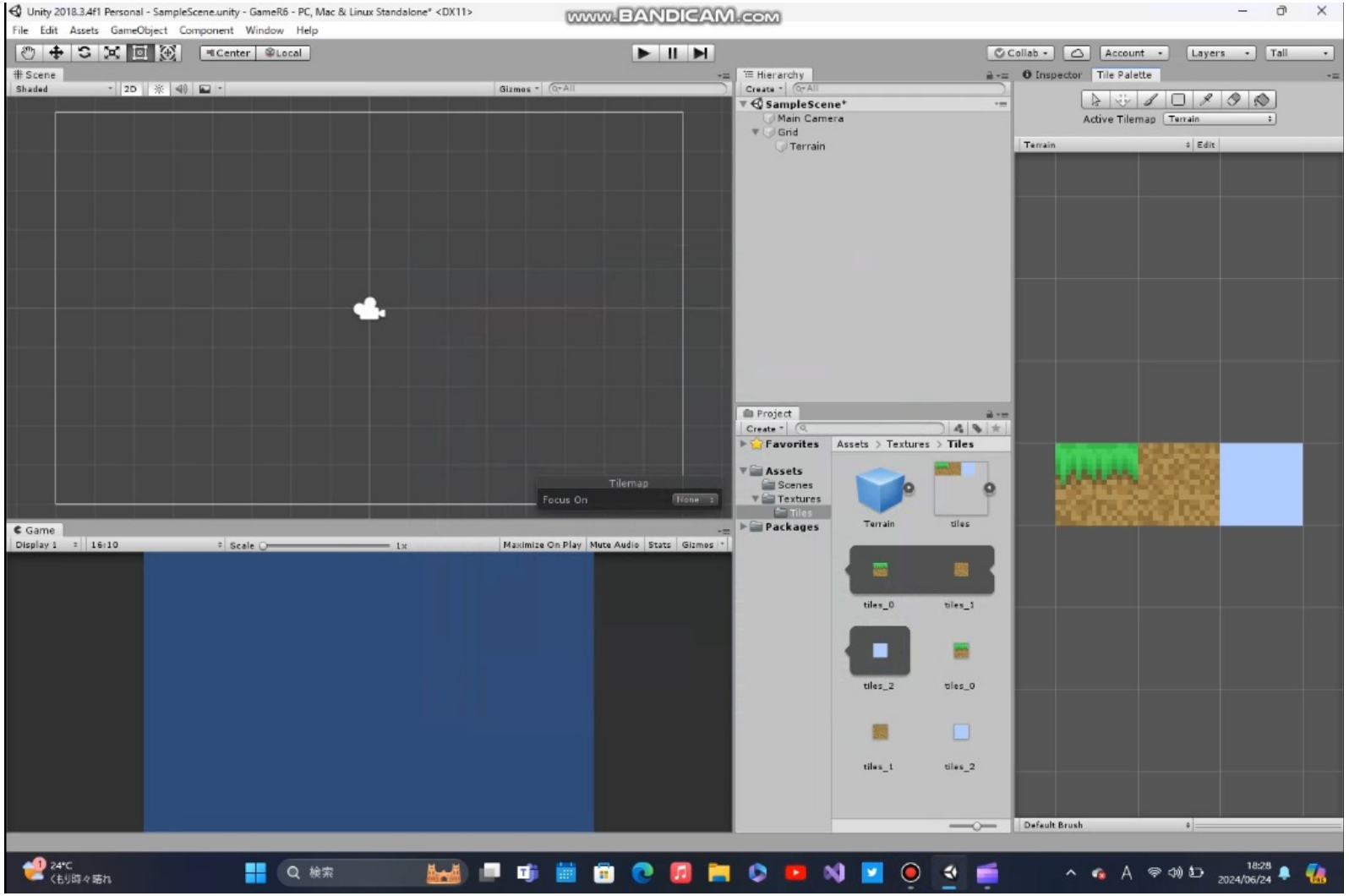


①Terrainクリック

②✏️クリック
（間違えたら消しゴムで消せる）

③  クリックして書く

 も同様に

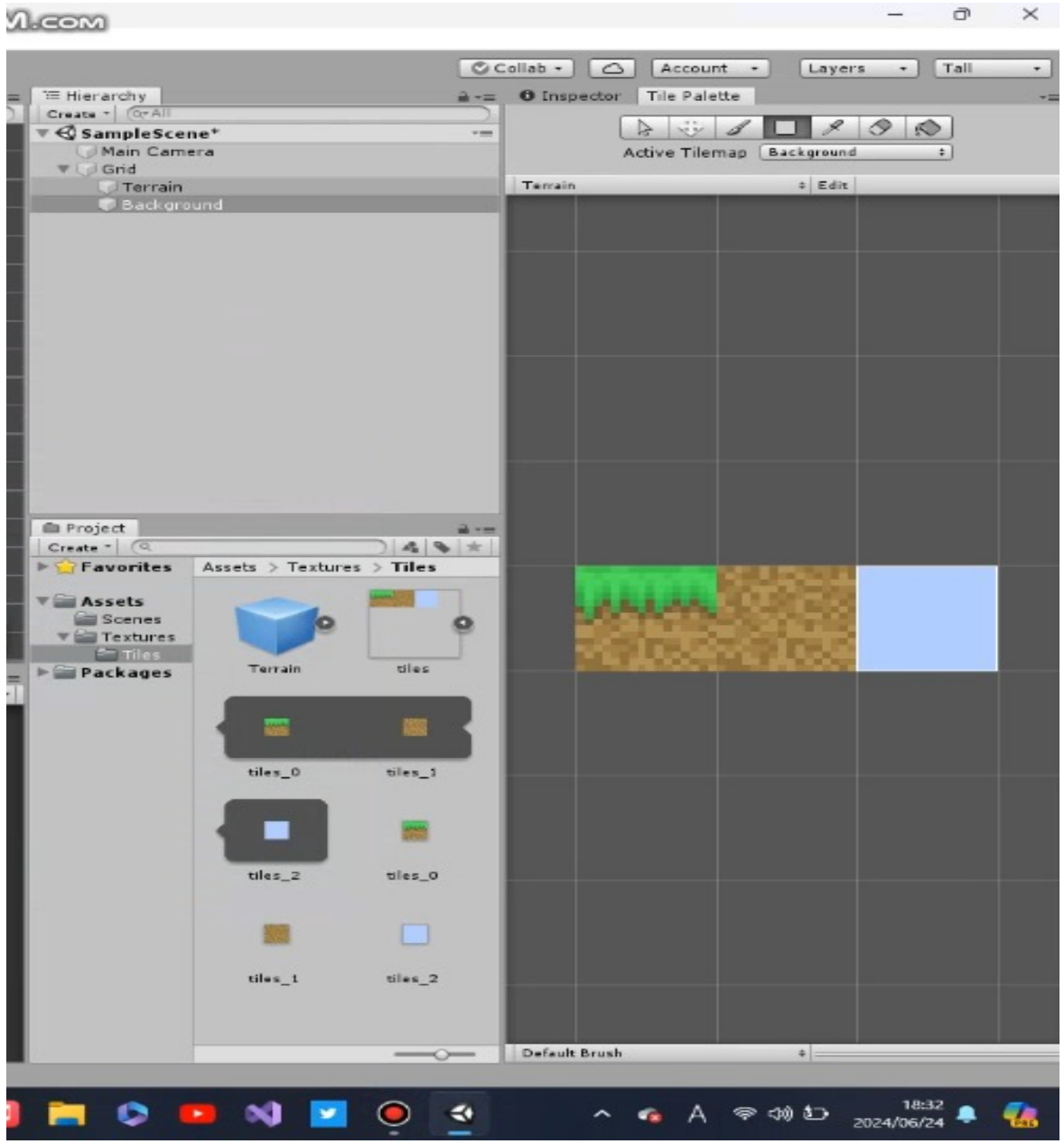


【Hierarchy】
・Gridクリック
→Create
→2DObject→Tilemap

名前：Background

Backgroundを選択している状態から背景を書く

当たり判定

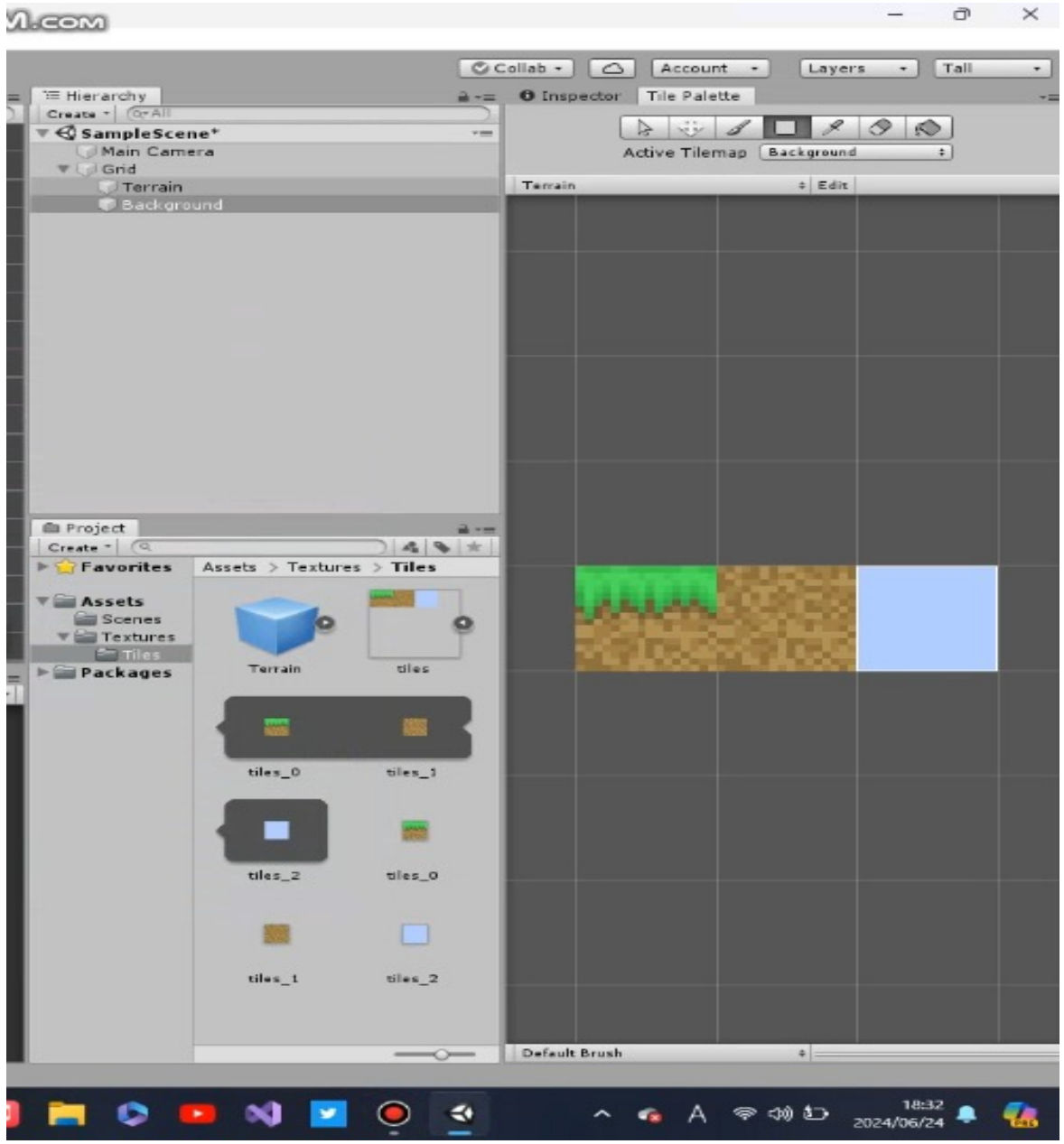


【Inspector】

Terrain選択→Add Componetクリック

- **Composite Collider 2D**
  （Rigidbody 2Dも追加される）

- **Tilemap Collider 2D**
  :タイルマップ用の当たり判定を作る



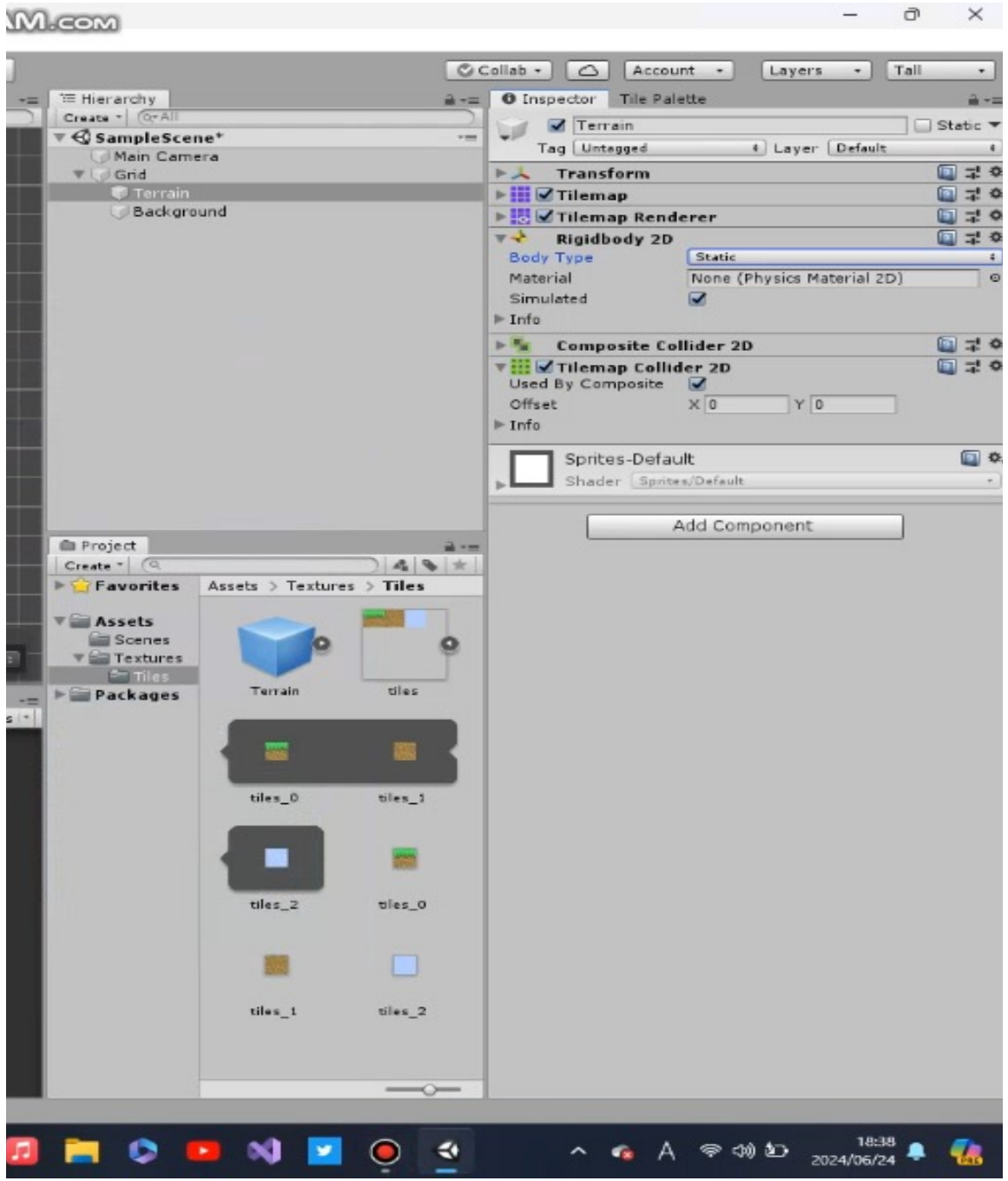
【Inspector】

Tilemap Collider 2D

・Used by Compositeチェック

Rigidbody 2D

・Body Type→Static



Playerが背景の後ろに隠れているのを修正する

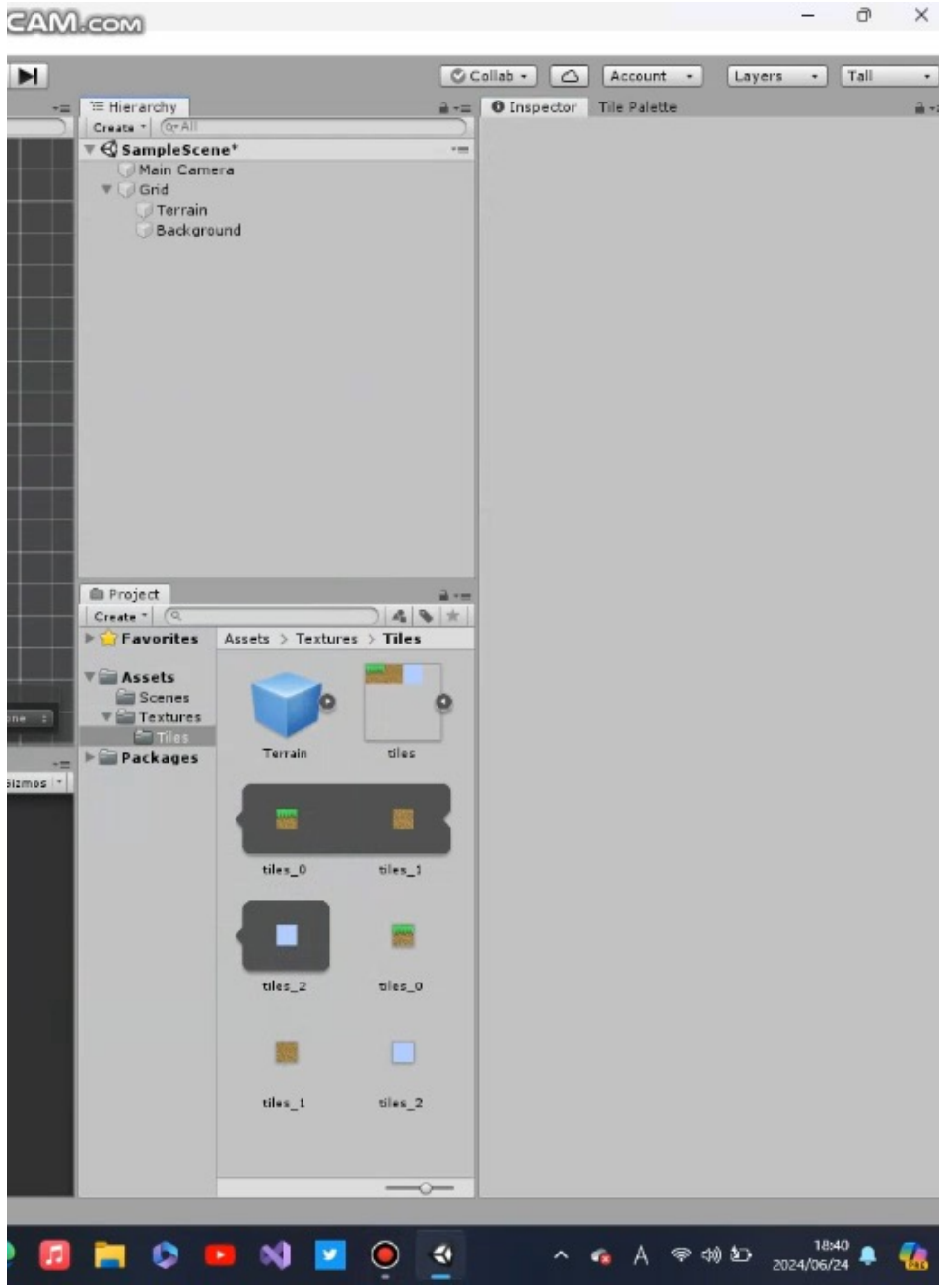
①Backgroundクリック

②【Tilemap Renderer】
Sorting Layer→Add Sorting Layer…

③「＋」→名前Background
　Defaultの上に持ってくる

④ Sorting Layer→ Background

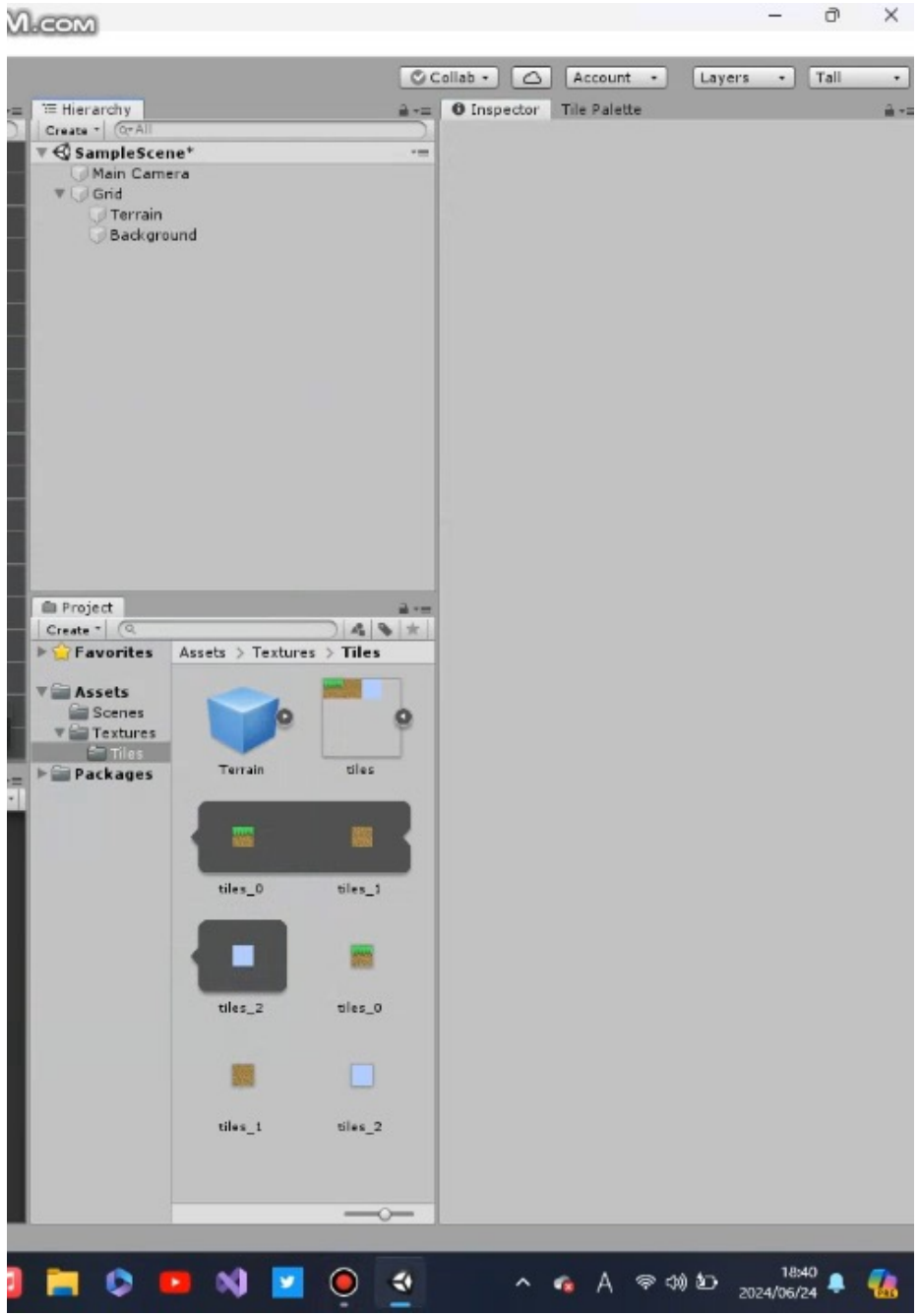
プレイヤーの作成



【Hierarchy】

Create→2DObject→Sprite

名前：Player



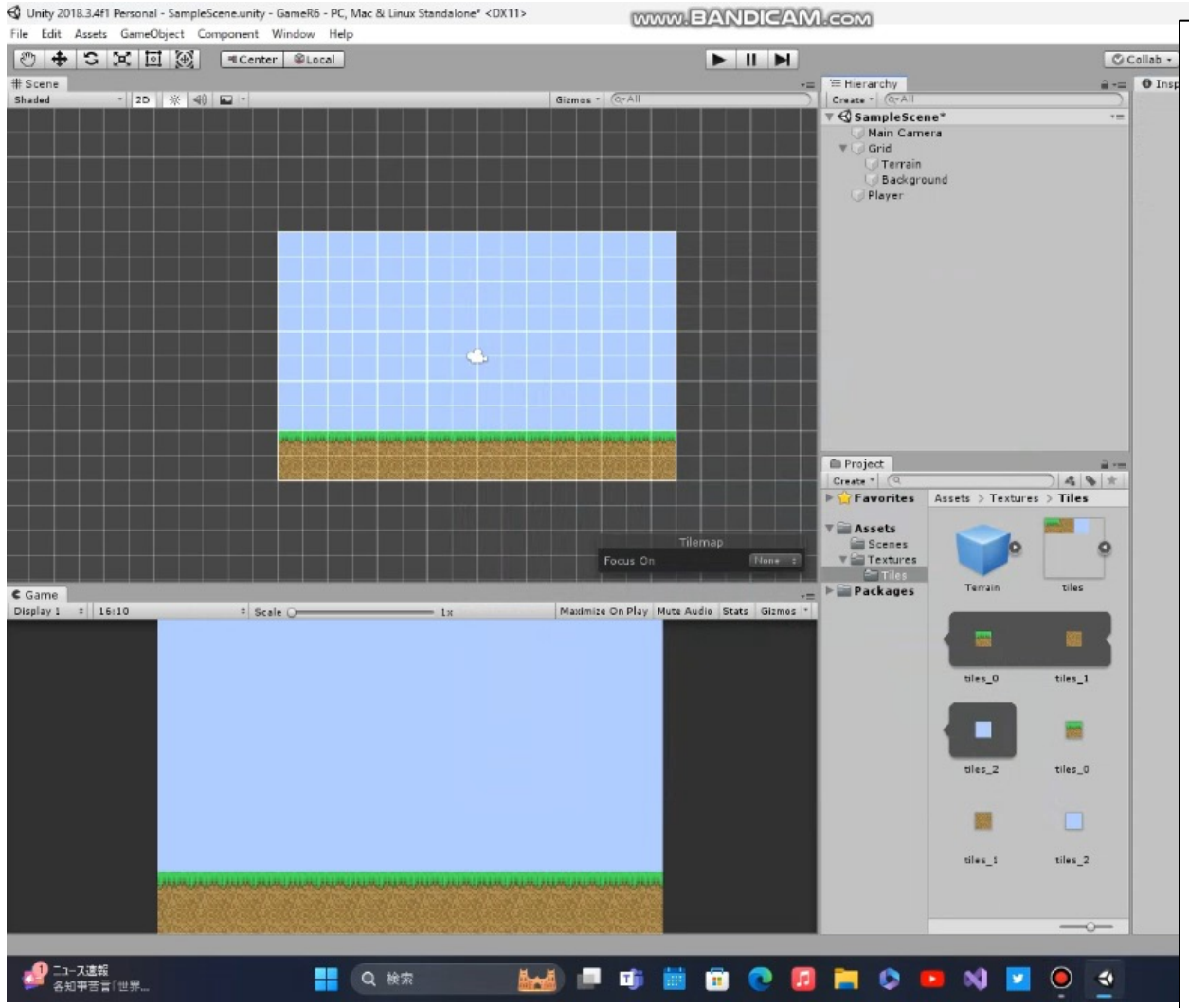
【Inspector】

Player選択→Add Componetクリック

・**Rigidbody 2D** 追加

・**Box Collider** 2D 追加

＊Size→ｘ １、 ｙ １



【Project】

Texturesフォルダの中に
Playerフォルダ作成

・Texturesをダブルクリック
Create→folder
名前：Player

Playerフォルダの中に画像を
ドラッグアンドドロップ

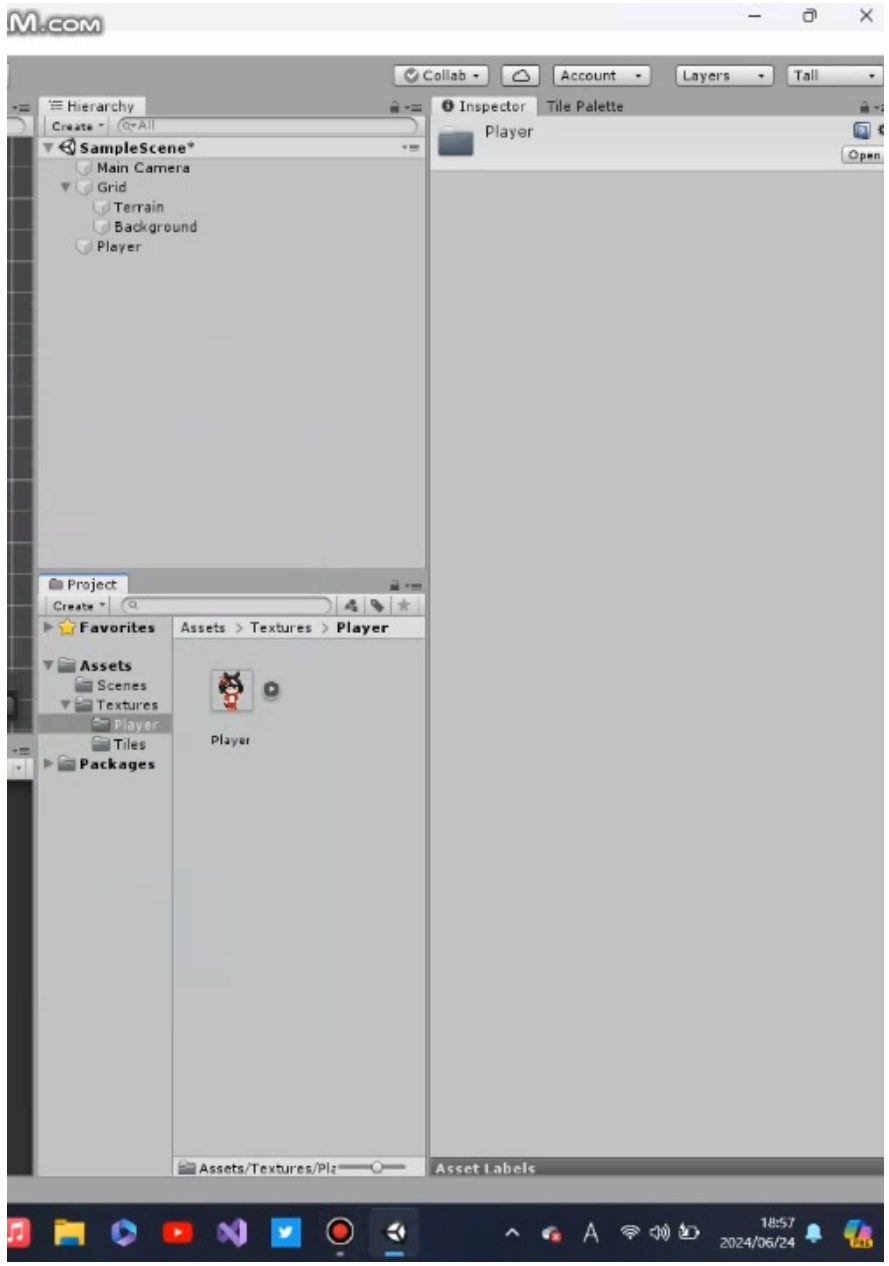
Assets

Scenes

Textures

Tiles

Player

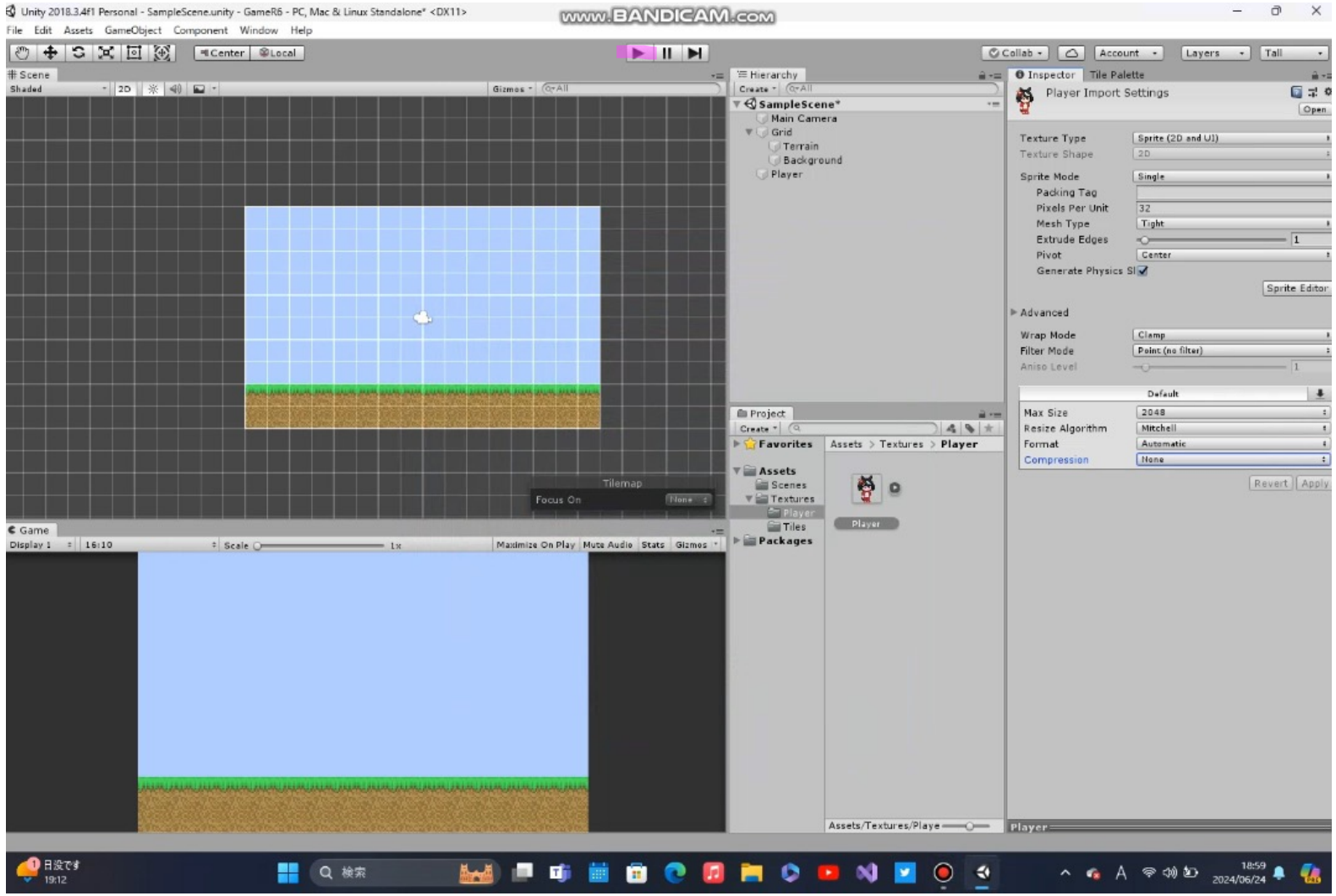


①Pixels Per Unit→32

②Filter Mode
→point（no filter)

③compression→None

④Applyクリック

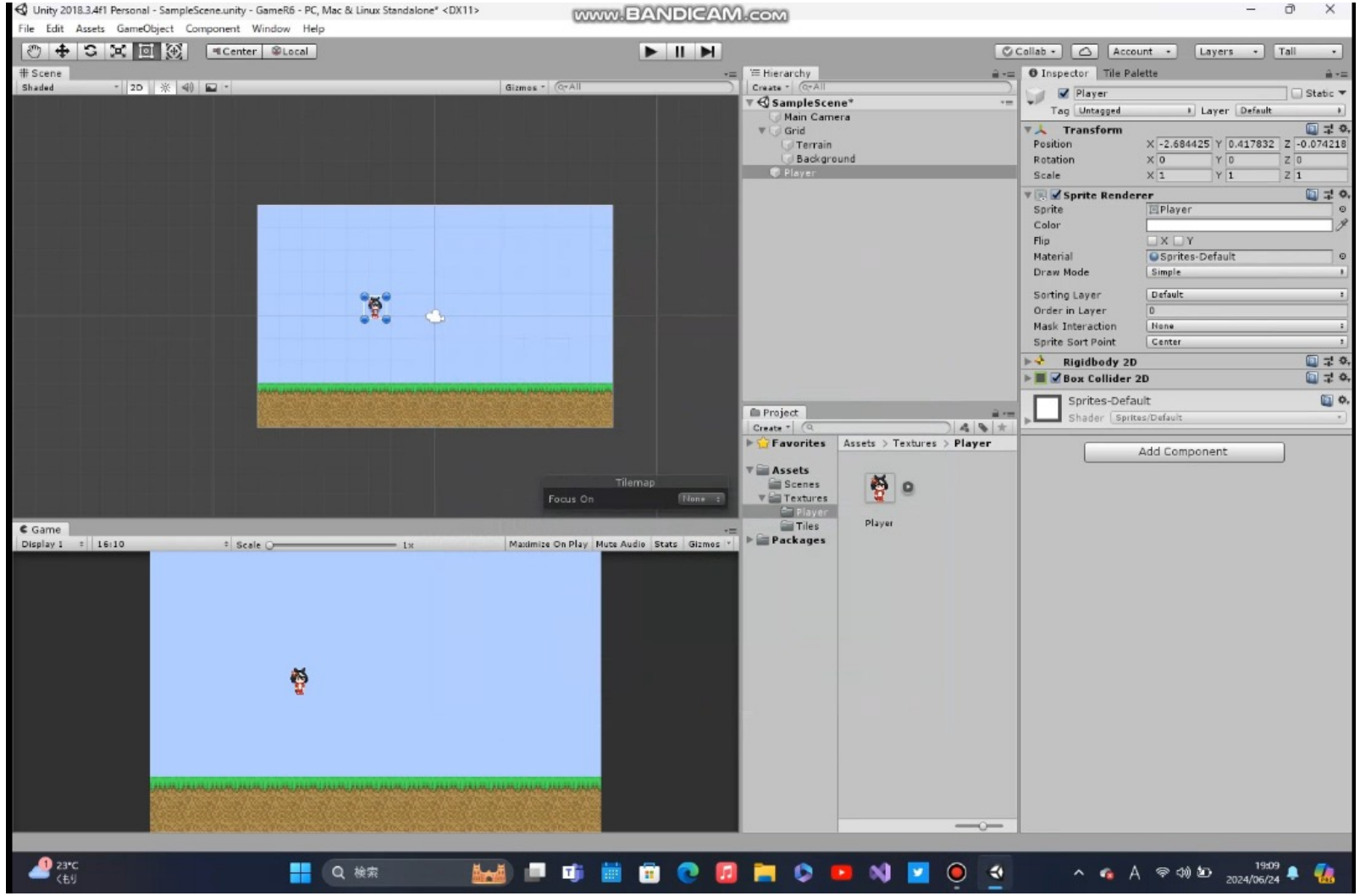


①Playerクリック

②Sprite Rendererの
　Spriteに画像をドラッグアンドドロップ

③▶で再生
　もう一回▶で停止

※必ず停止している状態で設定を行う。



【Project】

-Scriptsフォルダ作成-

①Assetsを選択

Create→folder

名前：Scripts

↓

② Scriptsフォルダをダブルクリック

③ Scriptsフォルダの中で右クリック

→Create→C＃Sprite

名前：Player

Assets

Scenes

Textures

Tiles

Player

Scripts

C#スクリプト
-Player-

（Playerが動いたり
するためのコード）

Assets

Scenes

Textures

Tiles

Player

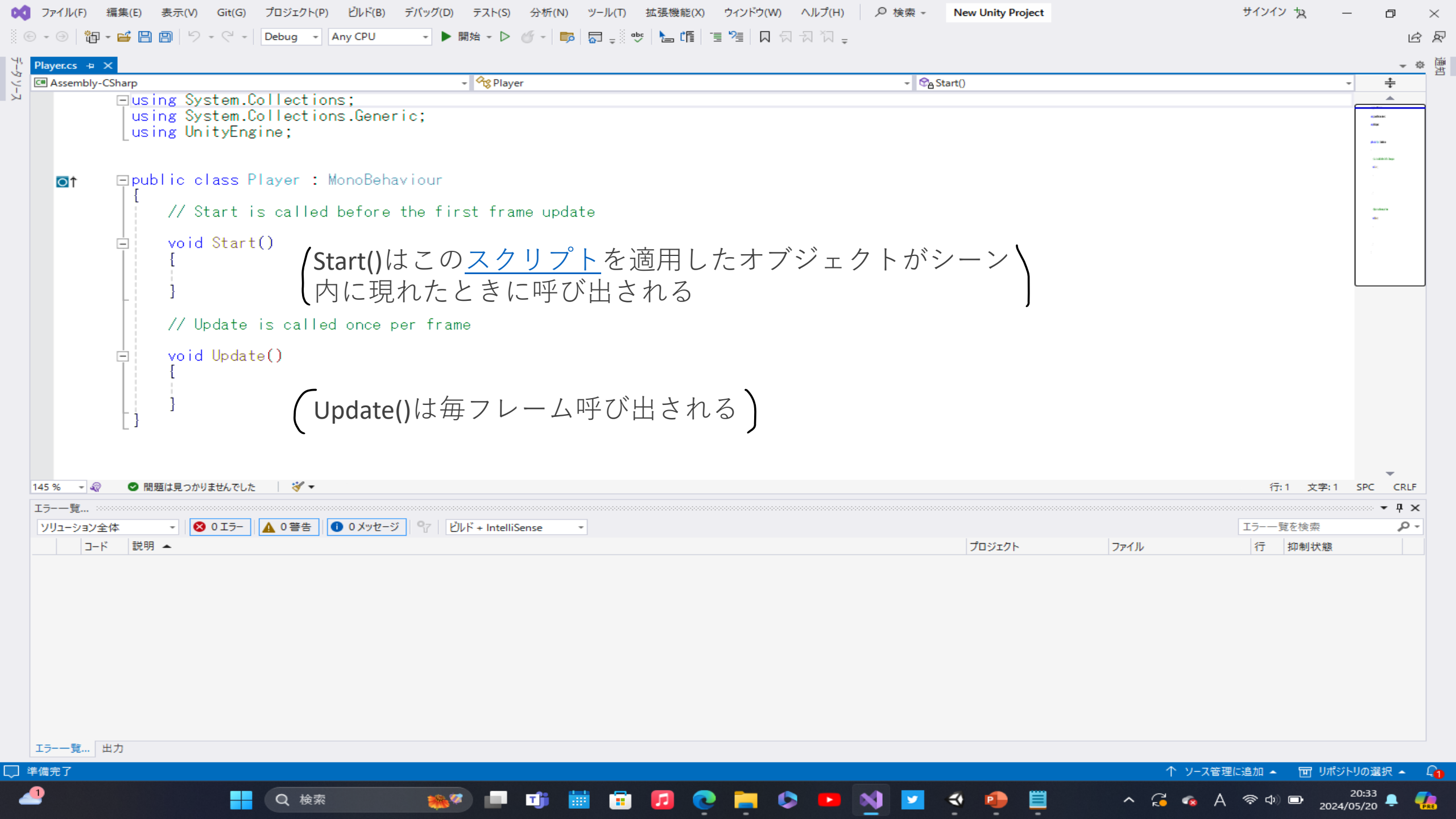
Scripts

C#スクリプト
-Player-

（Playerが動いたり
するためのコード）

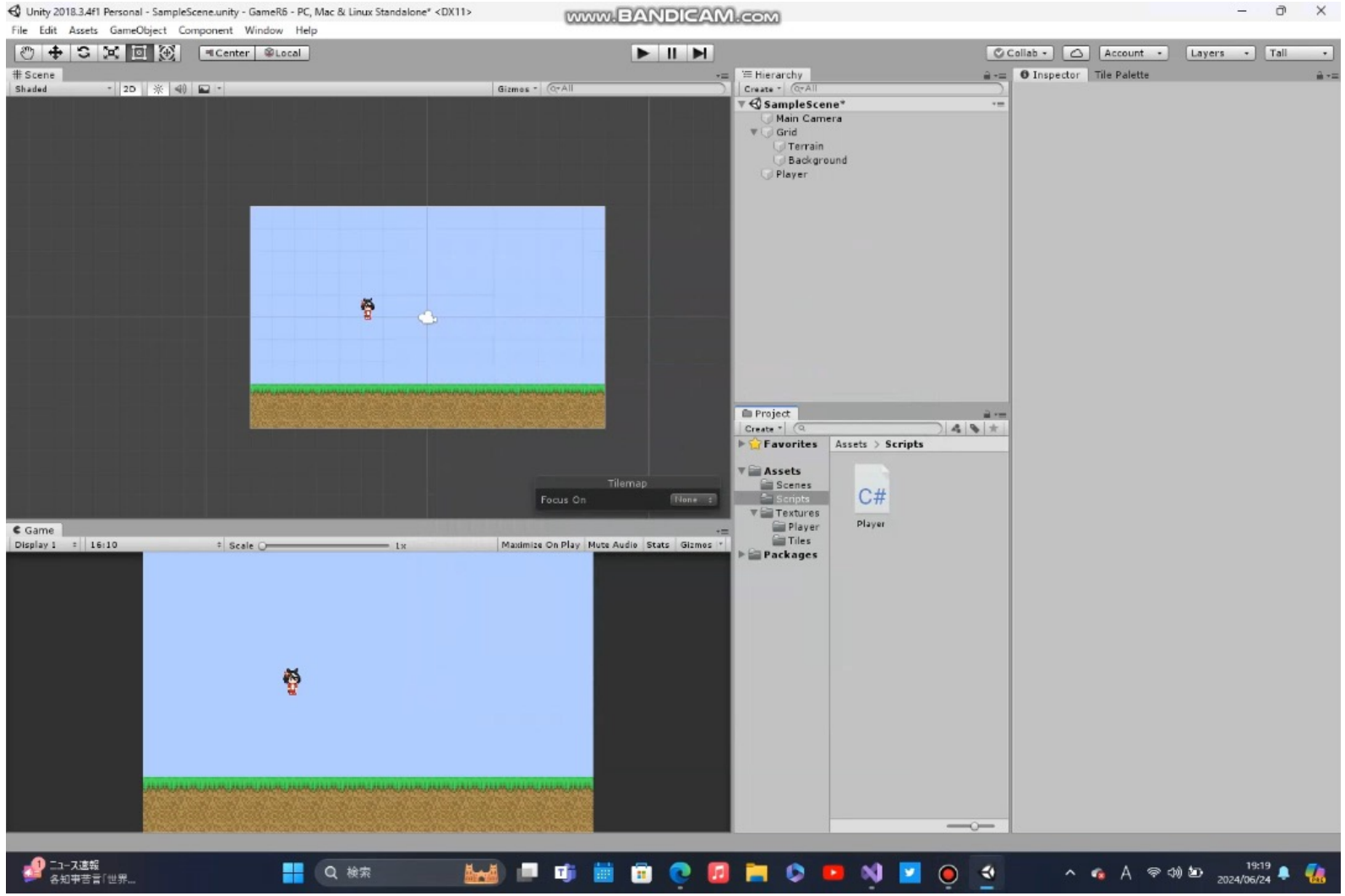
C#スクリプト
- CameraMovement -

（カメラの移動を制
御するコード）


ファイル(F) 編集(E) 表示(V) Git(G) プロジェクト(P) ビルド(B) デバッグ(D) テスト(S) 分析(N) ツール(T) 拡張機能(X) ウィンドウ(W) ヘルプ(H)
New Unity Project
Debug
Any CPU
開始
Player.cs
Assembly-CSharp
Player
Start()
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
public class Player : MonoBehaviour
{
// Start is called before the first frame update
void Start()
{
}
// Update is called once per frame
void Update()
{
}
}
Start()はこのスクリプトを適用したオブジェクトがシーン内に現れたときに呼び出される
Update()は毎フレーム呼び出される
問題は見つかりませんでした
エラー一覧...
ソリューション全体
0 エラー
0 警告
0 メッセージ
ビルド + IntelliSense
コード
説明
プロジェクト
ファイル
行
抑制状態
出力
準備完了
ソース管理に追加
リポジトリの選択
検索
20:33
2024/05/20

```
//Player.cs
using System.Collections;
using System.Collections.Generic;
using UnityEngine;

public class Player : MonoBehaviour
{
    private Rigidbody2D rb; //(a)

    void Start()
    {
        rb = GetComponent<Rigidbody2D>(); //(b)
    }

    void Update()
    {
        //Player Movement
        rb.velocity = new Vector2(Input.GetAxisRaw(“Horizontal”), rb.velocity.y); //(c)
        //この一文で、プレイヤーの左右移動を制御している
    }
}
```

スクリプトの適用



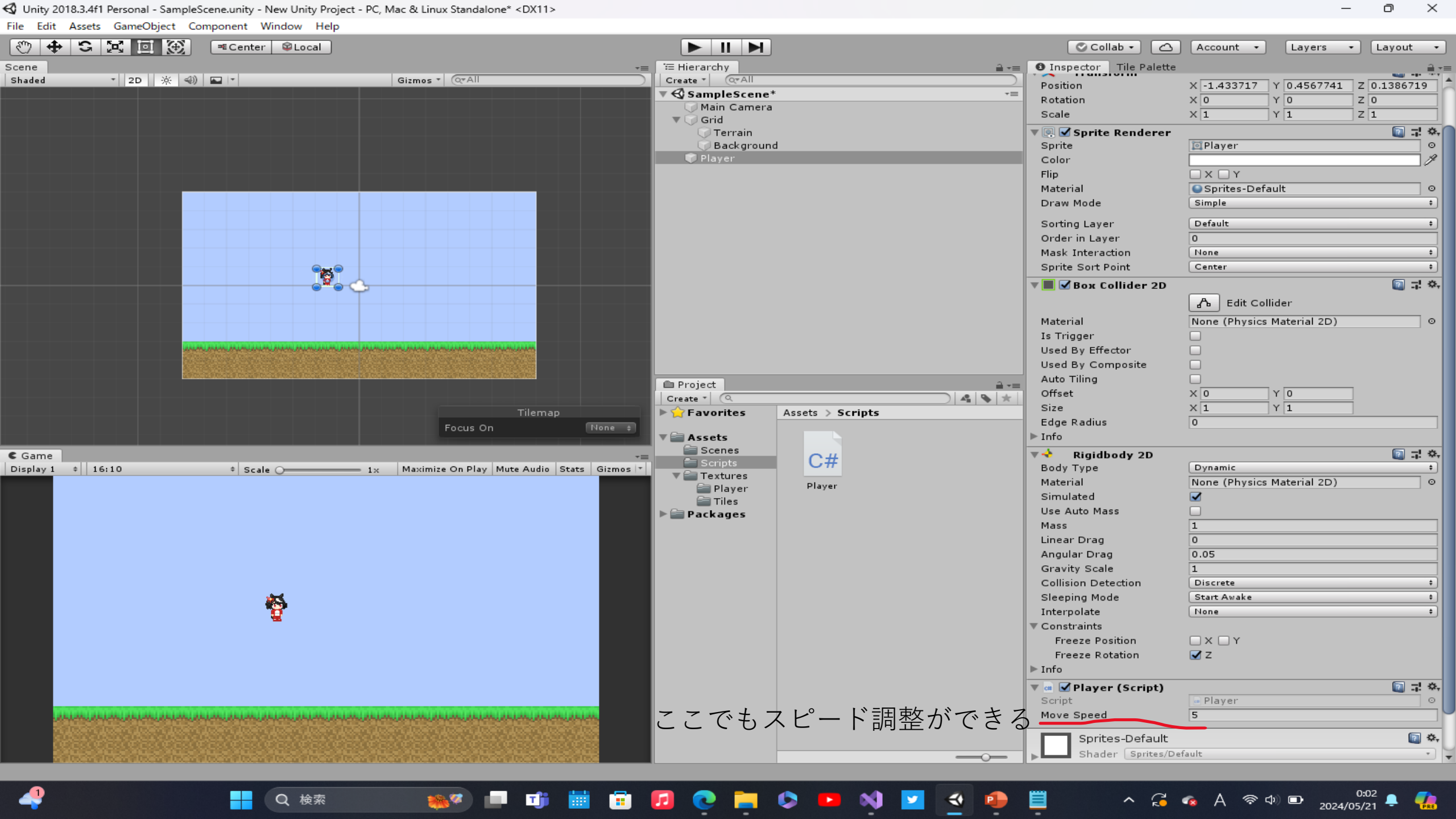
①Playerクリック

②Playerスクリプトをインスペクターに入れる

③Rigitbody２D
Freeze Rotation:　Z☑

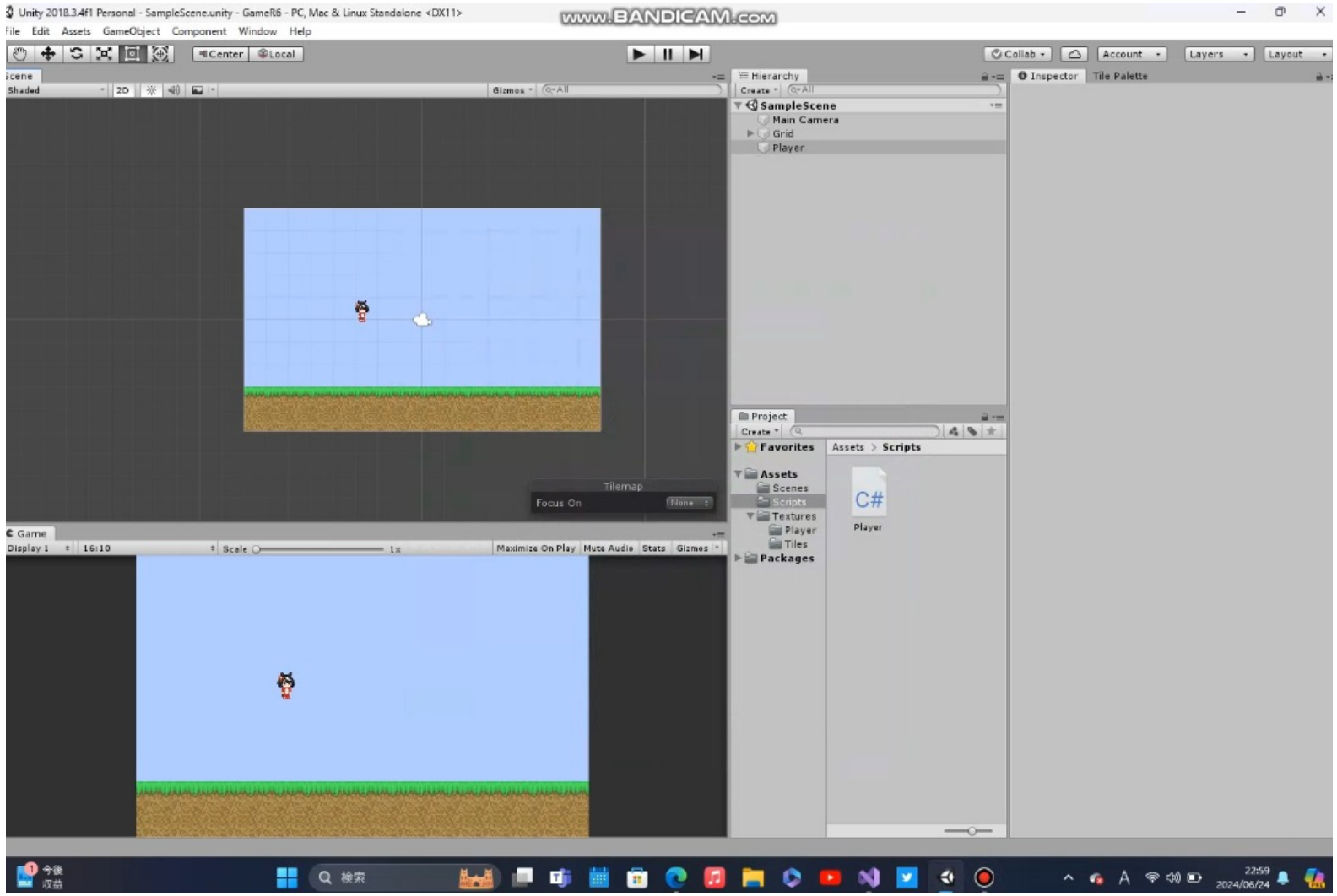
移動速度変更

```
//Player.cs
using UnityEngine;

public class Player : MonoBehaviour
{
    public float MoveSpeed = 5f;//(d)

    private Rigidbody2D rb;

    void Start()
    {
        rb = GetComponent<Rigidbody2D>();
    }

    void Update()
    {
        //Player Movement
        rb.velocity = new Vector2(Input.GetAxisRaw("Horizontal") * MoveSpeed, rb.velocity.y);//(e)
    }
}
```


Unity 2018.3.4f1 Personal - SampleScene.unity - New Unity Project - PC, Mac & Linux Standalone* <DX11>
File Edit Assets GameObject Component Window Help
Center Local
Collab Account Layers Layout
Scene
Shaded 2D Gizmos All
Tilemap
Focus On None
Hierarchy
Create All
SampleScene*
Main Camera
Grid
Terrain
Background
Player
Project
Create
Favorites
Assets
Scenes
Scripts
Textures
Player
Tiles
Packages
Assets > Scripts
C#
Player
Inspector
Tile Palette
Position X -1.433717 Y 0.4567741 Z 0.1386719
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Sprite Renderer
Sprite Player
Color
Flip X Y
Material Sprites-Default
Draw Mode Simple
Sorting Layer Default
Order in Layer 0
Mask Interaction None
Sprite Sort Point Center
Box Collider 2D
Edit Collider
Material None (Physics Material 2D)
Is Trigger
Used By Effector
Used By Composite
Auto Tiling
Offset X 0 Y 0
Size X 1 Y 1
Edge Radius 0
Info
Rigidbody 2D
Body Type Dynamic
Material None (Physics Material 2D)
Simulated
Use Auto Mass
Mass 1
Linear Drag 0
Angular Drag 0.05
Gravity Scale 1
Collision Detection Discrete
Sleeping Mode Start Awake
Interpolate None
Constraints
Freeze Position X Y
Freeze Rotation Z
Info
Player (Script)
Script Player
Move Speed 5
Sprites-Default
Shader Sprites/Default
Game
Display 1 16:10 Scale 1x Maximize On Play Mute Audio Stats Gizmos
ここでもスピード調整ができる
検索
0:02
2024/05/21

ジャンプ実装

```
//Player.cs
using UnityEngine;

public class Player : MonoBehaviour
{
    public float MoveSpeed = 3f;
    public float JumpForce = 15f;//(f)

    private Rigidbody2D rb;

    void Start()
    {
        rb = GetComponent<Rigidbody2D>();
    }

    void Update()
    {
        //Player Movement
        rb.velocity = new Vector2(Input.GetAxisRaw("Horizontal") * MoveSpeed, rb.velocity.y);

        if (Input.GetButtonDown("Jump"))//(g)
        {
            rb.velocity = new Vector2(rb.velocity.x, JumpForce);

        }
    }
}
```



[Rigitbody 2D]

①Gravity Scale→4
　：重力スケール

②collisionDetection
　　→Continuous
：衝突判定を連続的
→着地したときにめり込むのを防ぐ

[Box Collider 2D]

③Edit Collider

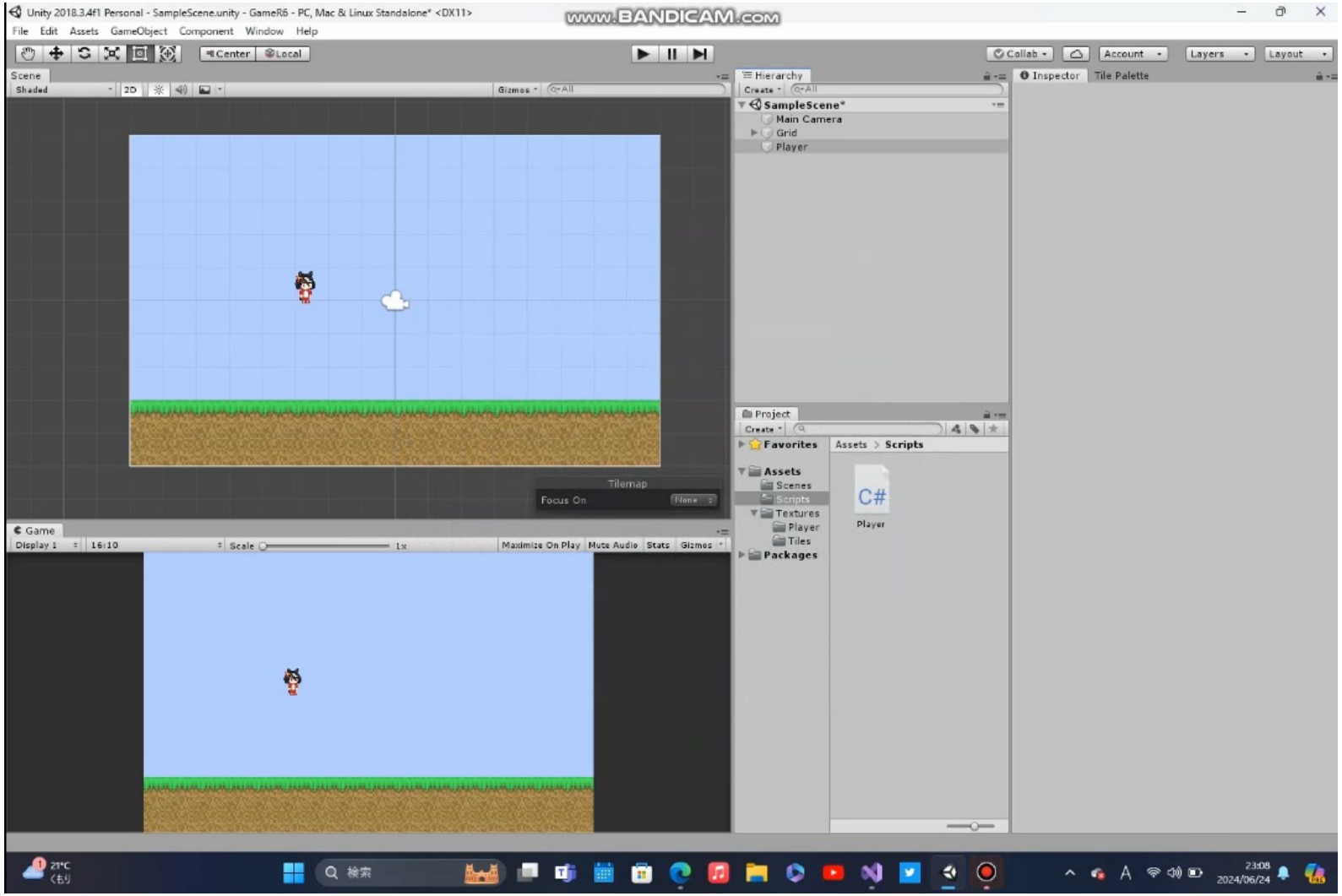
[Player(Script)]
Move Speed→3
Jump Force→15

向き変更

```
    //前のコード省略

    void Update()
    {
        //Player Movement
        rb.velocity = new Vector2(Input.GetAxisRaw("Horizontal") * MoveSpeed, rb.velocity.y);

        if (Input.GetButtonDown("Jump"))
        {
            rb.velocity = new Vector2(rb.velocity.x, JumpForce);
        }
        //Sprite Flip    //追加
        if (rb.velocity.x > 0)//(h)
        {
            GetComponent<SpriteRenderer>().flipX = false;
        }
        else if (rb.velocity.x < 0)
        {
            GetComponent<SpriteRenderer>().flipX = true;
        }
    }
}
```

着地判定

- 地面に足がついている状態でのみジャンプ



①Terrainを選択
Layer→Add Layer...

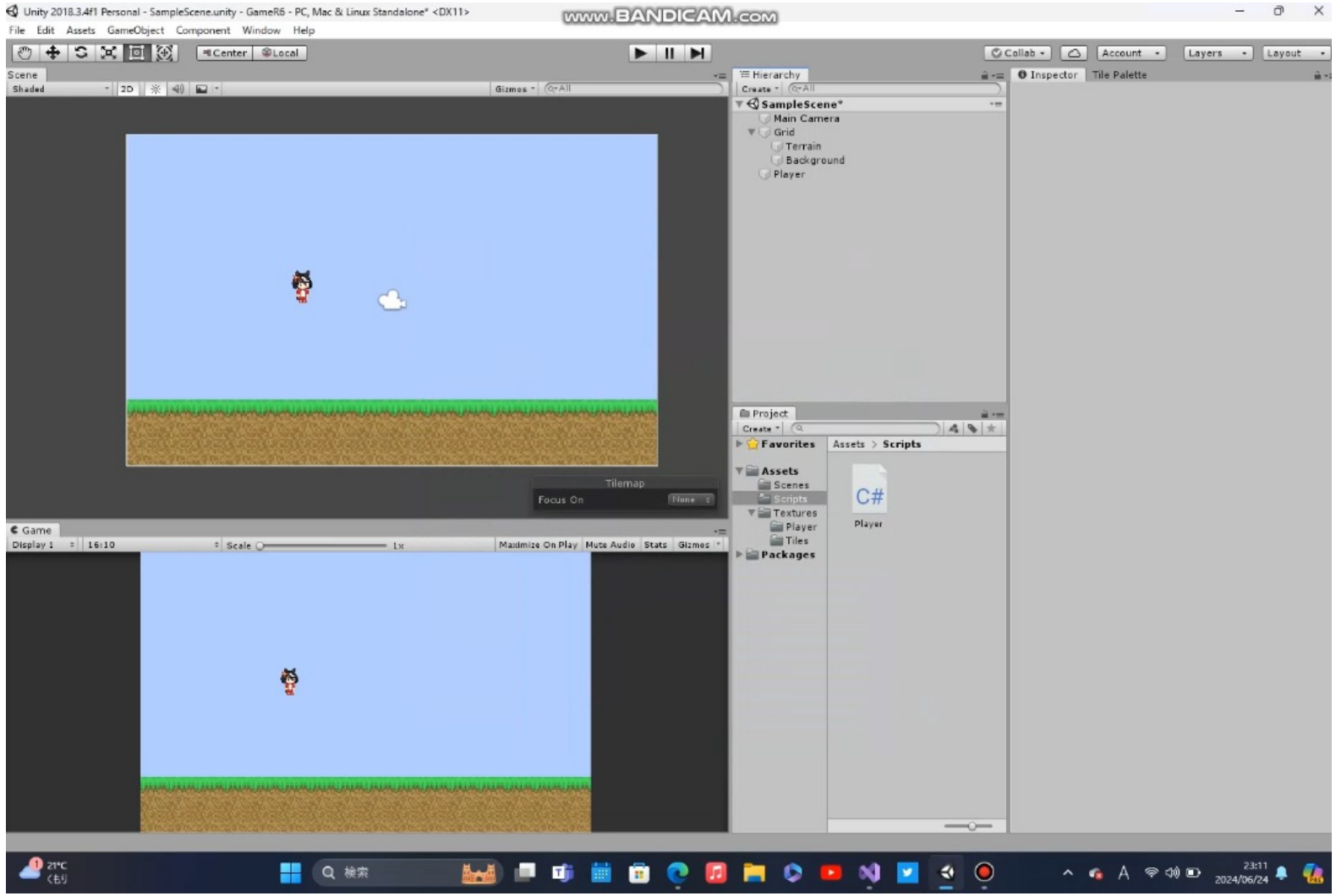
②User Layer 8
→Ground

※「ソートレイヤー」は表示上の設定に使うが、
こちらの「レイヤー」は主に衝突判定の設定で使う。
設定画面でのレイヤーの上下関係は関係はない。

```
//Player.cs
using UnityEngine;

public class Player : MonoBehaviour
{
    public float MoveSpeed = 3f;
    public float JumpForce = 15f;
    public LayerMask GroundLayer;//(i)

    private Rigidbody2D rb;

    void Start()
    {
        rb = GetComponent<Rigidbody2D>();
    }

}
```

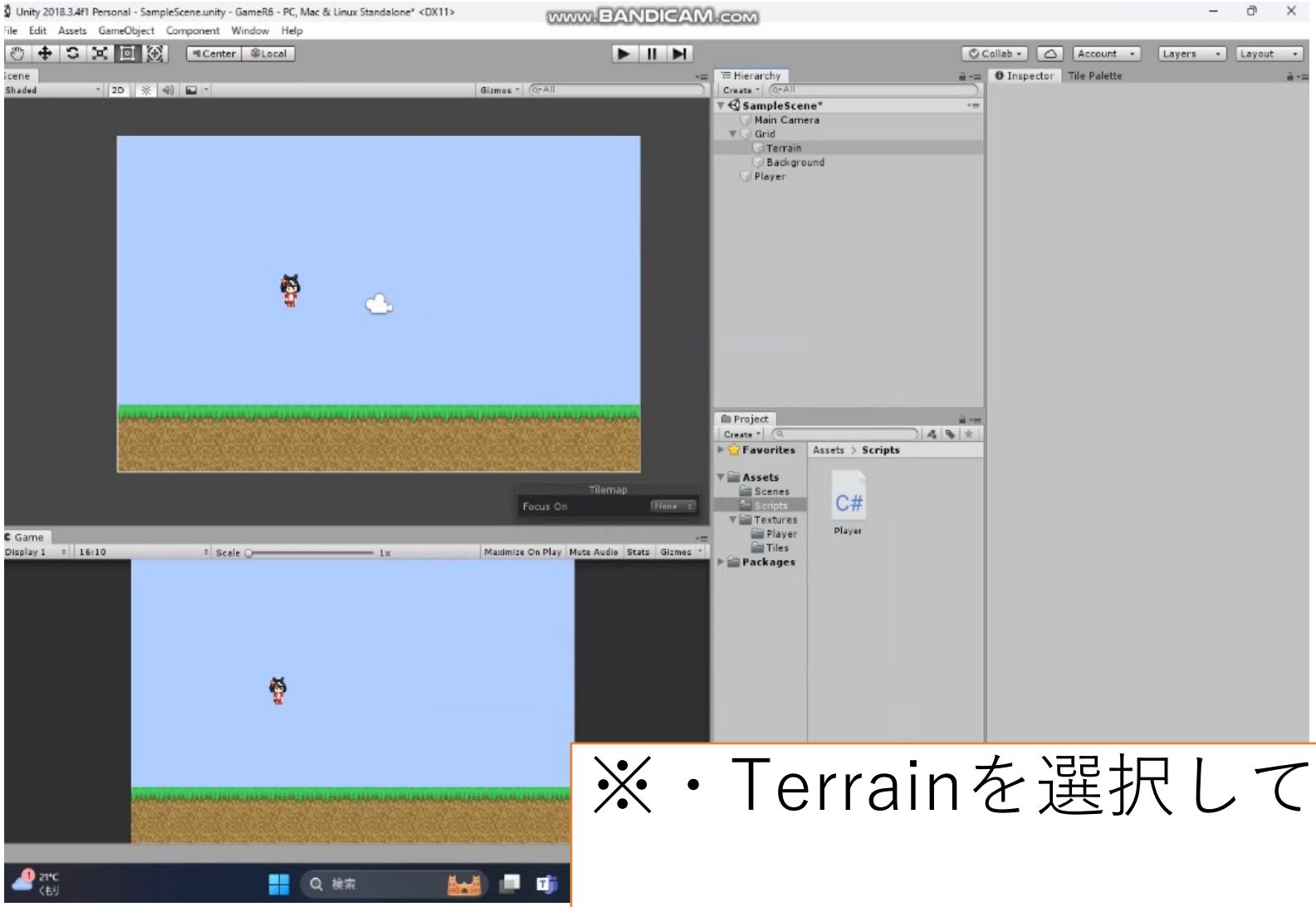
（前のコード省略）

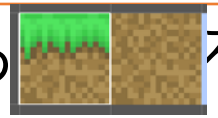
```
    void Update()
    {
        //Player Movement
        rb.velocity = new Vector2(Input.GetAxisRaw("Horizontal") * MoveSpeed, rb.velocity.y);

        if (Input.GetButtonDown("Jump") && isGrounded())//(j)
        {
            rb.velocity = new Vector2(rb.velocity.x, JumpForce);
        }
        //Sprite Flip
        //省略
    }

    private bool isGrounded()//(k)
    {
        BoxCollider2D c = GetComponent<BoxCollider2D>();
        return Physics2D.BoxCast(c.bounds.center, c.bounds.size, 0f, Vector2.down, .1f, GroundLayer);
    }
}
```

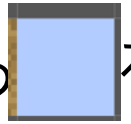


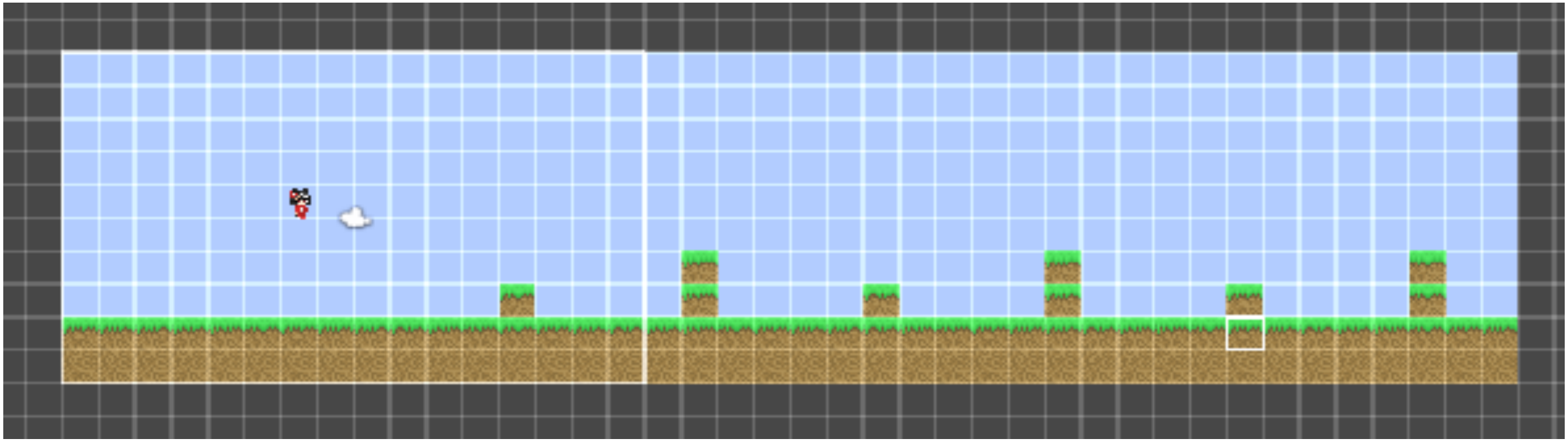
Playerを選択
[Player(Script)]
Ground Layer→Ground

マップ拡張・カメラ移動

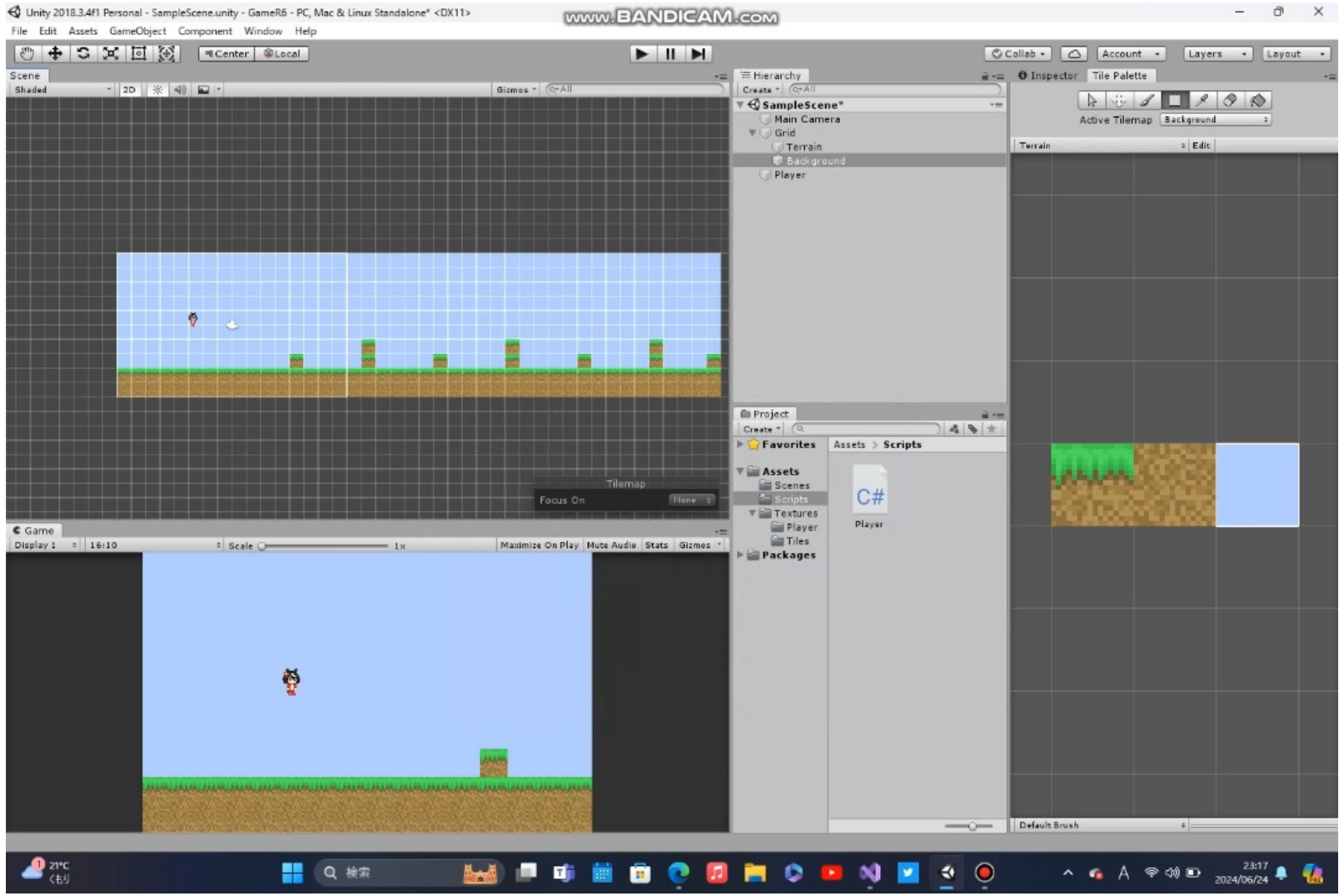


※・Terrainを選択してから  を挿入

・Backgroundを選択してから  を挿入



カメラ移動：カメラの移動を制御するスクリプトを作成



・scripts→Create
→C＃Script

名前：
CameraMovement

Assets

Scenes

Textures

Tiles

Player

Scripts

C#スクリプト
-Player-

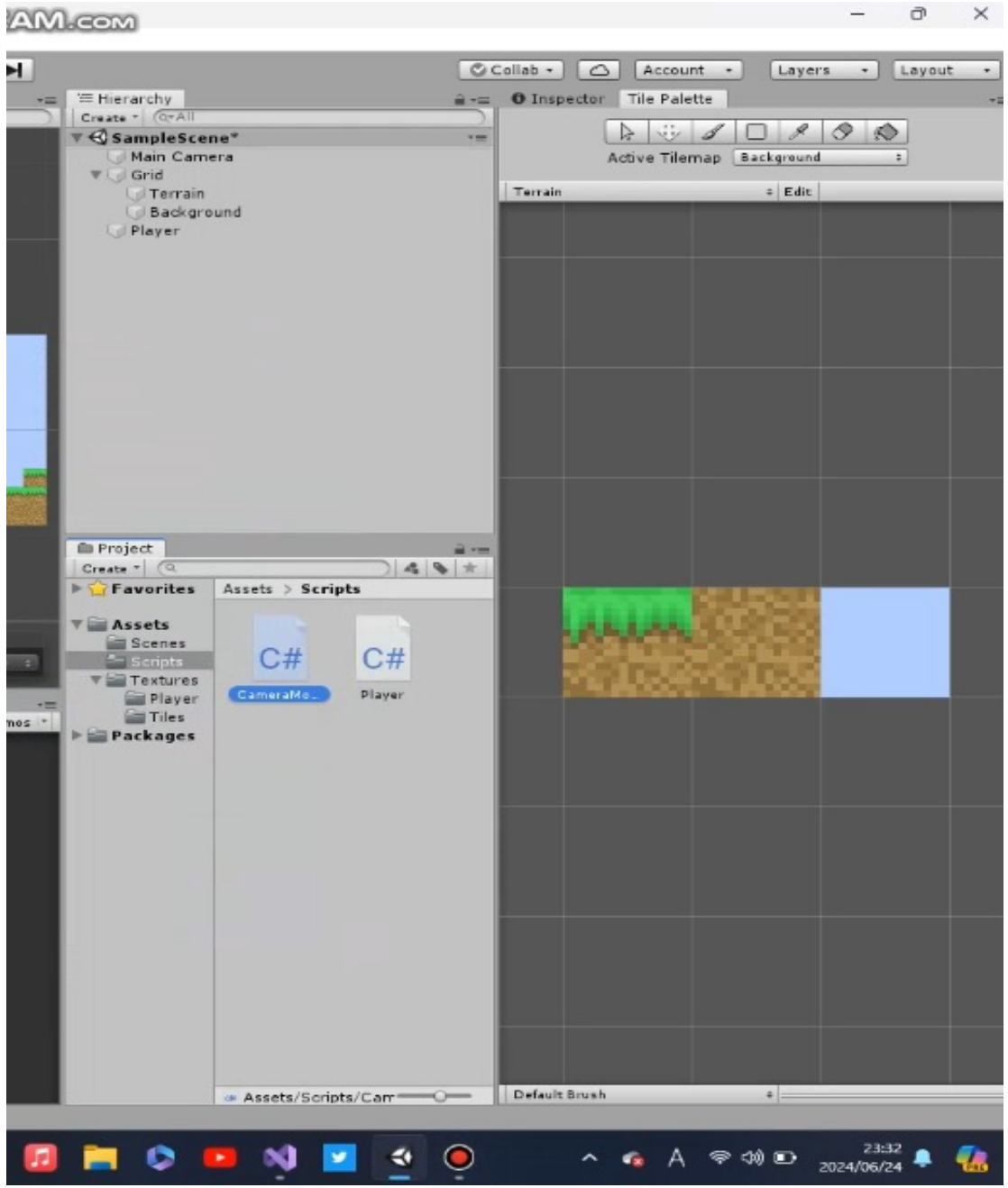
（Playerが動いたり
するためのコード）

C#スクリプト
- CameraMovement -

（カメラの移動を制
御するコード）

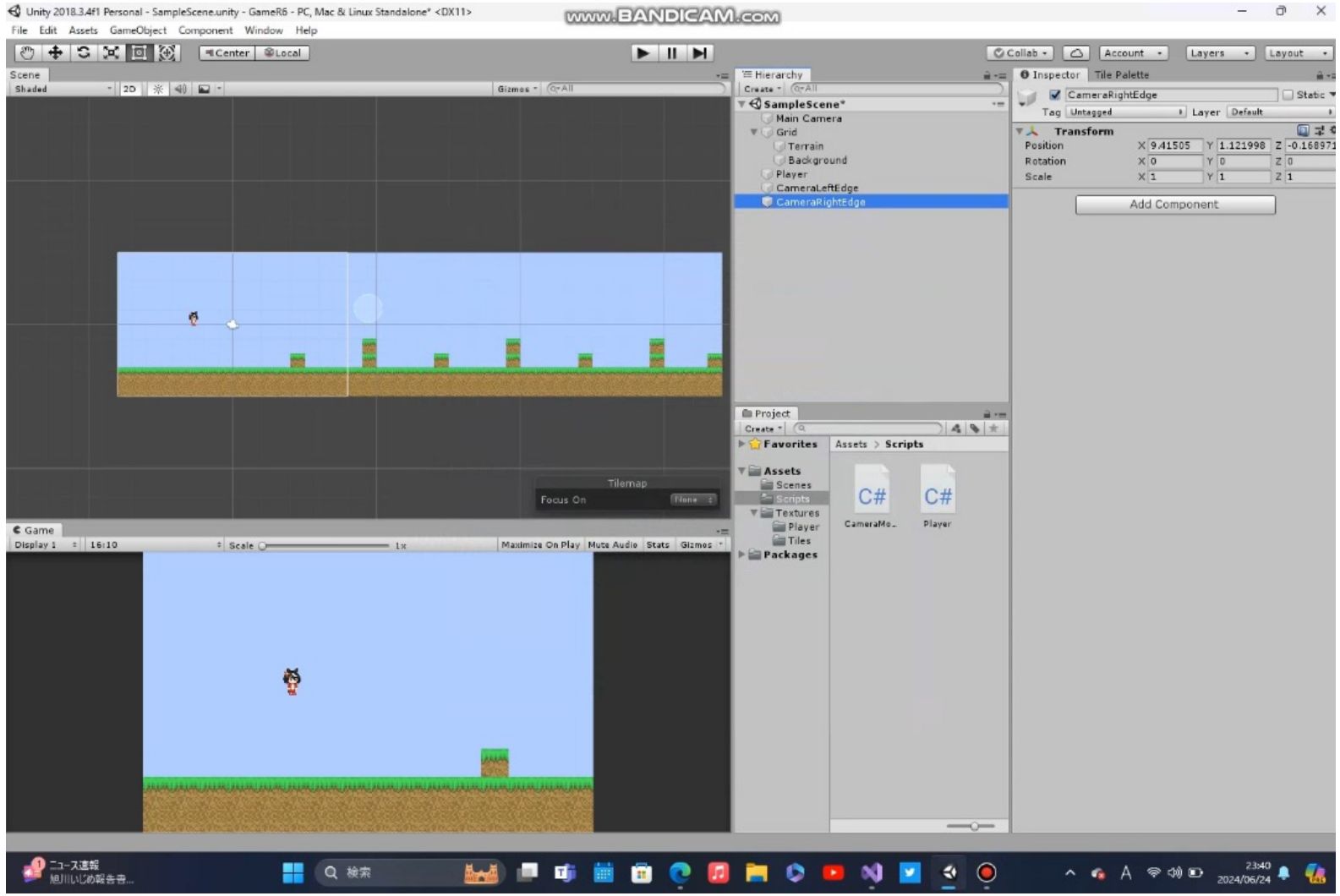
```
//CameraMovement.cs
using UnityEngine;

public class CameraMovement : MonoBehaviour
{
    public GameObject Target;
    public GameObject LeftEdge;
    public GameObject RightEdge;

    void Start()
    {

    }

    void Update()
    {
        this.transform.position = new Vector3(Target.transform.position.x, this.transform.position.y, this.transform.position.z);

        if (this.transform.position.x <= LeftEdge.transform.position.x)
        {
            this.transform.position = new Vector3(LeftEdge.transform.position.x, this.transform.position.y,
this.transform.position.z);
        }
        else if (this.transform.position.x >= RightEdge.transform.position.x)
        {
            this.transform.position = new Vector3(RightEdge.transform.position.x, this.transform.position.y,
this.transform.position.z);
        }
    }
}
```

端検知用オブジェクト



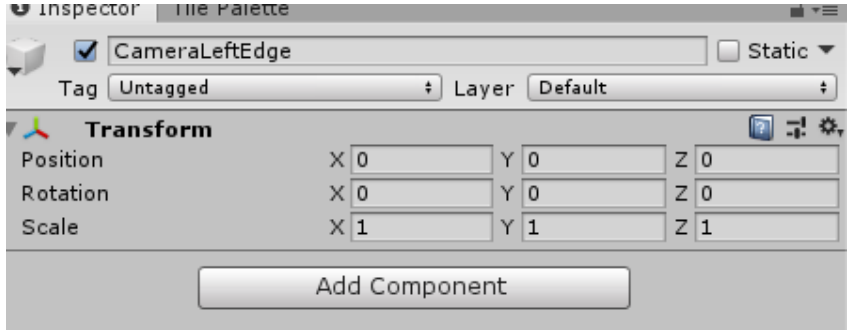
①MainCameraクリック
→CameraMovementスクリプト
をインスペクターに入れる

【Hierarchy】
②Create→Create Empty × 2
名前：CameraLeftEdge
　　CameraRightEdge

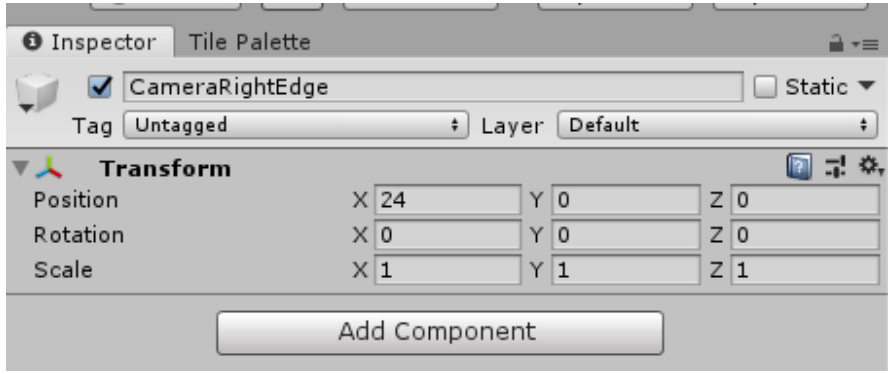


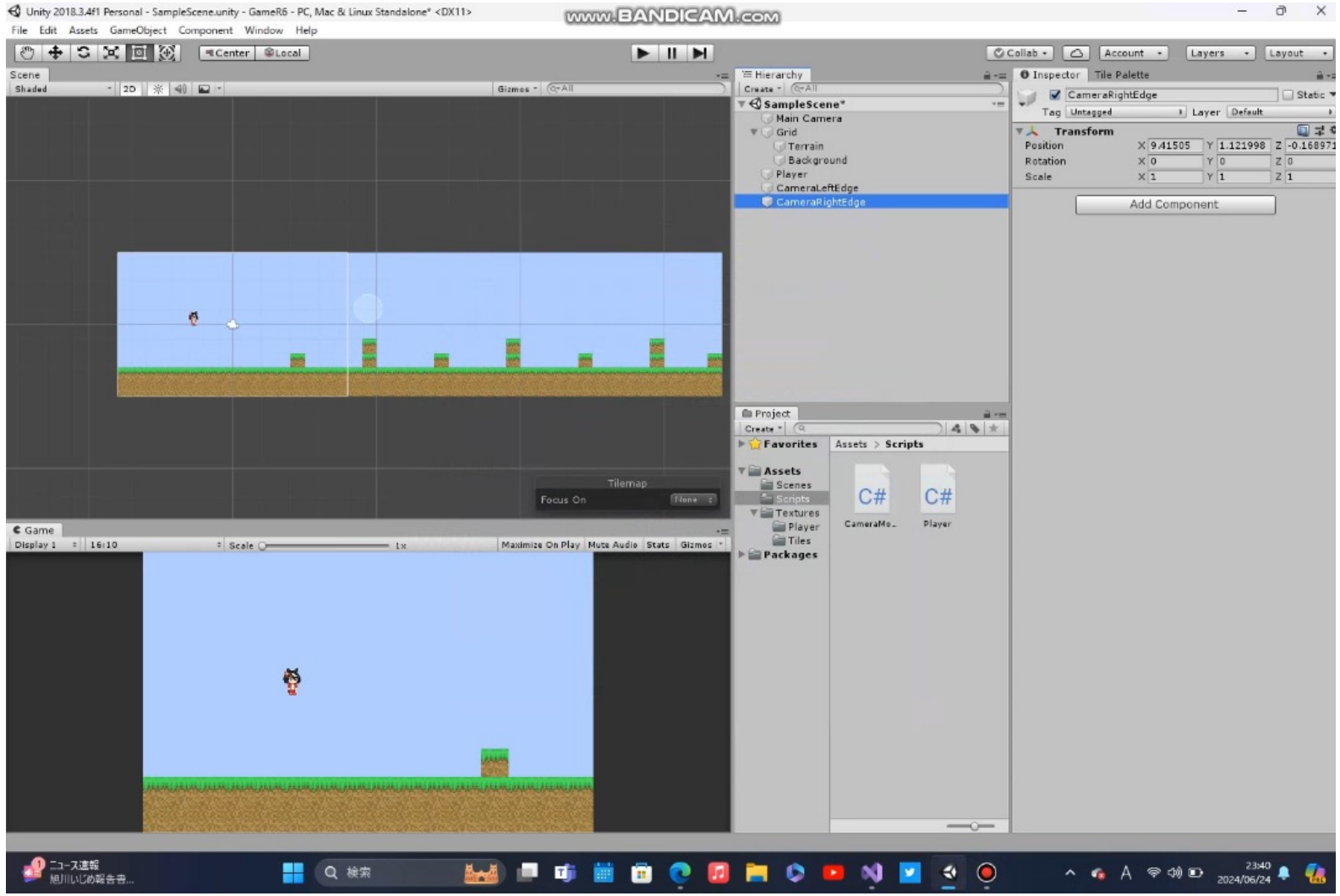
⚙→Reset
（値をリセットする）

CameraLeftEdge



CameraRightEdge
右端から８マスのところ

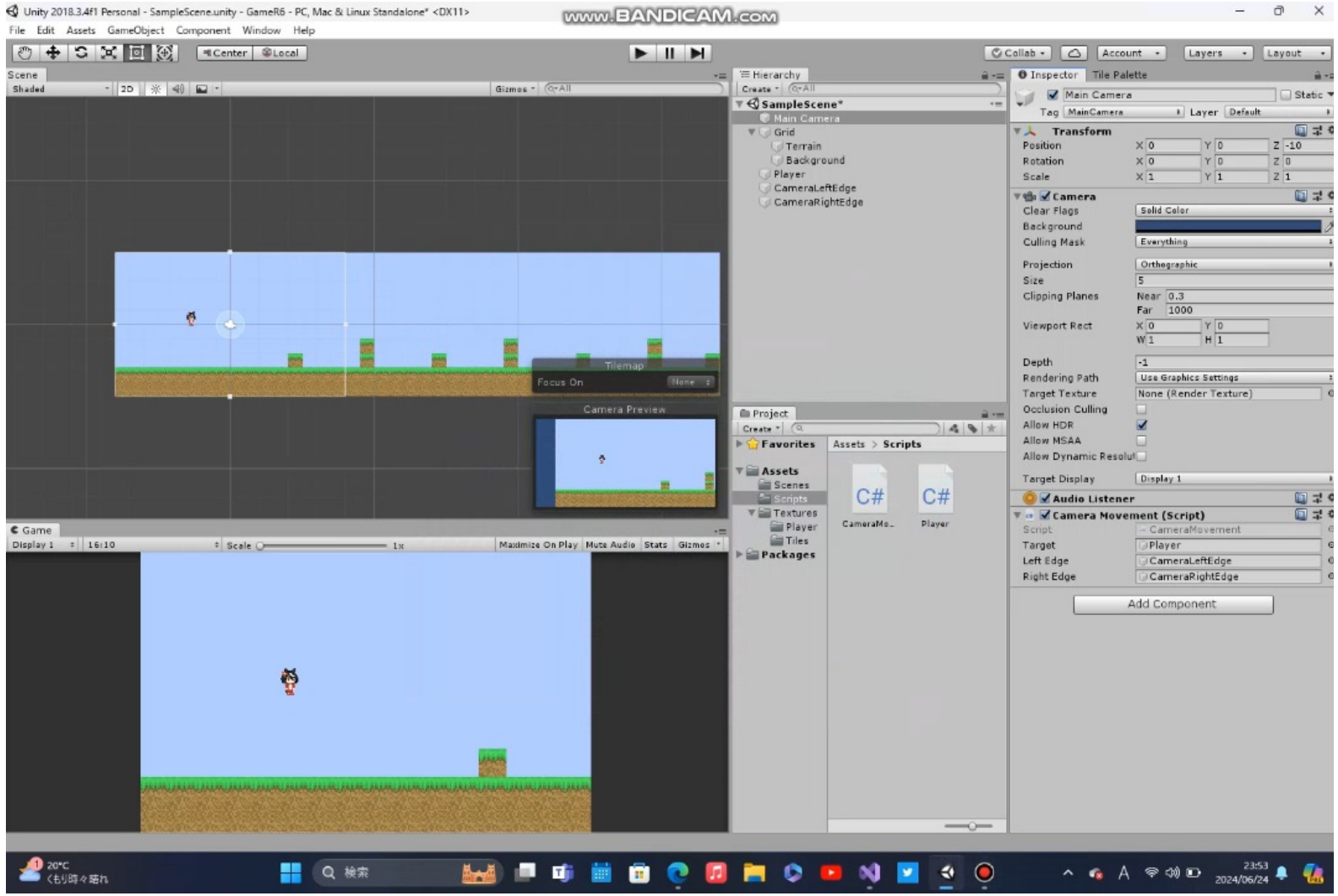




Camera Movement(Script)

- Target→Player
- LeftEdge
  →CameraLeftEdge
- Right Edge
  →CameraRightEdge

ジャンプ修正【移動とジャンプを同時に行うと壁に接触する】

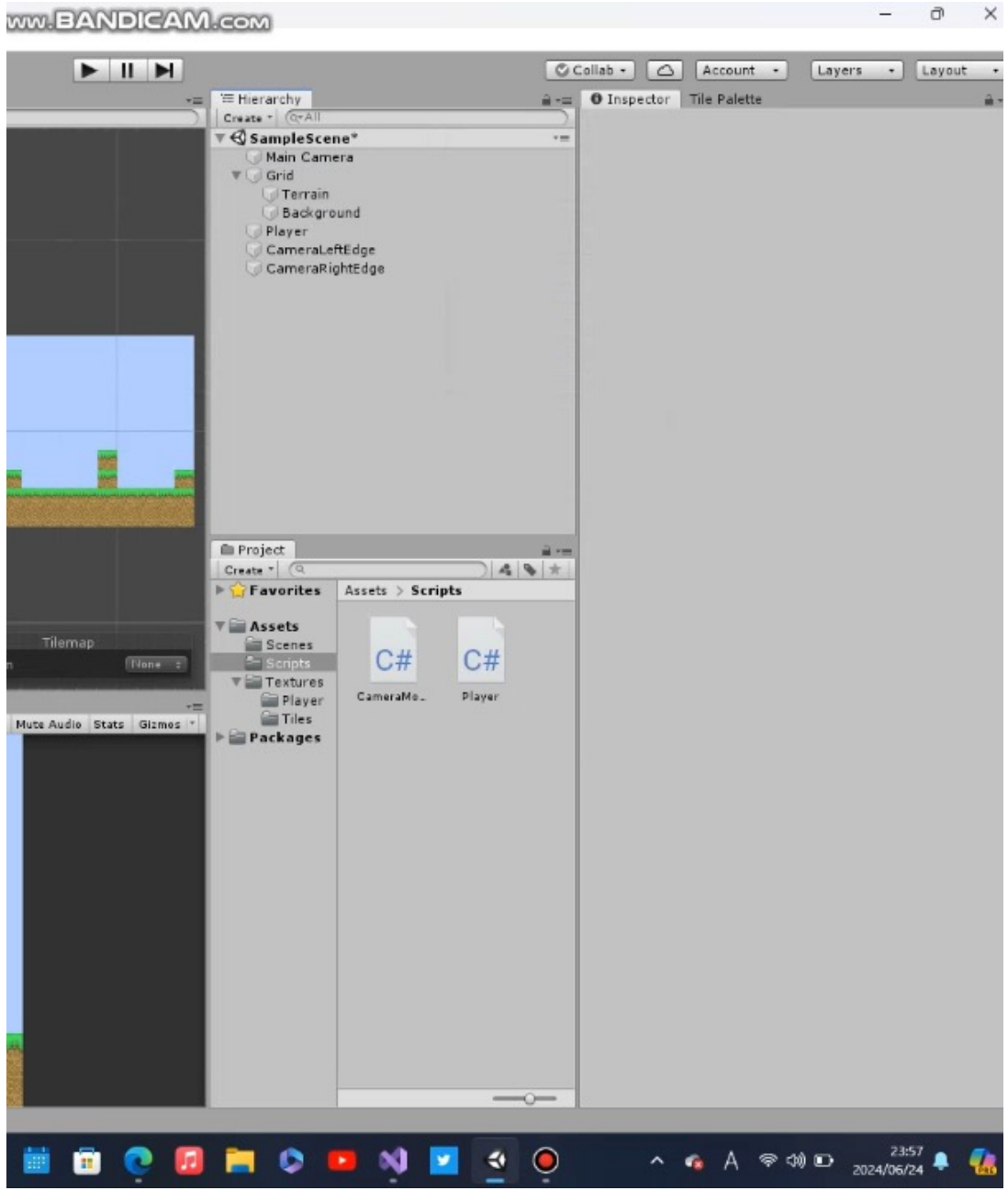


①Terrain選択
→Add Component
・**Platform Effector 2D**
追加

【 Composite Collider 2D 】
・Used By Effector☑

【Platform Effector 2D】
・Use One Way☐

デス判定

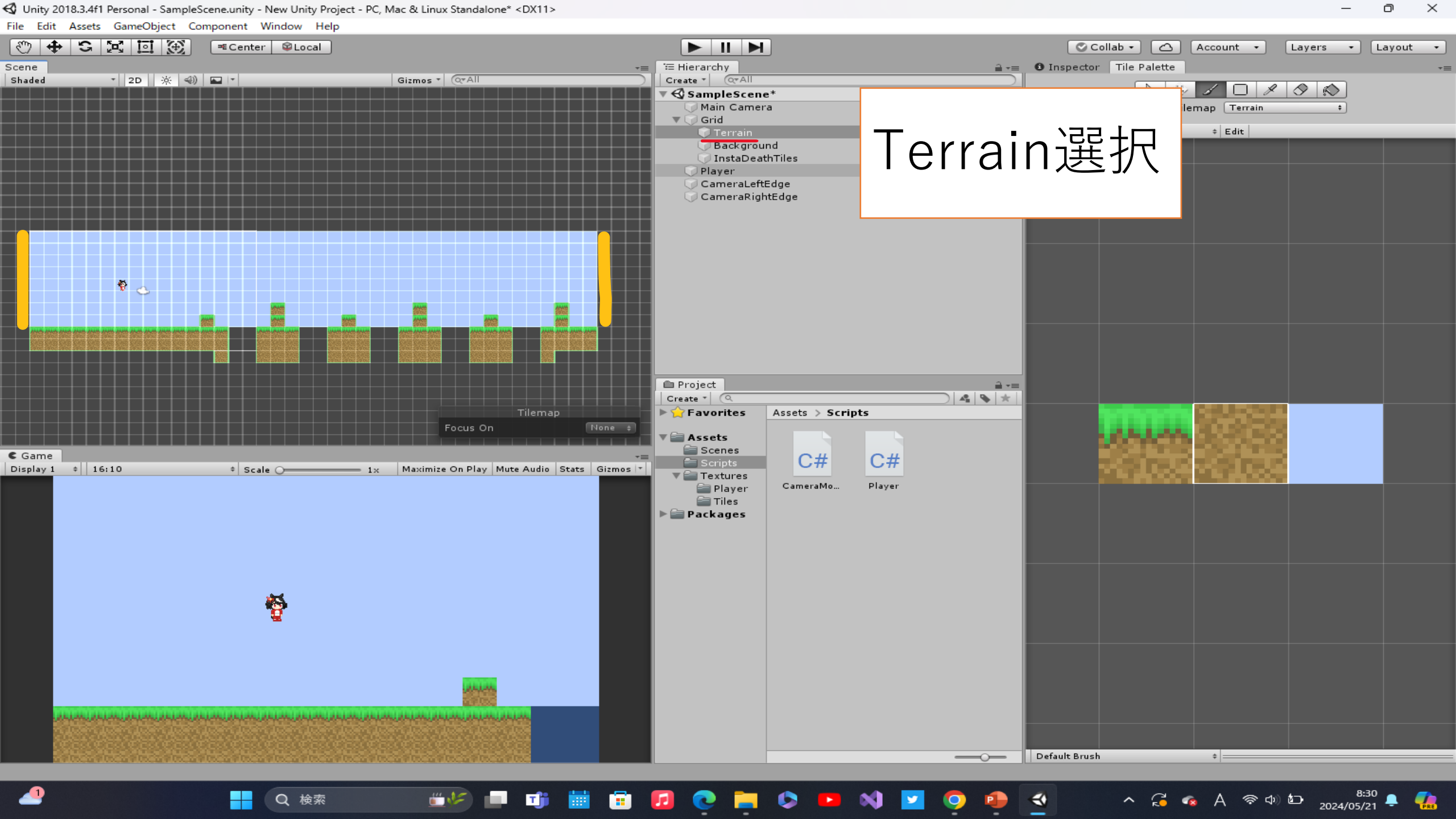


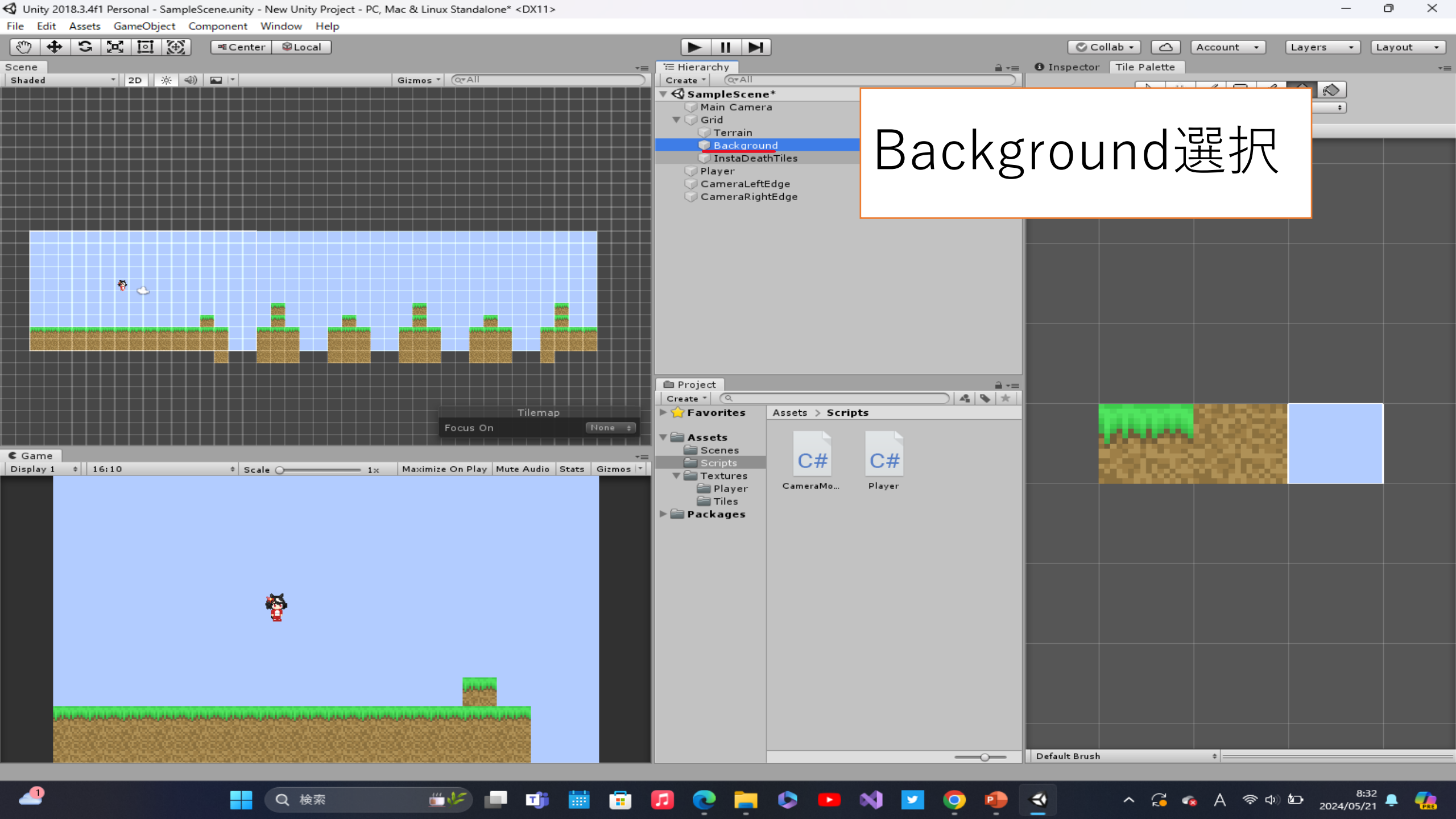
Grid選択

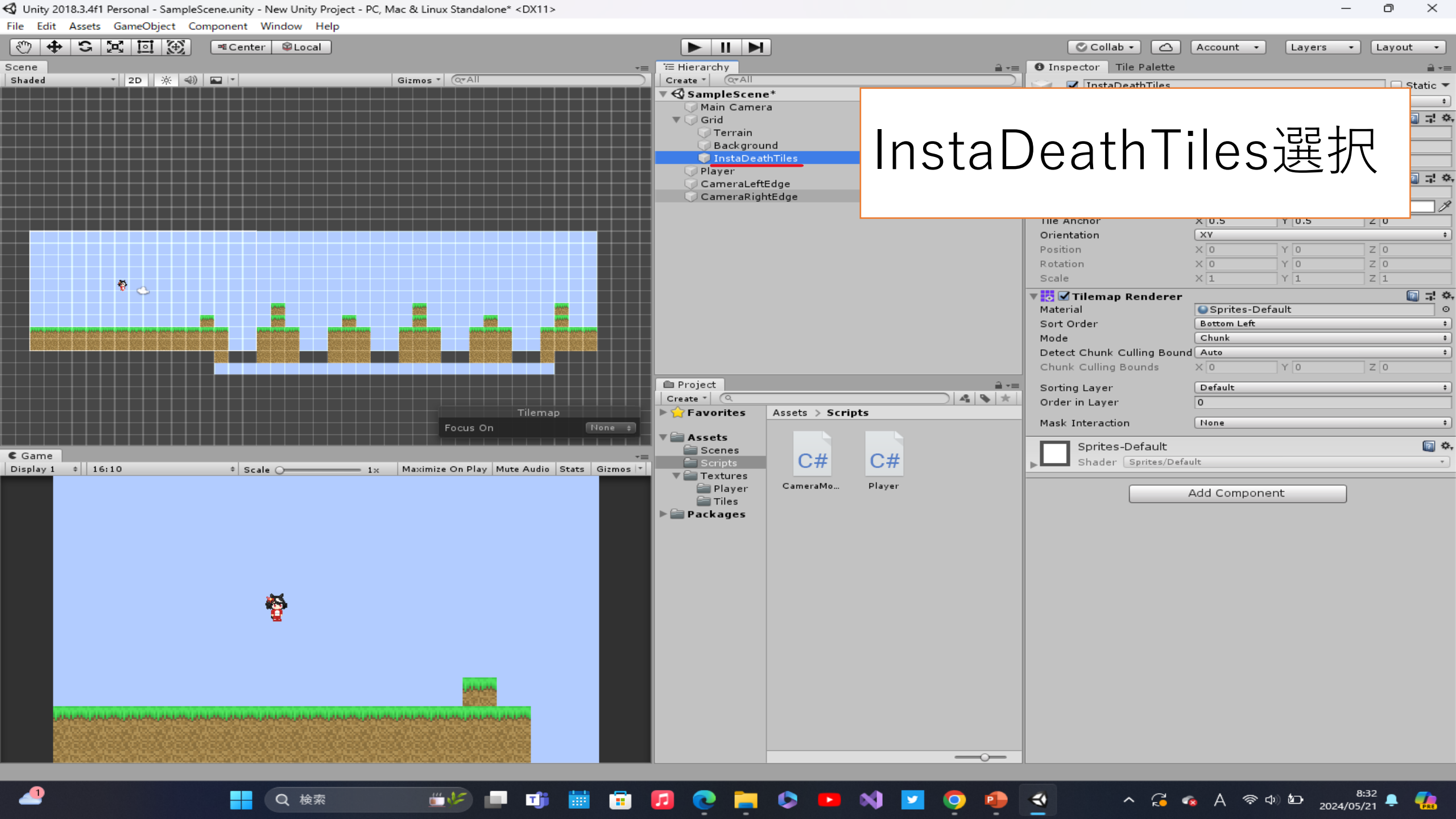
Create→2DObject

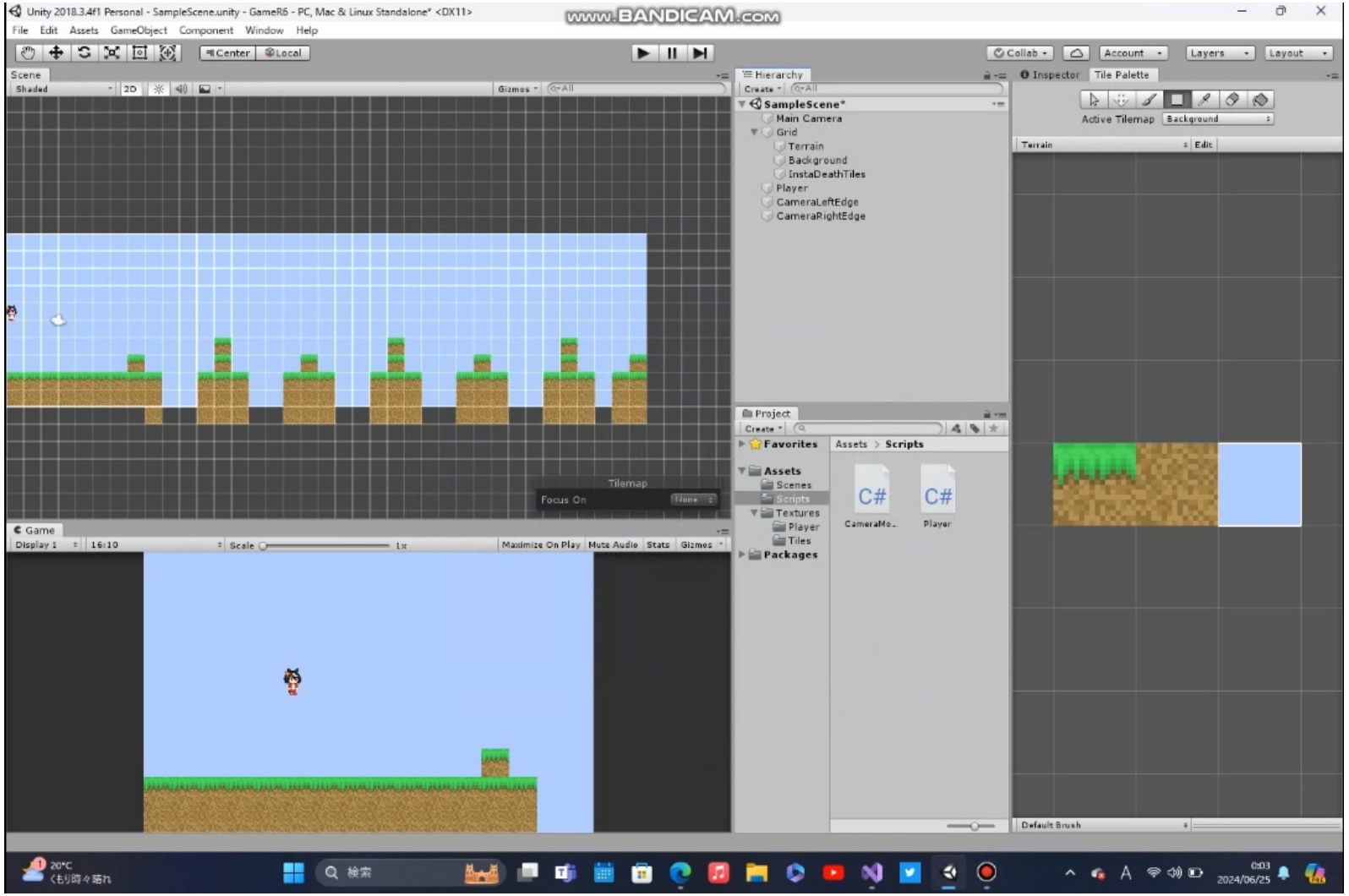
→Tilemap

名前：InstaDeathTiles


Unity 2018.3.4f1 Personal - SampleScene.unity - New Unity Project - PC, Mac & Linux Standalone* <DX11>
File Edit Assets GameObject Component Window Help
Center Local
Collab Account Layers Layout
Scene
Shaded 2D Gizmos All
Tilemap
Focus On None
Hierarchy
Create All
SampleScene*
Main Camera
Grid
Terrain
Background
InstaDeathTiles
Player
CameraLeftEdge
CameraRightEdge
Terrain選択
Inspector Tile Palette
Terrain
Edit
Project
Create
Favorites
Assets
Scenes
Scripts
Textures
Player
Tiles
Packages
Assets > Scripts
C#
CameraMo...
C#
Player
Default Brush
Game
Display 1 16:10 Scale 1x Maximize On Play Mute Audio Stats Gizmos
検索
8:30
2024/05/21


Unity 2018.3.4f1 Personal - SampleScene.unity - New Unity Project - PC, Mac & Linux Standalone* <DX11>
File Edit Assets GameObject Component Window Help
Center Local
Collab Account Layers Layout
Scene
Shaded 2D Gizmos All
Tilemap
Focus On None
Hierarchy
Create All
SampleScene*
Main Camera
Grid
Terrain
Background
InstaDeathTiles
Player
CameraLeftEdge
CameraRightEdge
Inspector Tile Palette
Project
Create
Favorites
Assets
Scenes
Scripts
Textures
Player
Tiles
Packages
Assets > Scripts
C#
CameraMo...
C#
Player
Default Brush
Game
Display 1 16:10 Scale 1x Maximize On Play Mute Audio Stats Gizmos
Background選択
検索
8:32
2024/05/21


Unity 2018.3.4f1 Personal - SampleScene.unity - New Unity Project - PC, Mac & Linux Standalone* <DX11>
File Edit Assets GameObject Component Window Help
Center Local
Collab Account Layers Layout
Scene
Shaded 2D Gizmos All
Tilemap
Focus On None
Hierarchy
Create All
SampleScene*
Main Camera
Grid
Terrain
Background
InstaDeathTiles
Player
CameraLeftEdge
CameraRightEdge
InstaDeathTiles選択
Inspector Tile Palette
InstaDeathTiles
Static
Tile Anchor X 0.5 Y 0.5 Z 0
Orientation XY
Position X 0 Y 0 Z 0
Rotation X 0 Y 0 Z 0
Scale X 1 Y 1 Z 1
Tilemap Renderer
Material Sprites-Default
Sort Order Bottom Left
Mode Chunk
Detect Chunk Culling Bound Auto
Chunk Culling Bounds X 0 Y 0 Z 0
Sorting Layer Default
Order in Layer 0
Mask Interaction None
Sprites-Default
Shader Sprites/Default
Add Component
Project
Create
Favorites
Assets
Scenes
Scripts
Textures
Player
Tiles
Packages
Assets > Scripts
C#
CameraMo...
C#
Player
Game
Display 1 16:10 Scale 1x Maximize On Play Mute Audio Stats Gizmos
検索
8:32
2024/05/21



InstaDeathTilesの
Inspectorからタグを追加

①InstaDeathTiles選択
→Tag→Add Tag

②「＋」
→名前：InstaDeath
→Save

③ Tag→InstaDeath

④InstaDeathTiles選択
→Add Component
→**Tilemap Collider 2D**
追加

```
//Player.cs
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement; //(m)

//途中のコード省略

    private bool isGrounded()
    {
        BoxCollider2D c = GetComponent<BoxCollider2D>();
        return Physics2D.BoxCast(c.bounds.center, c.bounds.size, 0f, Vector2.down, .1f, GroundLayer);
    }
    private void OnCollisionEnter2D(Collision2D collision)//(m)
    {
        if (collision.gameObject.CompareTag("InstaDeath"))
        {
            SceneManager.LoadScene("SampleScene");
        }
    }
  //タグがInstaDeathのオブジェクトと衝突した際に、シーンをロードし直すという処理を行っている
}
```