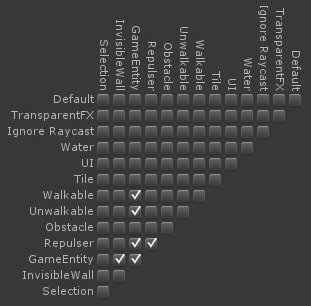
Работа #4 - Физические слои в Unity

цель работы: освоить работу с физическими слоями в Unity



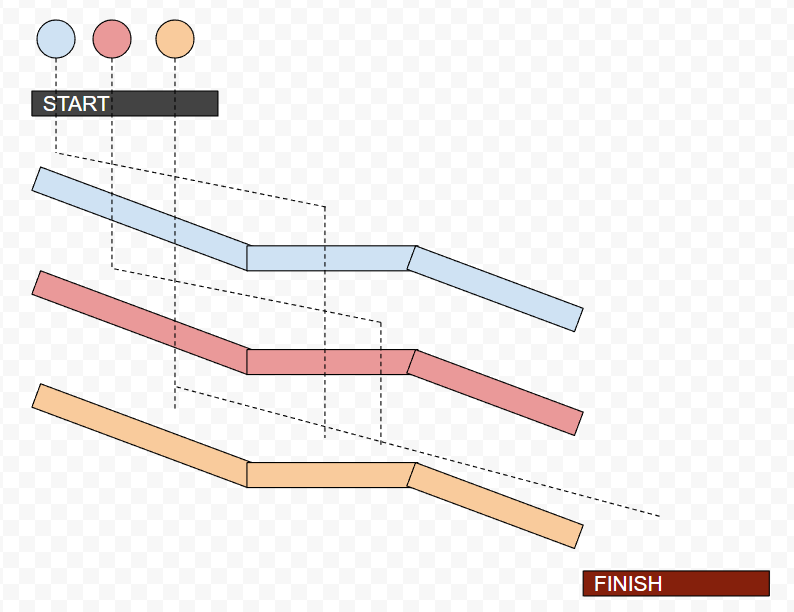
## Рекомендуемые источники для изучения:

1. [Physical Layer Unity Documentation](https://docs.unity3d.com/Manual/LayerBasedCollision.html)
2. Тема Ragdoll [в материалах лекции](https://bigdigital-gamelab.ru/visual-prog/)

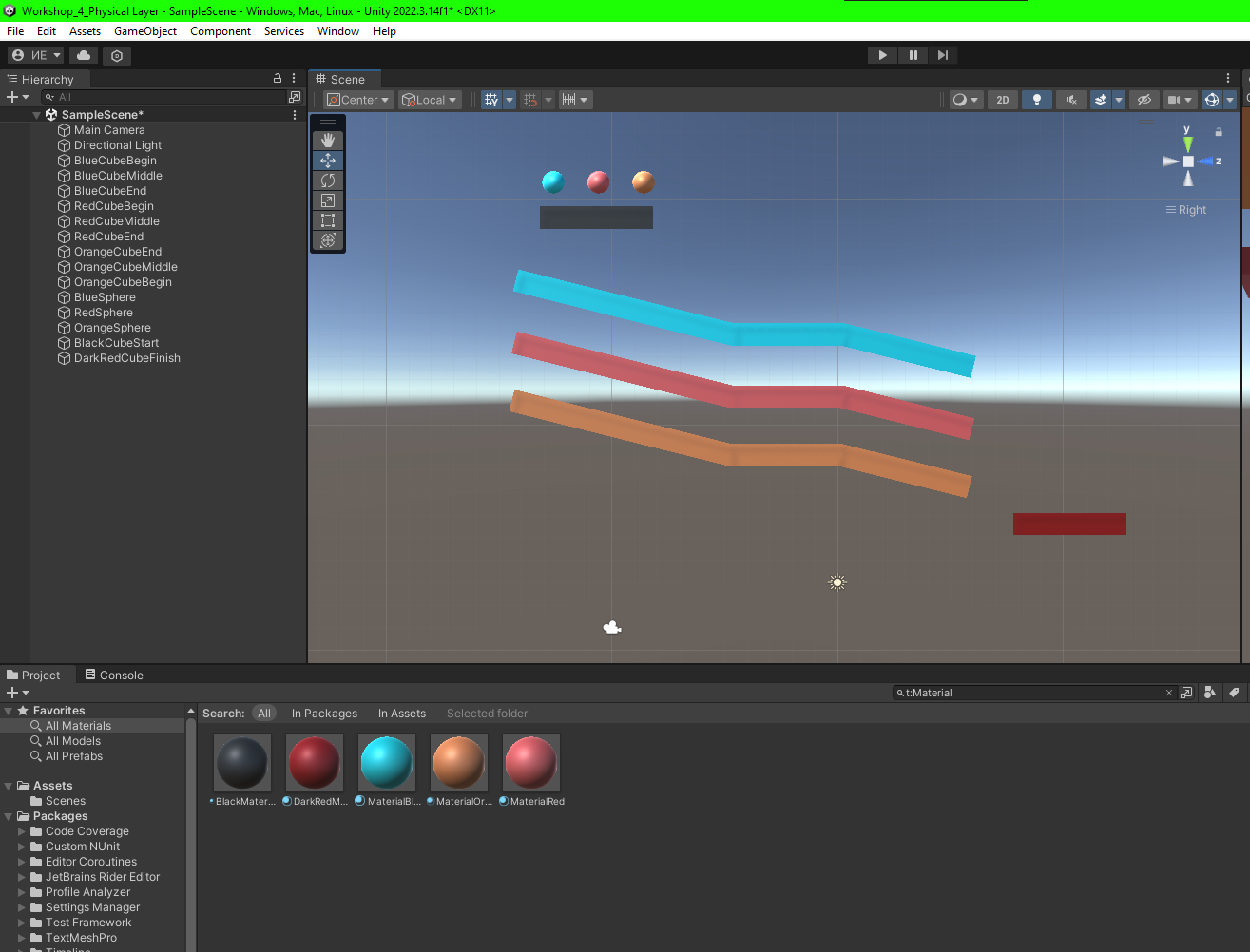
## Задания к работе

## Задание 1 Настройка физических слоев в соответствии с требуемым сценарием

1.1 Создайте сцену, которая состоит из трех сфер и системы платформ, изображенной на рисунке ниже (платформы можно создать из обычных игровых объектов GameObject Cube). Пунктирными линиями показан будущий путь прохождения широв, который будет реализован в следующих пунктах задания.



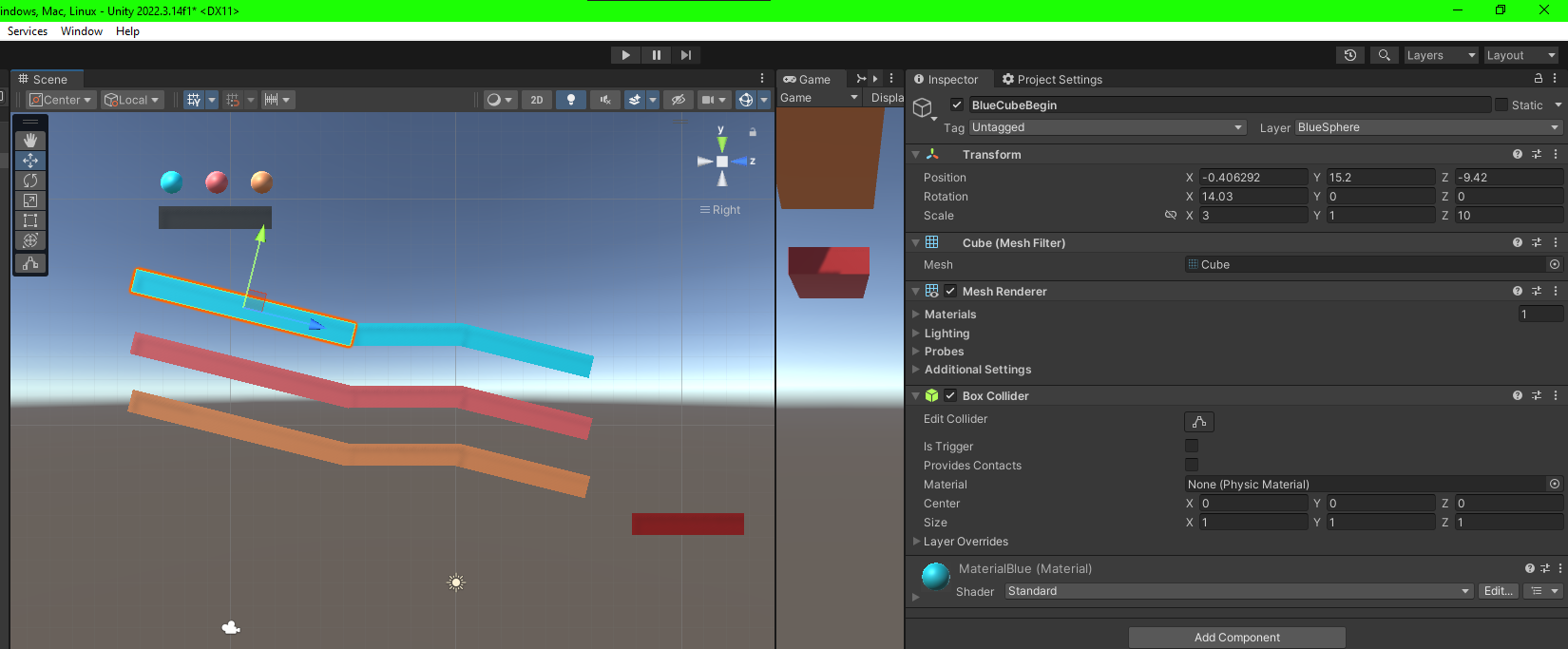
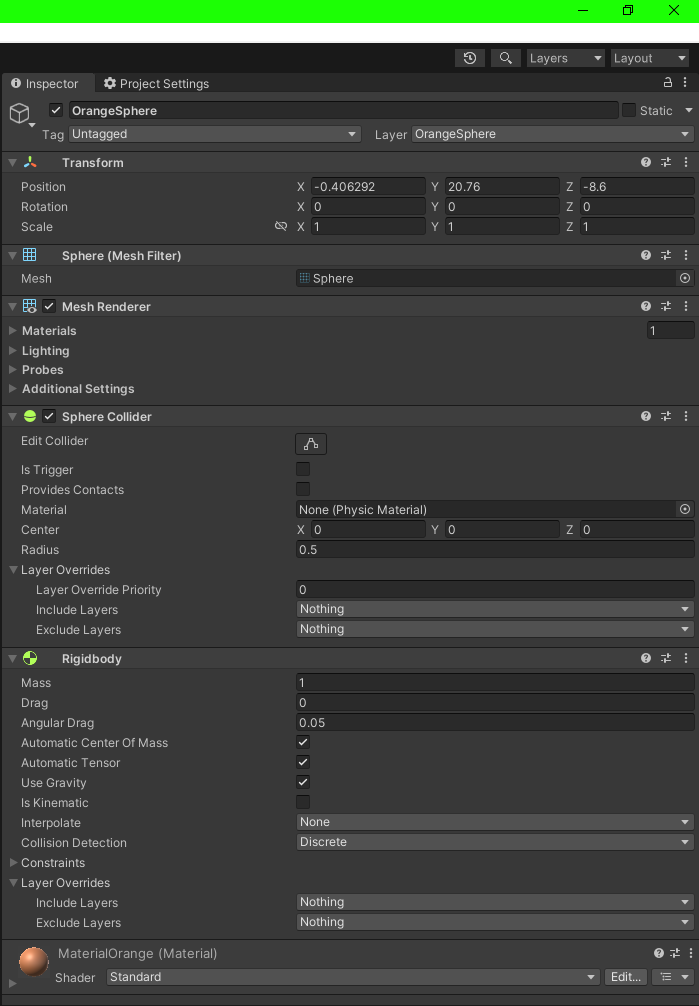
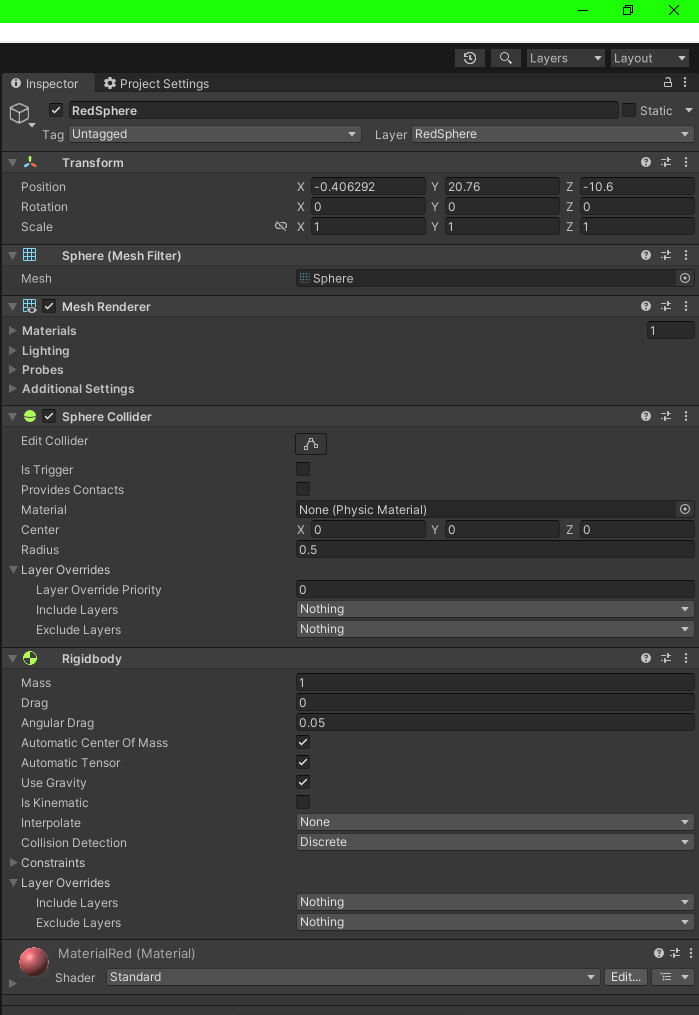
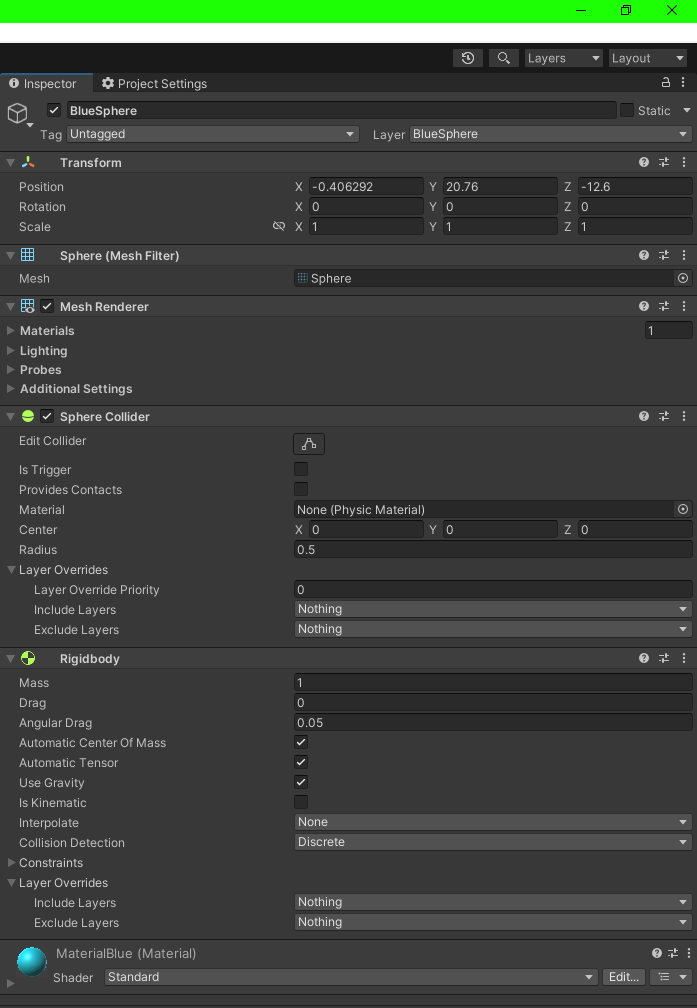
Создал сцену, показанную выше \*добавил на сцену объекты сферы: BlueSphere, RedSphere, OrangeSphere; объекты платформ: BlueCube(Begin, Middle, End соответственно), RedCube(Begin, Middle, End соответственно), OrangeCube(Begin, Middle, End соответственно); объекты платформ старта и финиша: BlackCubeStart, DarkRedCubeFinish\* и назначил на каждый объект свой материал.

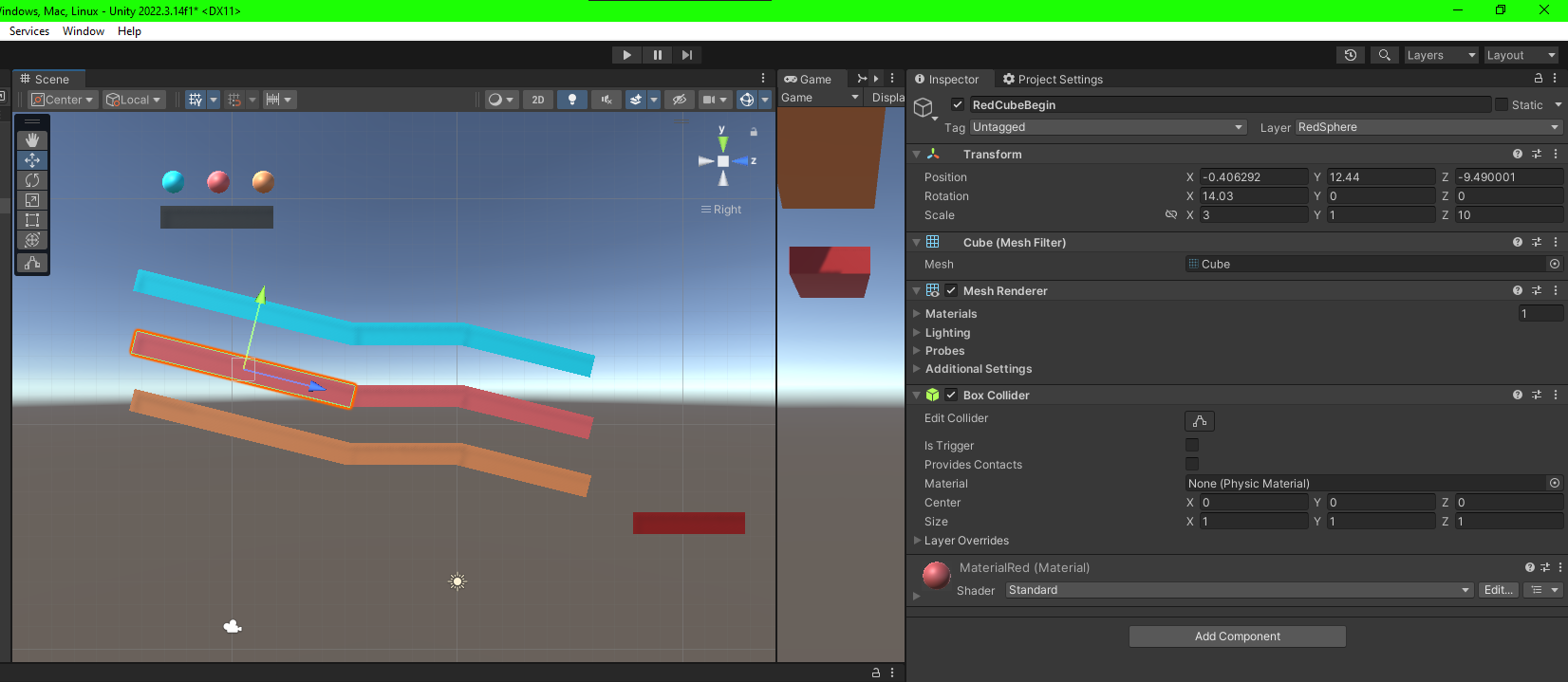


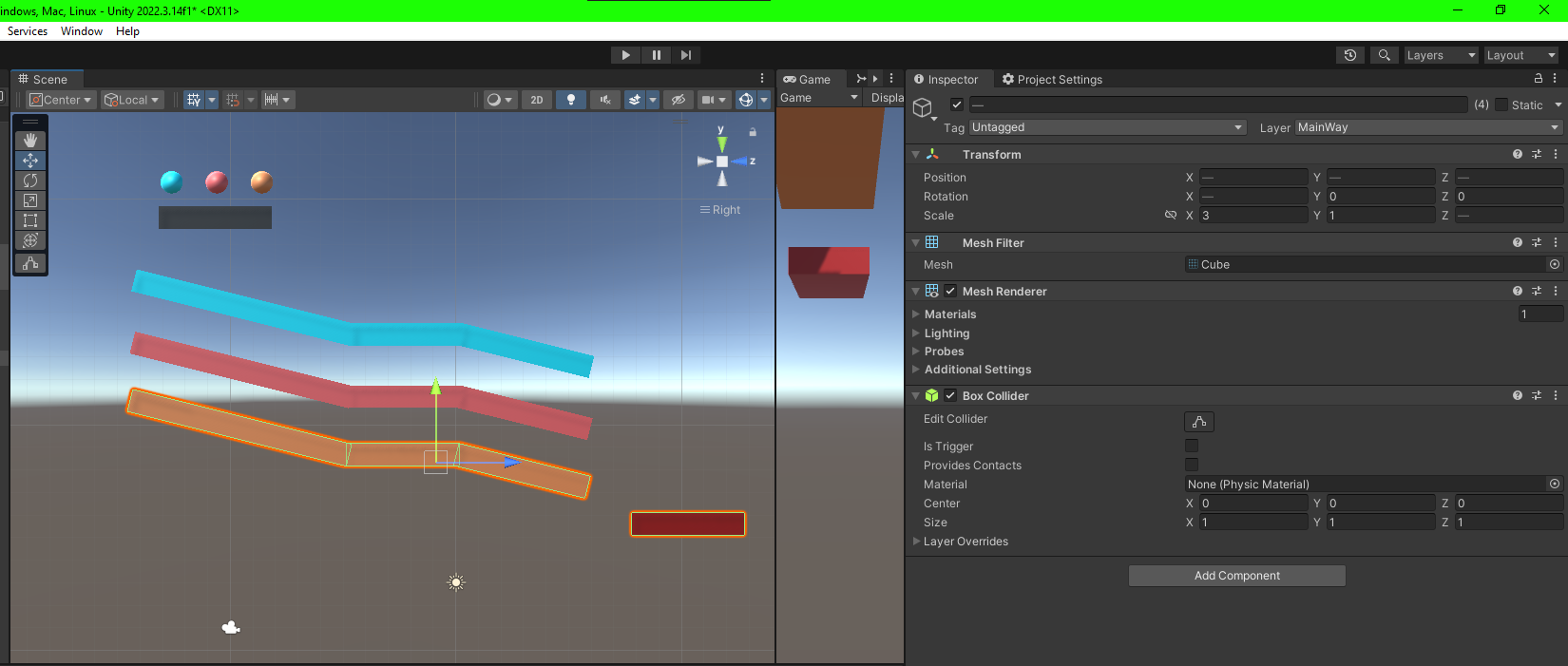
1.2 Выбор слоя для игрового объекта происходит в настройках инспектора Layer. Назначьте на созданные объекты физические слои (при необходимости создайте новые) так, чтобы путь сфер проходил через Start к Finish по указанному маршруту. Настройка матрицы коллизий производится в настройках Edit - Project Settings - Physics:

|  |  |
| --- | --- |
| Настройка слоев в окне Inspector | Настройка матрицы коллизий в Project Settings - Physics |

Добавил на объекты сфер компонент Rigidbody, чтобы подключить к ним физику \*чтобы они начали падать (Use Gravity)\*, после чего назначил каждому объекту свой определённый слой (как сферам, так и платформам) и настроил матрицу коллизий для взаимодействия слоёв.







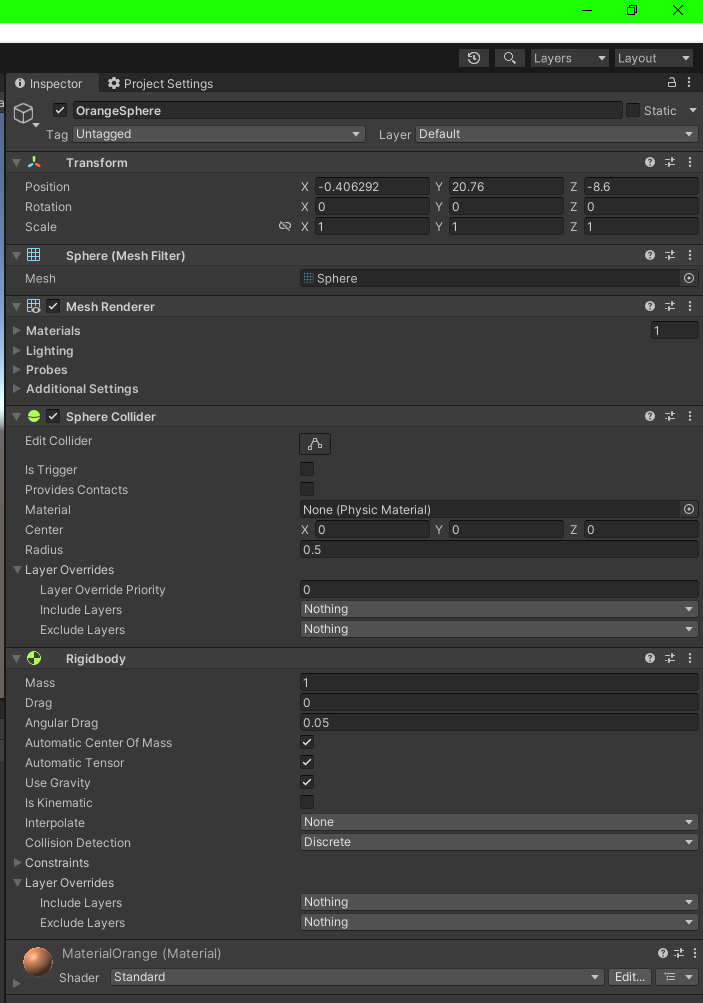


1.3 Опишите корректную настройку матрицы коллизий. Какие слои вы создали дополнительно? Является ли предложенное вами решение оптимальным?

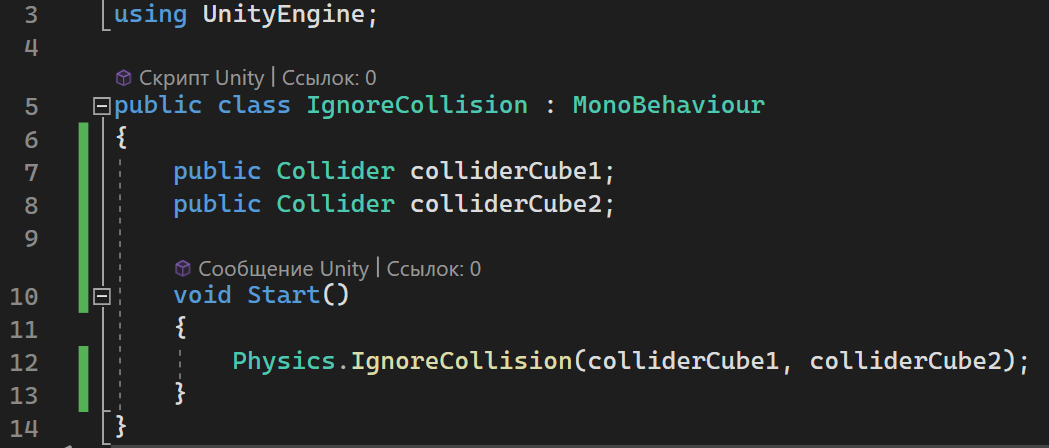
|  |
| --- |
| **Создал 4 слоя и использовал 1 предустановленный (Default): один (Default) для платформы BlackCubeStart и платформ, по которым не будут катиться сферы; три (BlueSphere, RedSphere, OrangeSphere) для каждой сферы и платформы, по которой будет катиться своя сфера; ещё один (MainWay) для последних платформ, по которым будут катиться все сферы (плюс платформа DarkRedCubeFinish).**  **Создал 4 дополнительных слоя: 3 слоя для каждой сферы и платформы, по которой она катится; 1 слой для платформ, по которым катятся все сферы.**  **Предложенное мной решение является оптимальным, так как “Чем меньше слоёв нужно обсчитывать на предмет коллизий, тем выше производительность”, в предложенном мною решении на предмет коллизий приходится обсчитывать всего 4 слоя.** |

1.4 Модифицируйте настройку слоев таким образом, чтобы до финиша доходило только две сферы.

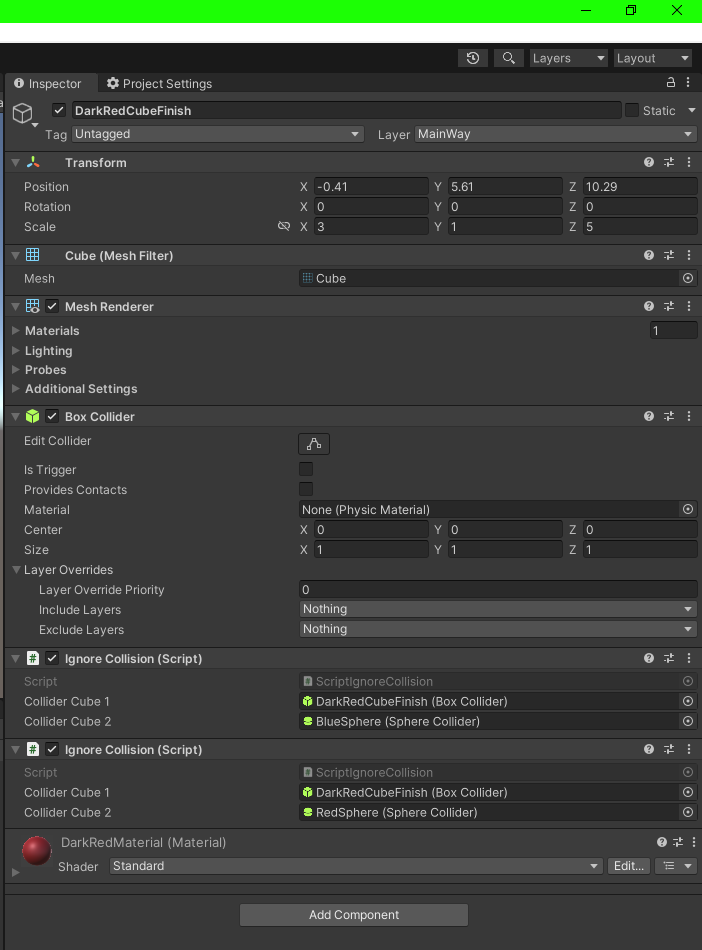
Модифицировал настройку слоёв таким образом, чтобы до финиша доходило только две сферы: для оранжевой сферы назначил слой Default, который не контактирует со всеми остальными слоями (OrangeSphere просто падает вниз через все платформы).



1.5 Используя скрипт, приведенный ниже, отключите возможность взаимодействия коллайдеров двух оставшихся сфер на финише:

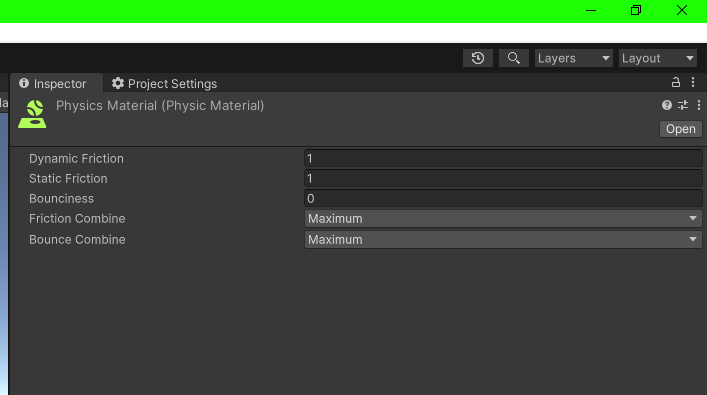


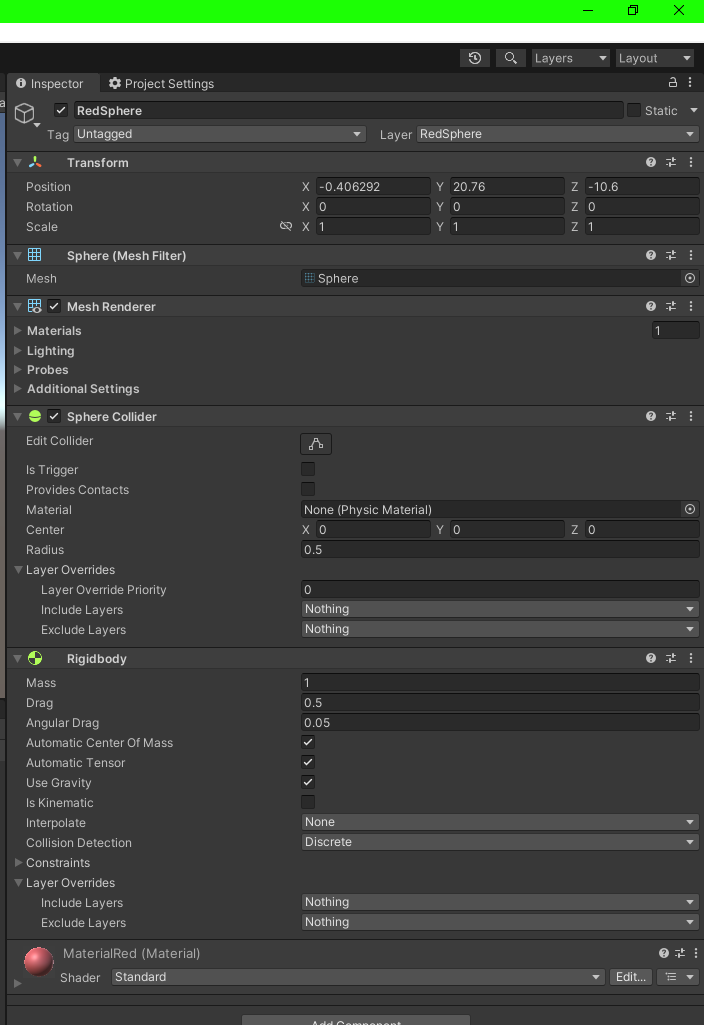
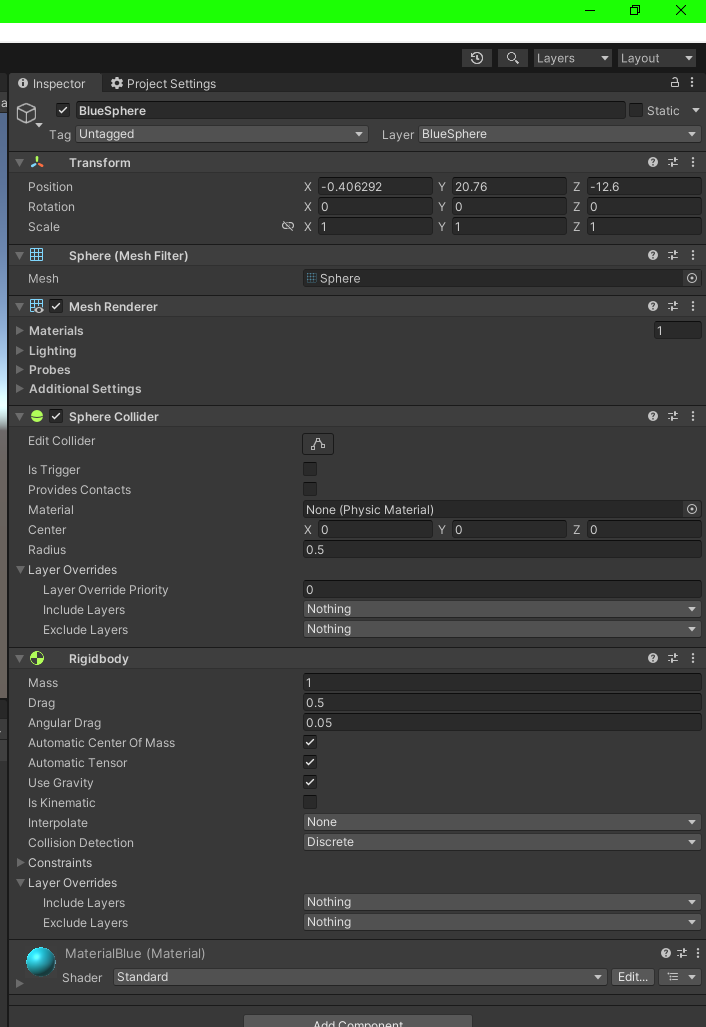
Используя скрипт, указанный выше, отключил возможность взаимодействия коллайдеров двух оставшихся сфер на финише, назначив данный скрипт два раза на платформу финиш (DarkRedCubeFinish), перечислив коллайдеры, которые не должны взаимодействовать друг с другом.



1.6 Модифицируйте матрицу коллизий таким образом, чтобы до финиша доходили только две сферы. Сферы должны останавливаться на финише и не должны иметь возможность взаимодействия друг с другом. При необходимости модифицируйте параметры материала коллайдера на финише. Опишите, как вы реализовали данное решение:

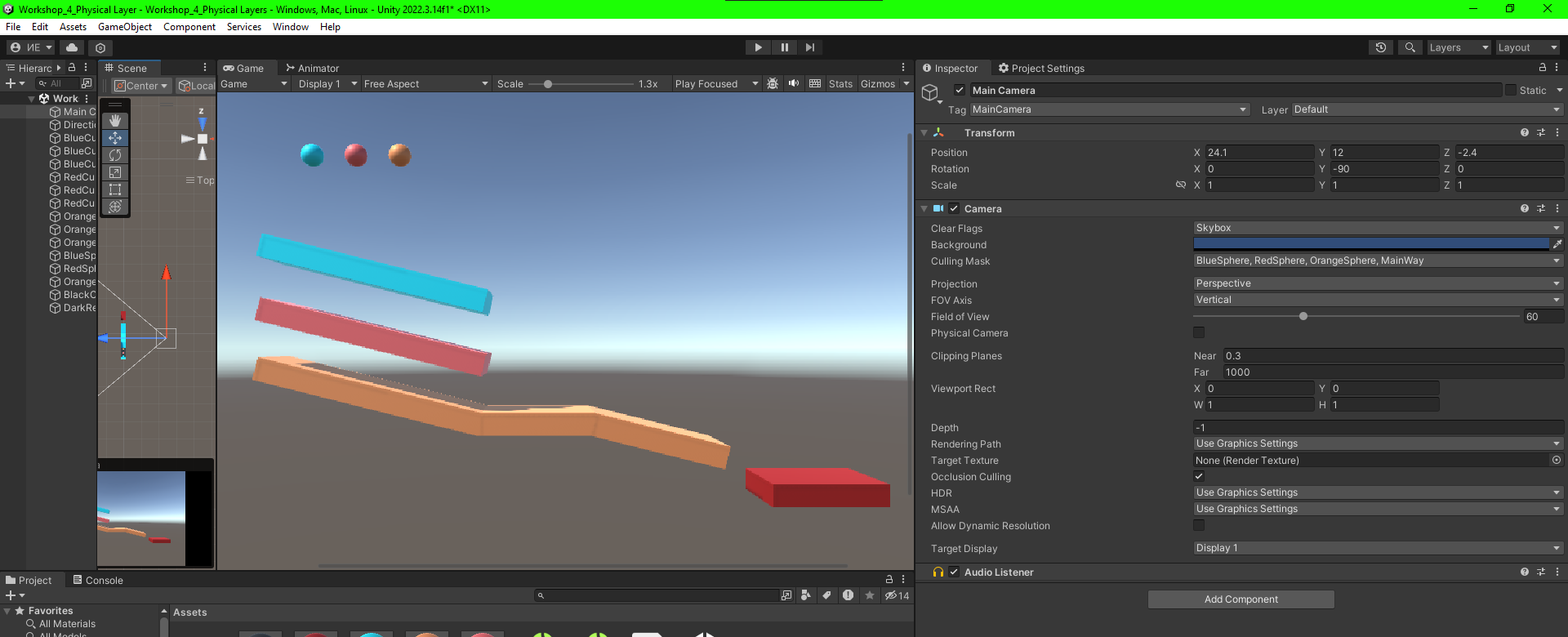
|  |
| --- |
| **В голове возникло два решения:**  **1. Изменить коллайдер платформы DarkRedCubeFinish.**  **2. Увеличить сопротивление среды для объектов BlueSphere и RedSphere.**  **Первый вариант не стал реализовывать, потому что посчитал его слишком лёгким, та и про изменение самого коллайдера у платформы в задании не сказано; Решил реализовать второй вариант (да, про изменение параметров сопротивления среды у объектов тоже ничего не сказано, но других вариантов придумать не смог).**  **У каждой сферы поставил значение сопротивления среды, равное 0.5; Создал физический материал коллайдера и применил его к платформе финиш (в параметрах материала указал Dynamic Friction = 1, Static Friction = 1, Bounciness = 0, Friction Combine = Maximum, Bounce Combine = Multiply);** |





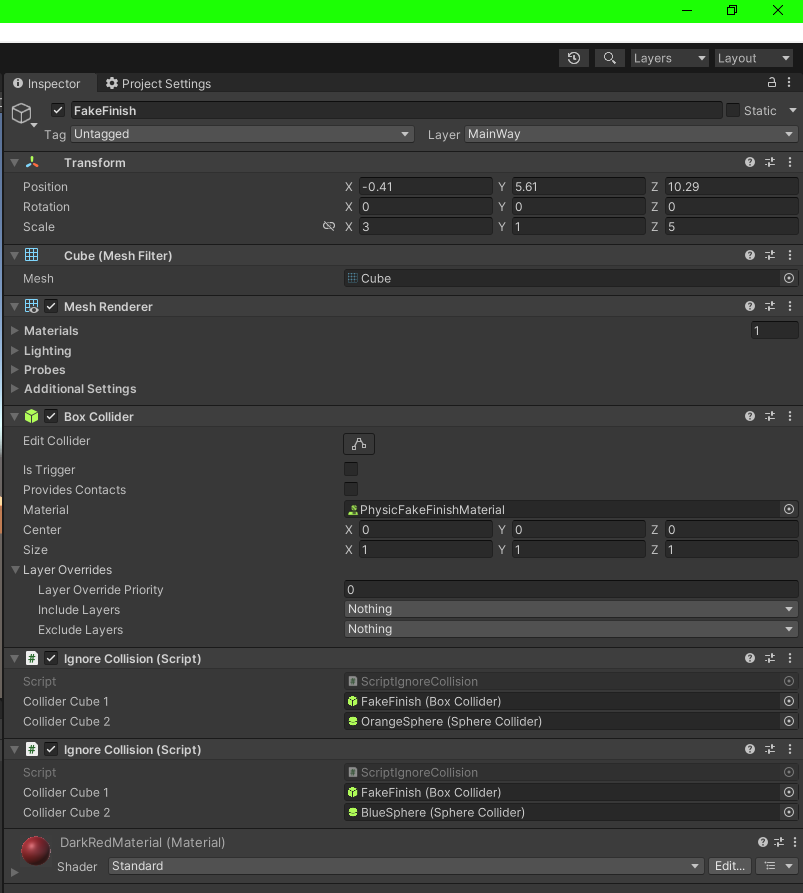
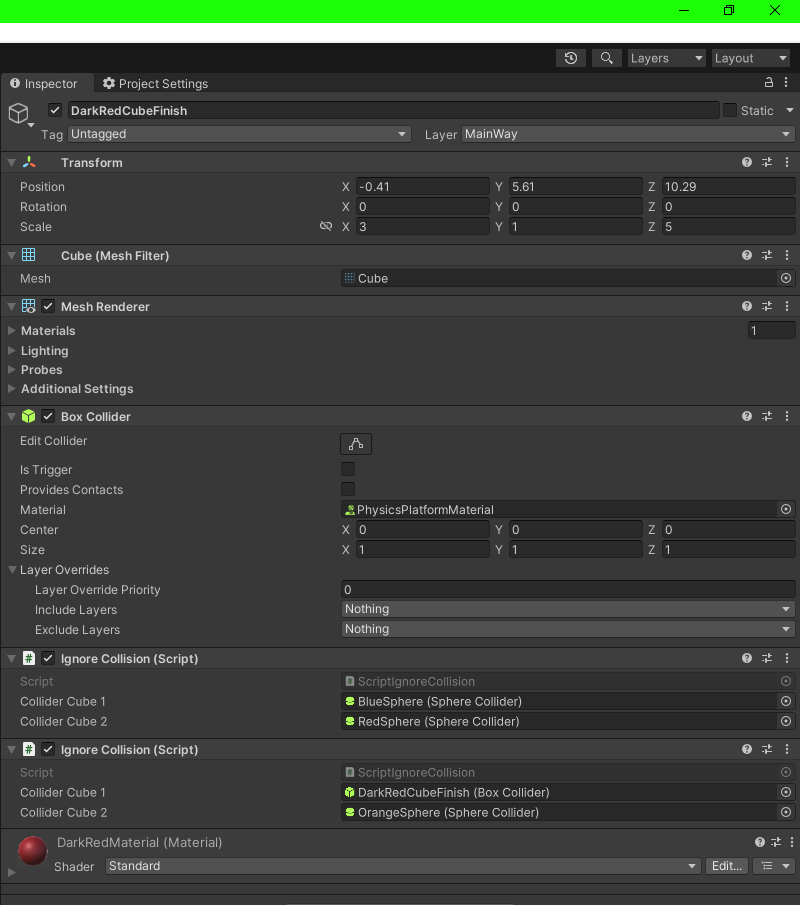
1.7 В свойстве Culling Mask объекта Main Camera есть возможность рендерить лишь выбранные слои. Настройте камеру таким образом, чтобы игрок в окне Game видел единственный путь на сцене, по которому сферы достигают финиша. О существовании других путей “игрок” не должен знать.

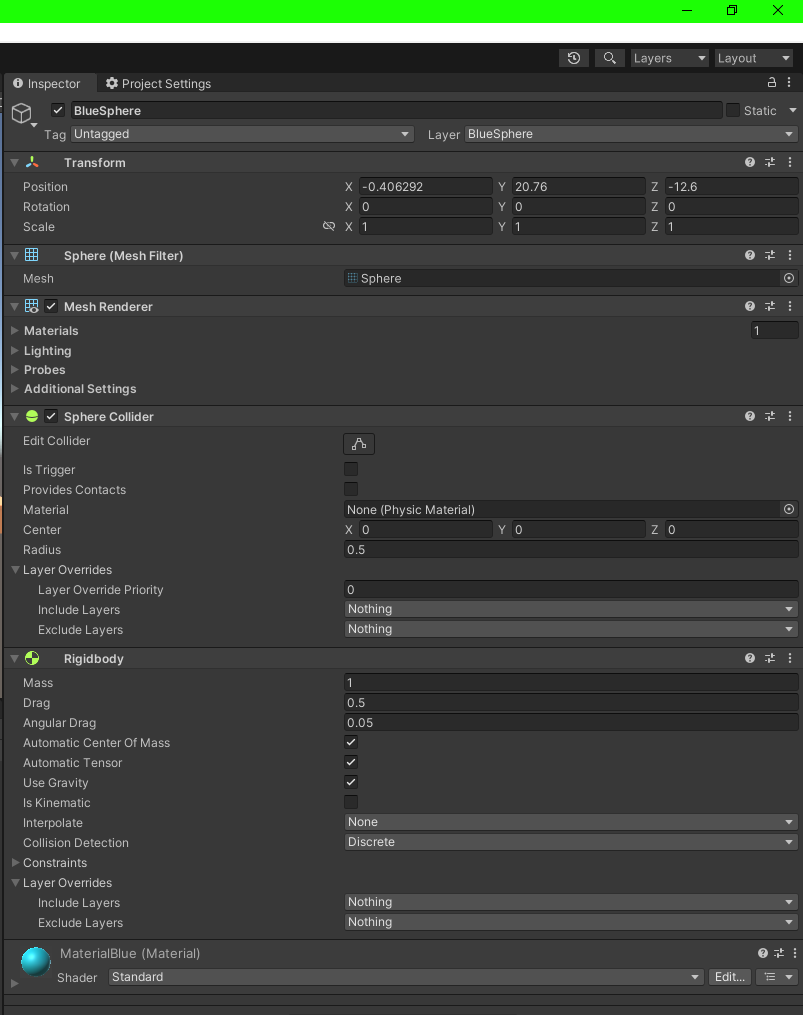
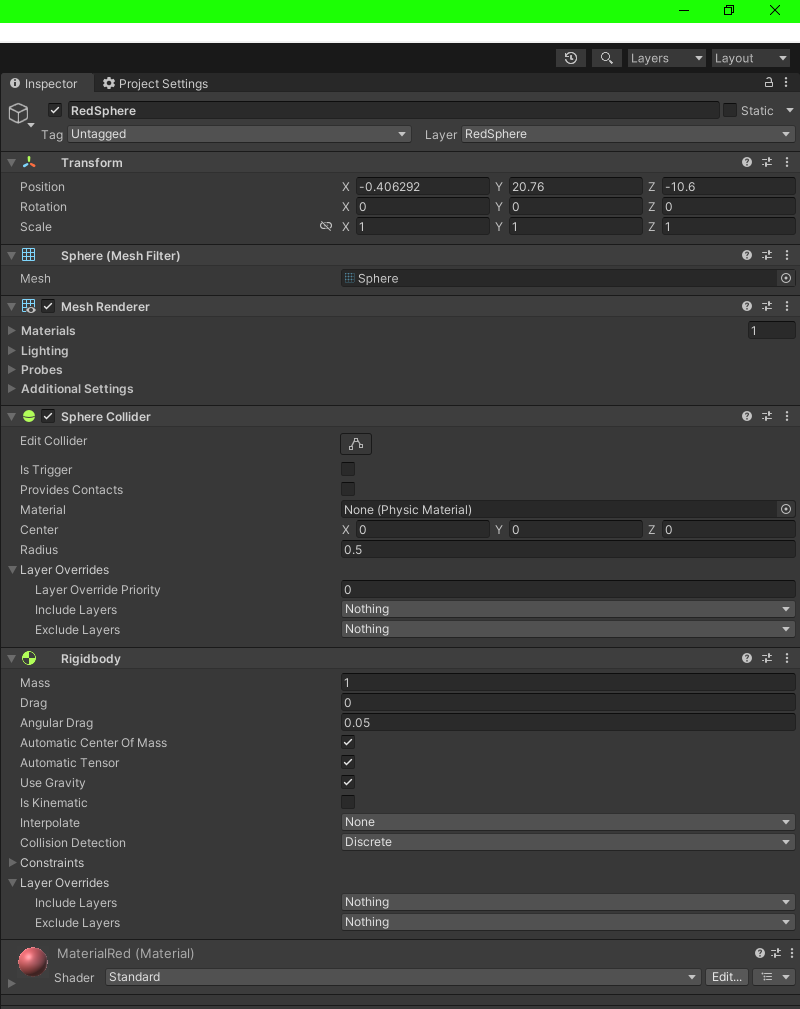
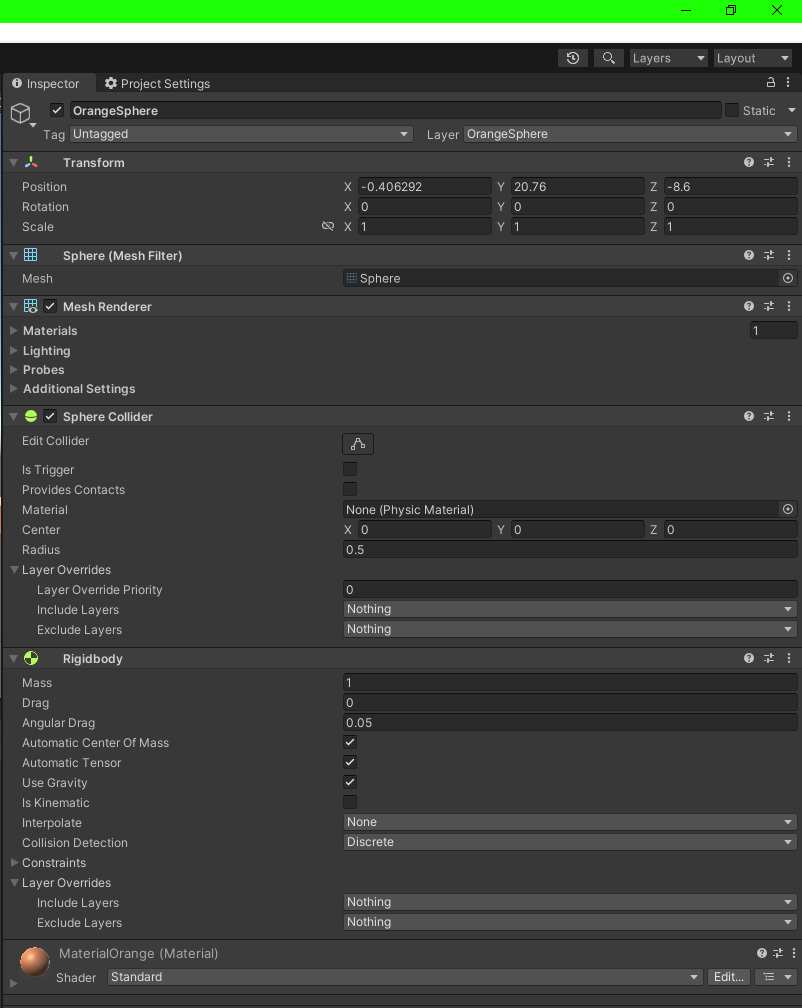
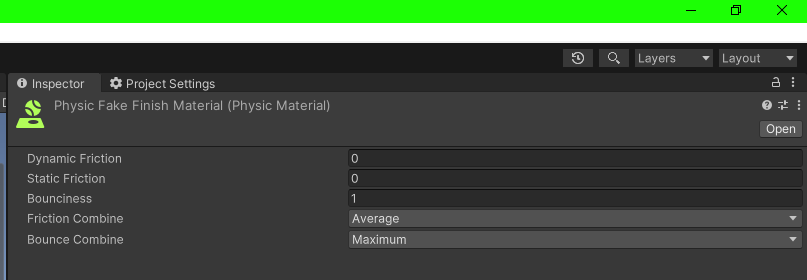
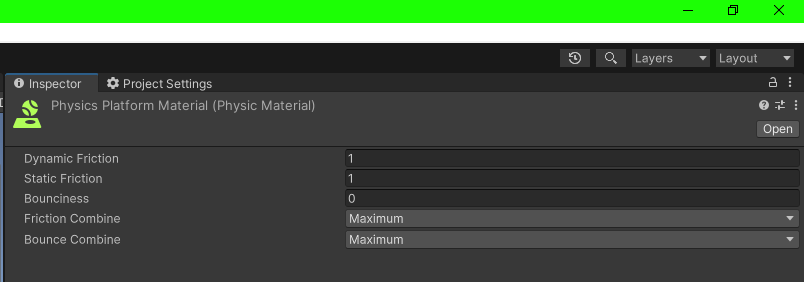
В свойстве Culling Mask объекта main Camera перечислил слои, которые должны отображаться во время проигрывания сцены (BlueSphere, RedSphere, OrangeSphere, MainWay).



1.8 Модифицируйте сцену таким образом, чтобы при достижении финиша одна из сфер отскакивала от финиша, а вторая оставалась на нём. Третья сфера должно проходить сквозь объект FINISH. Опишите, как вы реализовали данное решение:

|  |
| --- |
| **Данное решение реализовал следующим образом:**  **Создал дополнительную платформу FakeFinish с такими же размерами и на том же месте, что и платформу финиша DarkRedCubeFinish; назначил на неё новый материал коллайдера PhysicFakeFinishMaterial со следующими настойками (Dynamic Friction = 0, Static Friction = 0, Bounciness = 1, Friction Combine = Average, Bounce Combine = Multiply), чтобы добиться отскока от данной платформы; применил на неё скрипт ScriptIgnoreCollision 2 раза: первый раз указал коллайдеры платформы и оранжевой сферы, чтобы она падала сквозь платформу; второй раз указал коллайдеры платформы и синей сферы, чтобы сфера не отскакивала от платформы, попав на неё.**  **Объект DarkRedCubeFinish: к коллайдеру применил материал PhysicsPlatformMaterial, с помощью которого синяя сфера будет замедляться и останавливаться на платформе (настройки материала коллайдера: Dynamic Friction = 1, Static Friction = 1, Bounciness = 0, Friction Combine = Maximum, Bounce Combine = Multiply); применил скрипт ScriptIgnoreCollision два раза: первый раз указал коллайдеры синей и красной сферы, чтобы они не имели возможности контактировать друг с другом; второй раз указал коллайдеры платформы и оранжевой сферы, чтобы сфера проходила сквозь платформу и падала вниз.**  **Объект BlueSphere: двигается по своему пути, после чего останавливается на платформе (в компоненте Rigidbody изменил параметр сопротивления среды Drag на значение 0,5 , после чего останавливается на платформу финиша DarkRedCubeFinish.**  **Объект RedSphere: сфера двигается по своему маршруту, после чего отскакивает от платформы финиша FakeFinish.**  **Объект OrangeSphere: сфера движется по своему маршруту, после чего проходит сквозь платформы DarkRedCubeFinish и FakeFinish.** |

1.12 Напишите развернутый вывод по проделанной работе. Как знания о матрице слоев и возможной настройки коллизий помогут в создании интерактивных приложений, игр и симуляторов?

|  |
| --- |
| **В ходе данной лабораторной работы были выполненные поставленные задания и изучены основные принципы работы с механикой слоёв в Unity; Потренировался с создании своих собственных слоёв, а также в использовании уже имеющихся; Поработал с настройкой слоёв, а также матрицы коллизий; В настройках камеры поигрался с параметром, ответственным за отрисовку назначенных слоёв.**  **Полученные знания будут полезны при создании и настройке сцен окружения, а также взаимодействия с ними через параметры настройки слоёв и матрицы коллизий; В играх можно будет таким образом ставить невидимые преграды перед игроком (не очень приветствуется, но такое можно сделать) с помощью настройки отображения слоёв в параметрах камеры (Culling Mask); Также можно будет прикрутить механику отключения взаимодействия коллизий у объектов при их пересечении или выполнении какого-то другого условия (используя скрипт).** |