



МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«МИРЭА – Российский технологический университет»

РТУ МИРЭА

ОТЧЕТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ
по дисциплине «Моделирование сред и разработка приложений виртуальной
и дополненной реальности»

Практическая работа № 1

Студент группы

(подпись)

Старший
преподаватель

(подпись)

Отчет представлен

«___»____202__г.

Москва 2025 г.

1. В качестве персонального проекта была выбрана тема «3D платформер».

2. Пример применения C# скриптов продемонстрирован в листинге 1.

Листинг 1 - Код скрипта Move_Rigidbody.cs на C#

```
using UnityEngine;

public class Move_Rigidbody : MonoBehaviour
{
    public float speedForce;
    public Rigidbody cubeRigidbody;

    // Start is called once before the first execution of Update after the
    // MonoBehaviour is created
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
        Vector3 input = new Vector3(Input.GetAxis("Horizontal"), 0,
        Input.GetAxis("Vertical"));
        cubeRigidbody.AddForce(speedForce * Time.deltaTime * input,
        ForceMode.Impulse);
    }
}
```

3. Применение принципов ООП продемонстрировано в листинге 2.

Листинг 2 - Код скрипта MoveTransform.cs на C#

```
using UnityEngine;

public class MoveTransform : MonoBehaviour
{
    public float moveSpeed = 5f;
    public float rotateSpeed = 100f;

    // Start is called once before the first execution of Update after the
    MonoBehaviour is created
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
        float horizontal = Input.GetAxis("Horizontal");
        float vertical = Input.GetAxis("Vertical");
        Vector3 move = new Vector3(horizontal, 0, vertical) * moveSpeed *
Time.deltaTime;
        transform.Translate(move);
        if (Input.GetKey(KeyCode.Q))
        {
            transform.Rotate(0, -rotateSpeed * Time.deltaTime, 0);
        }
        if (Input.GetKey(KeyCode.E))
        {
            transform.Rotate(0, rotateSpeed * Time.deltaTime, 0);
        }
    }
}
```

4. Генерация объектов происходит благодаря префаб объекту Cube prefab, пустому объекту Spawner, который прикреплен к «игровому объекту» Cube и скрипту Spawner.cs. Объекты представлены на рис. 1, а скрипт в листинге 3.

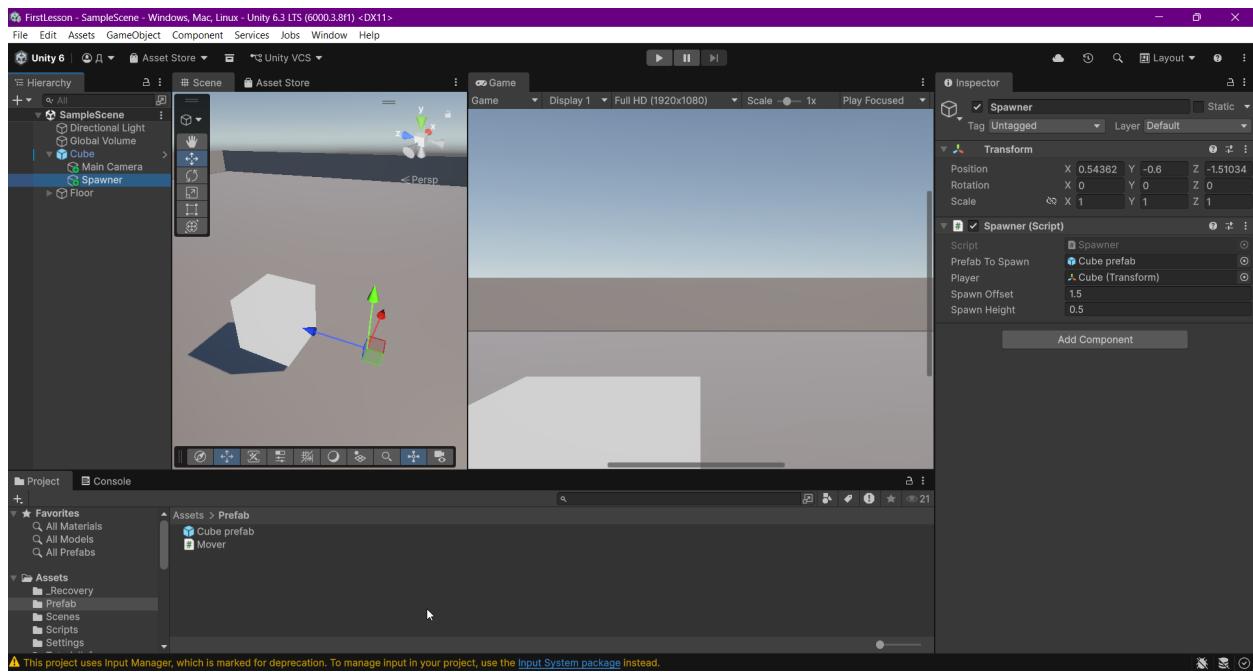


Рисунок 1 — Скриншот приложения Unity

Листинг 3 - Код скрипта Spawner.cs на C#

```
using UnityEngine;

public class Spawner : MonoBehaviour
{
    public GameObject prefabToSpawn;
    public Transform player;
    public float spawnOffset = 1.5f;
    public float spawnHeight = 0.5f;

    // Start is called once before the first execution of Update after the
    // MonoBehaviour is created
    void Start()
    {

    }

    // Update is called once per frame
    void Update()
    {
        if (Input.GetKeyDown(KeyCode.Return))
        {
            SpawnObject();
        }
    }

    void SpawnObject()
    {
        if (player == null) return;
        Vector3 spawnPos = player.position + player.forward * spawnOffset +
Vector3.up * spawnHeight;
        Instantiate(prefabToSpawn, spawnPos, player.rotation);
        UnityEngine.Debug.DrawLine(player.position, spawnPos, Color.red, 1f);
        UnityEngine.Debug.DrawRay(spawnPos, Vector3.up, Color.green, 1f);
    }
}
```

Ещё один используемый скрипт, но не описанный ранее представлен в листинге 4.

Листинг 4 - Код скрипта Mover.cs на C#

```
using UnityEngine;

public class Mover : MonoBehaviour
{
    public float speed = 100f;

    // Start is called once before the first execution of Update after the
    // MonoBehaviour is created
    void Start()
    {
        Destroy(gameObject, 0.25f);
    }

    // Update is called once per frame
    void Update()
    {
        transform.Translate(Vector3.forward * speed * Time.deltaTime);
    }
}
```