

**Національний Технічний Університет України КПІ**

Факультет Інформатики та Обчислювальної Техніки  
Кафедра інформатики та програмної інженерії

**Лабораторна робота №1**

З дисципліни «Розробка ігрових застосувань. Unity рішення»

Дослідження базового патерну ігрового рушія Unity на прикладі двовимірної технології

**Перевірив:**

Викладач

Катін Павло Юрійович

Оцінка:\_\_\_\_\_\_\_\_\_\_\_**Виконав:**

Студент групи ІТ-92

Михайлов П.П.

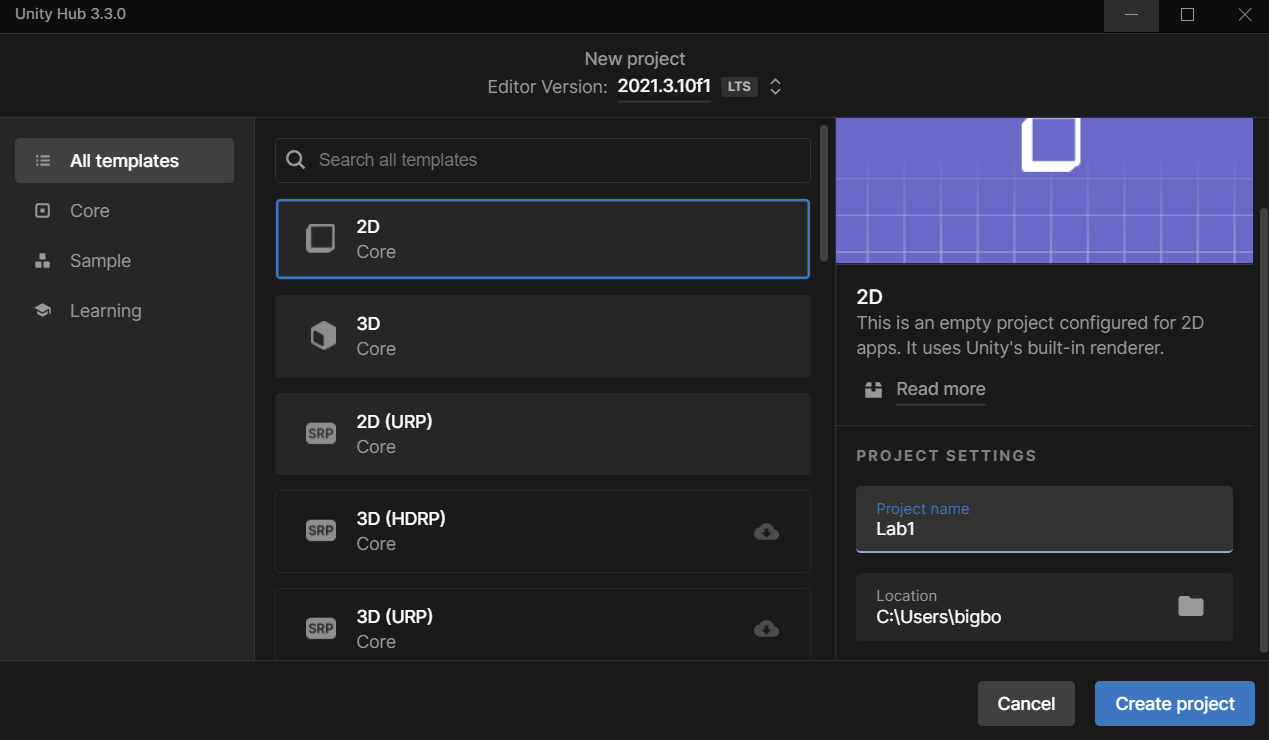
Варіант №6

Дата: \_\_\_\_\_\_\_\_\_\_\_\_\_

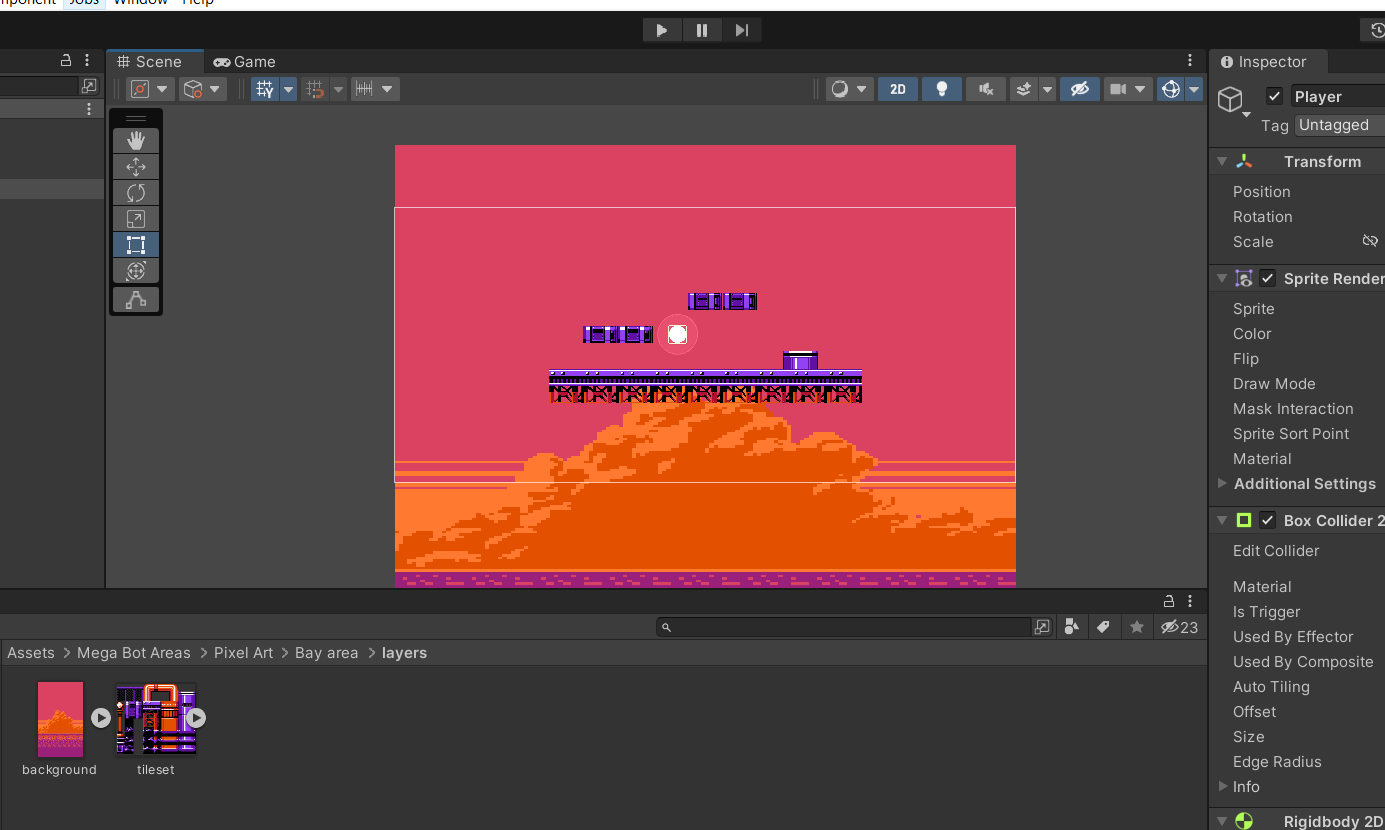
**Мета:** набуття знань, умінь та навичок з технології розроблення основ проекту з використанням обраної мови програмування у обраній парадигмі.

**Виконання лабораторної роботи**

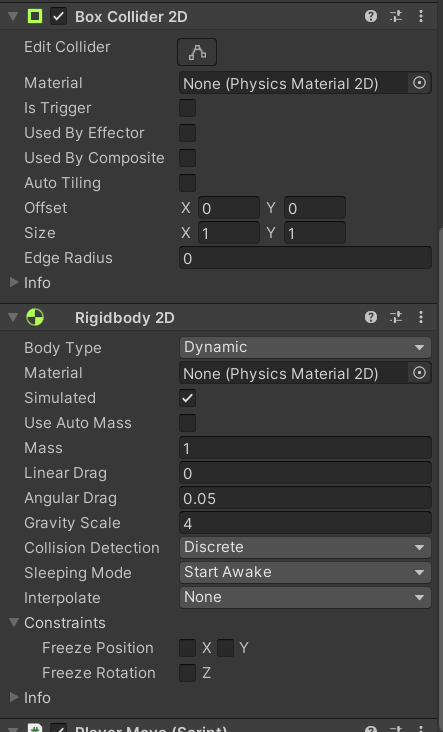
Створимо проект. Для цього у Unity Hub ми натиснемо кнопку “New project” та введемо всі дані:



У створений проект ми додали ассети відповідно до варіанту, створили сцену, та додали персонажа у вигляді білого круга:



До гравця ми додаємо такі компоненти як RigidBody 2D для того, щоб наш об’єкт мав фізичні властивості, Box Collider 2D для того, щоб наш круг міг контактувати з іншими об’єктами



Також додамо скрипт, де ми визначимо поведінку нашого об’єкту в залежності від ситуації (переміщення, стрибки). Код виглядає наступним чином:

using System.Collections;

using System.Collections.Generic;

using UnityEngine;

public class PlayerMove : MonoBehaviour

{

public Rigidbody2D rb;

public BoxCollider2D bc;

public LayerMask groundLayerMask;

public float velocity = 1000f;

public float jump\_force = 500f;

// Start is called before the first frame update

void Start()

{

rb = GetComponent<Rigidbody2D>();

bc = GetComponent<BoxCollider2D>();

}

// Update is called once per frame

void FixedUpdate()

{

float horizontal\_input = Input.GetAxis("Horizontal");

rb.velocity = new Vector2(horizontal\_input \* velocity \* Time.deltaTime, rb.velocity.y);

}

void Update()

{

if (Input.GetKeyDown(KeyCode.Space) && isGrounded(bc))

rb.AddForce(new Vector2(0, jump\_force));

}

public bool isGrounded(BoxCollider2D character)

{

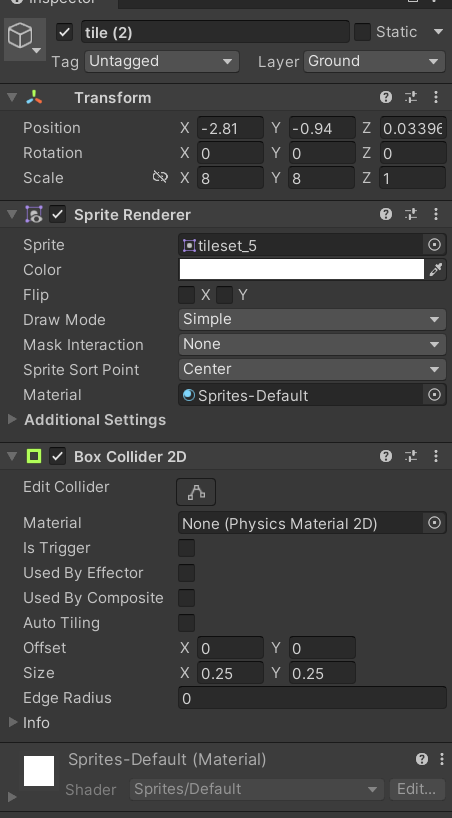
RaycastHit2D raycastHit2D = Physics2D.Raycast(character.bounds.center, Vector2.down, character.bounds.extents.y + 0.02f, groundLayerMask);

return raycastHit2D.collider != null;

}

}

Для кожної платформи та для «землі» ми також визначимо Box Collider 2D та закинемо усі ці об’єкти до шару «Ground». Це знадобиться для умови стрибка гравця.



**Висновок**

У цій лабораторній роботі ми навчилися розробленню основ проекту з використанням обраної мови програмування у обраній парадигмі. Також ми отримали основні навички розробки з використанням IDE ігрового рушія.

Посилання на GitHub із проектом: github