

Лабораторная работа №2: Анимация в Unity

Цель:

Целью данной лабораторной работы является знакомство с процессом импорта анимированных объектов в среде Unity.

Справка:

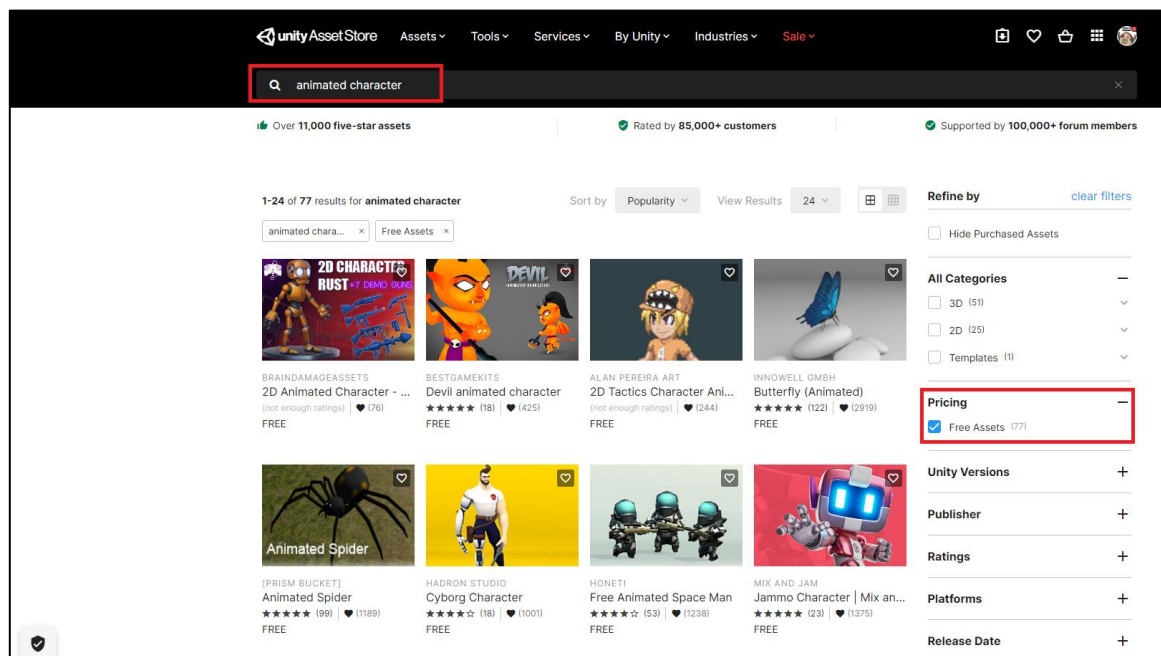
Анимированные объекты можно получить с сайта Mixamo: <https://www.mixamo.com/>

Либо из Asset Store: <https://assetstore.unity.com/>

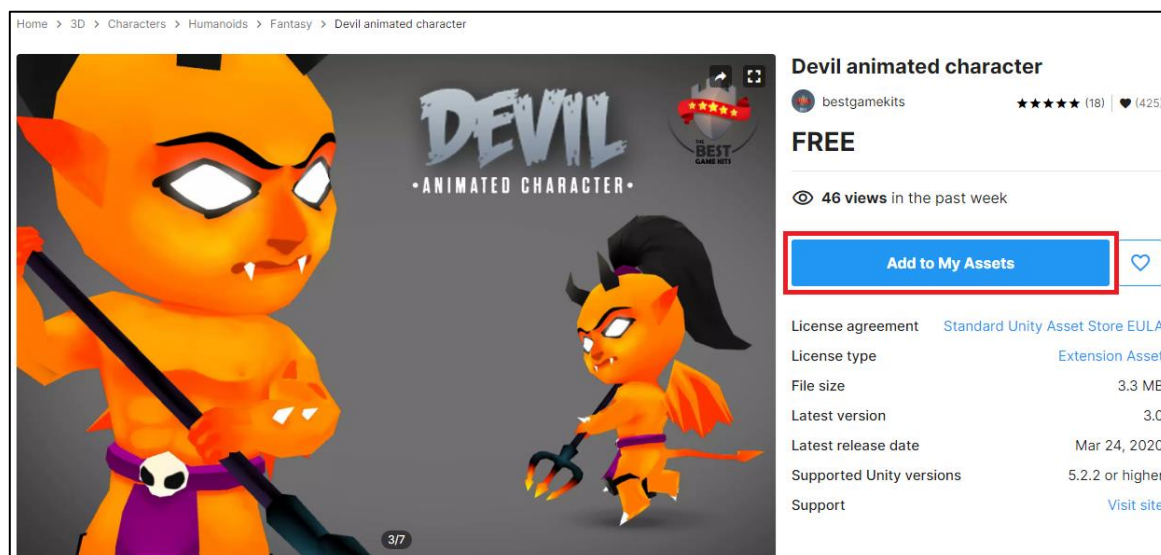
Импорт анимации

Чтобы добавить анимации из **Unity Asset Store**, перейдите по адресу: <https://assetstore.unity.com/>

Введите в строку поиска запрос и выберите диапазон цен:



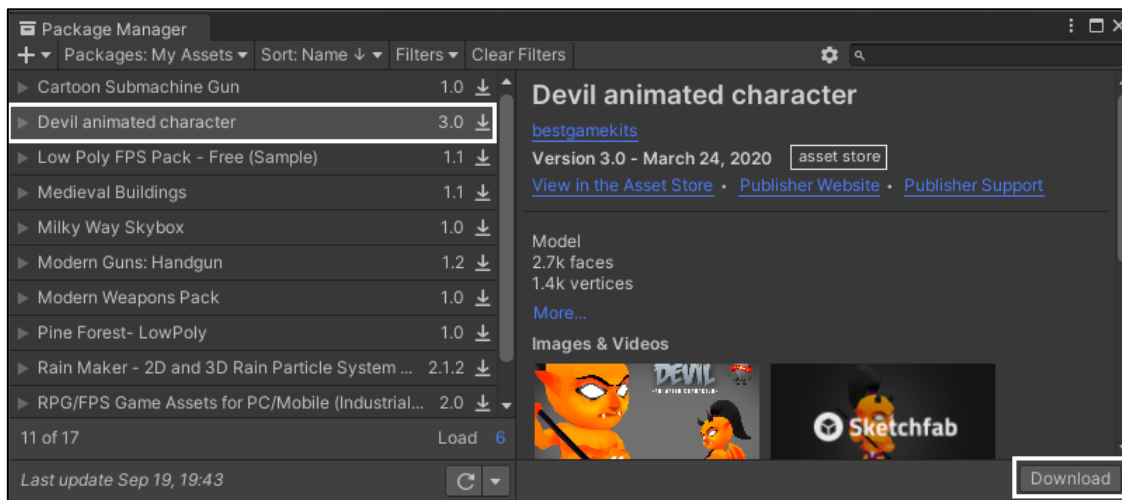
Перейдите на страницу желаемой модели и нажмите кнопку Add to My Assets:



В появившейся форме нажмите Open in Unity:



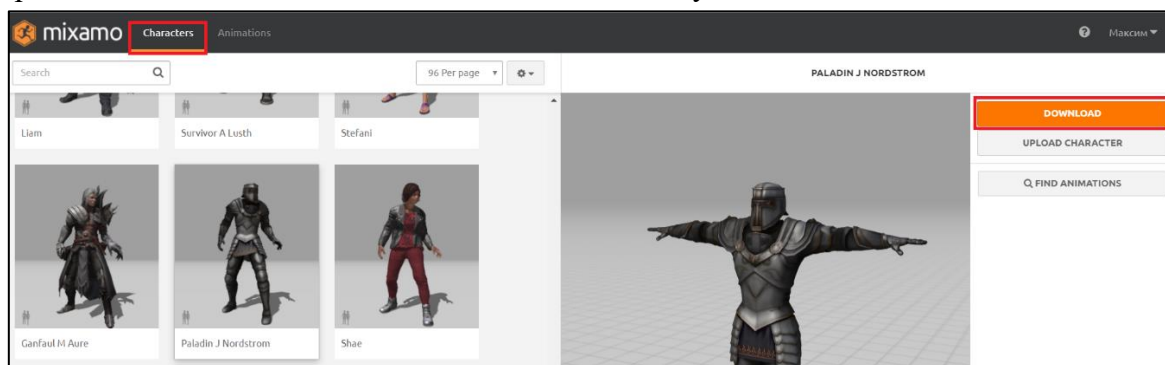
В открывшемся менеджере пакетов, выберите загруженную модель и нажмите Download:



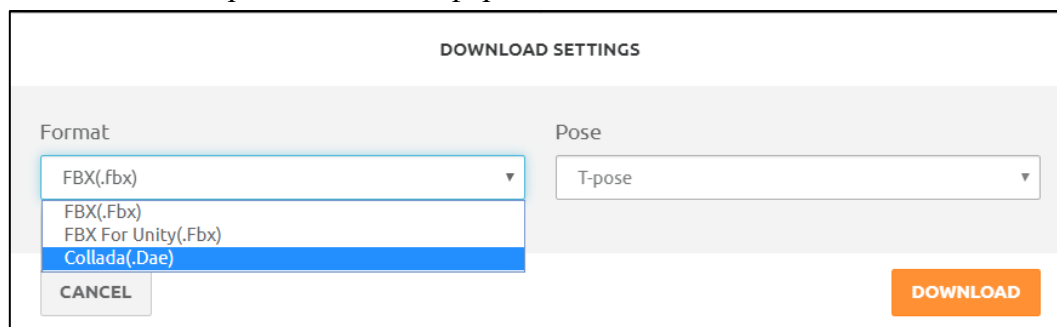
В завершении, кликните в появившемся окне Import.

Альтернативным вариантом, является добавление моделей с сайта **mixamo**:

1. Зарегистрируйтесь на сайте, или воспользуйтесь для входа аккаунтом Google/Facebook.
2. Выберите модель в списке Characters и нажмите кнопку Download:



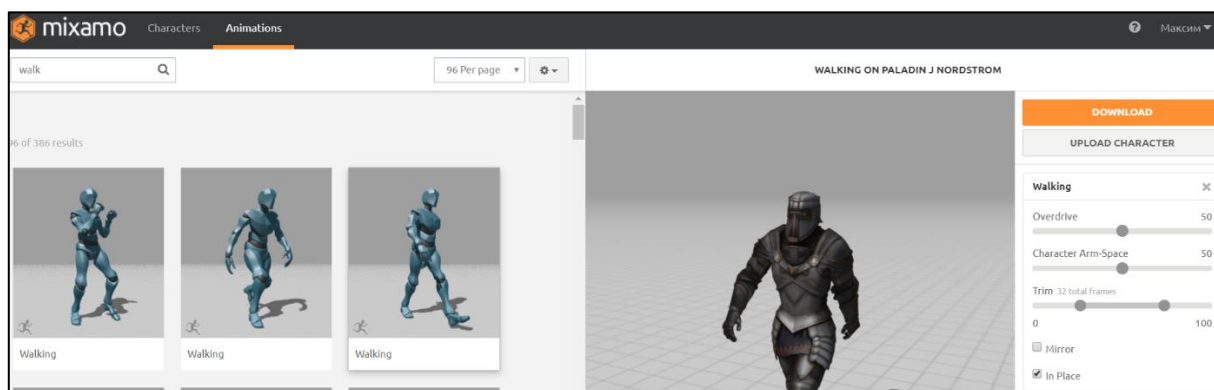
3. В появившемся окне, выберите формат модели FBX For Unity и скачайте модель.
4. Если выбранная модель имеет текстуры, то нажмите Download ещё раз, а затем, в появившемся окне, выберите в качестве формата Collada:



Распакуйте скачанный архив в директории с моделью. Итоговый вид:

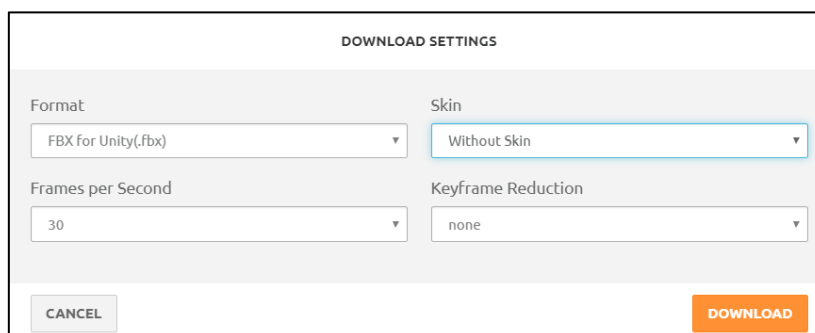
textures	09.06.2019 11:53	Папка с файлами	
paladin_j_nordstrom.fbx	09.06.2019 11:53	3D Object	10 728 KБ

5. Перейдите на вкладку Animations и выберите желаемую анимацию.



Примечание: Слева сверху находится поле для поиска анимации по названию. Справа находится окно просмотра анимации для выбранного персонажа и панель настроек параметров анимации. Для анимации перемещения имеет смысл отметить поле In Place, чтобы исключить перемещение модели.

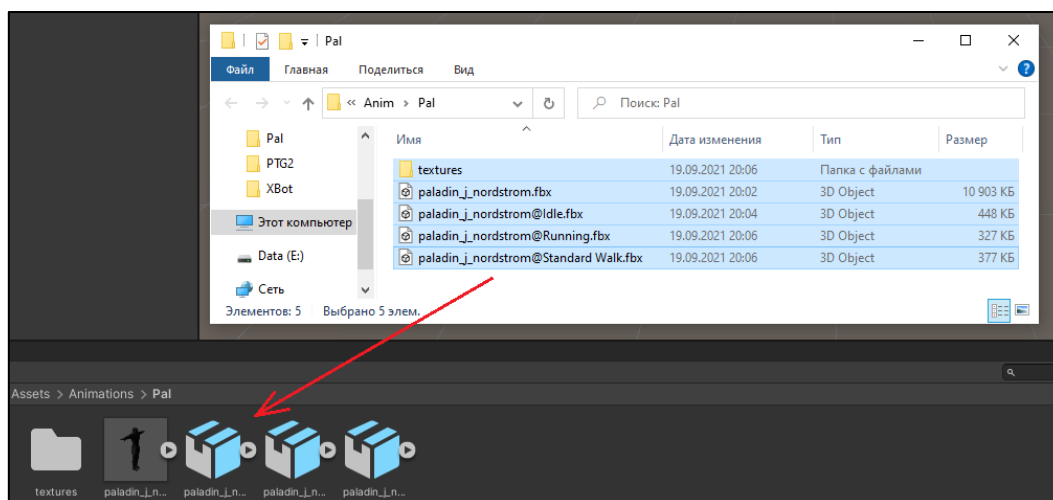
1. После выбора анимации, нажмите кнопку Download и выберите следующие параметры:



2. После загрузки модели, текстур и анимаций, директория будет выглядеть следующим образом:

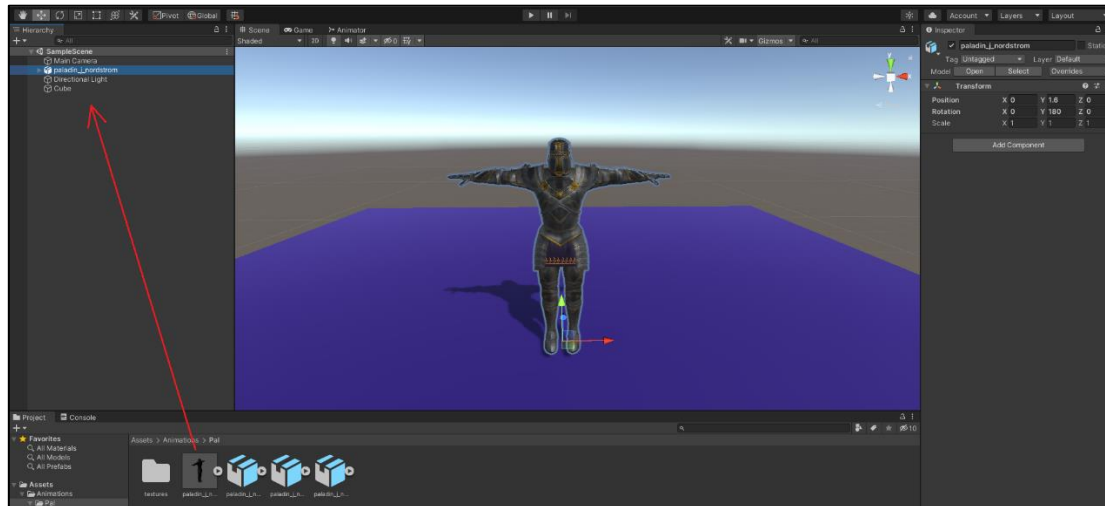
textures	09.06.2019 11:53	Папка с файлами	
paladin_j_nordstrom.fbx	09.06.2019 11:53	3D Object	10 728 КБ
paladin_j_nordstrom@Cross Punch.fbx	09.06.2019 12:03	3D Object	355 КБ
paladin_j_nordstrom@Idle.fbx	09.06.2019 12:02	3D Object	595 КБ
paladin_j_nordstrom@Jumping.fbx	09.06.2019 12:04	3D Object	457 КБ
paladin_j_nordstrom@Running.fbx	09.06.2019 12:03	3D Object	276 КБ
paladin_j_nordstrom@Walking.fbx	09.06.2019 12:02	3D Object	307 КБ

3. Для того, чтобы импортировать модель и анимации, создайте соответствующую папку в вашем проекте и перетащите туда все необходимые файлы:

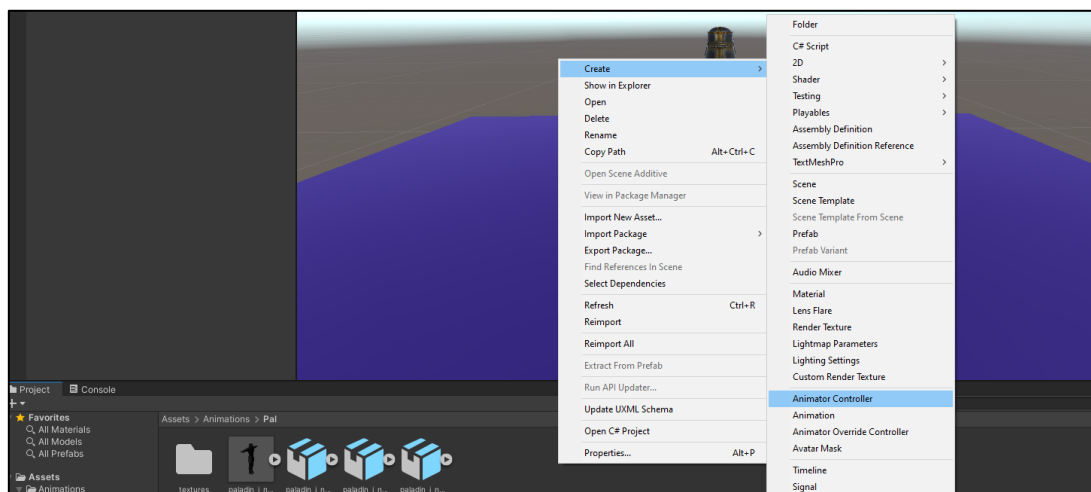


Добавление модели и назначение анимаций

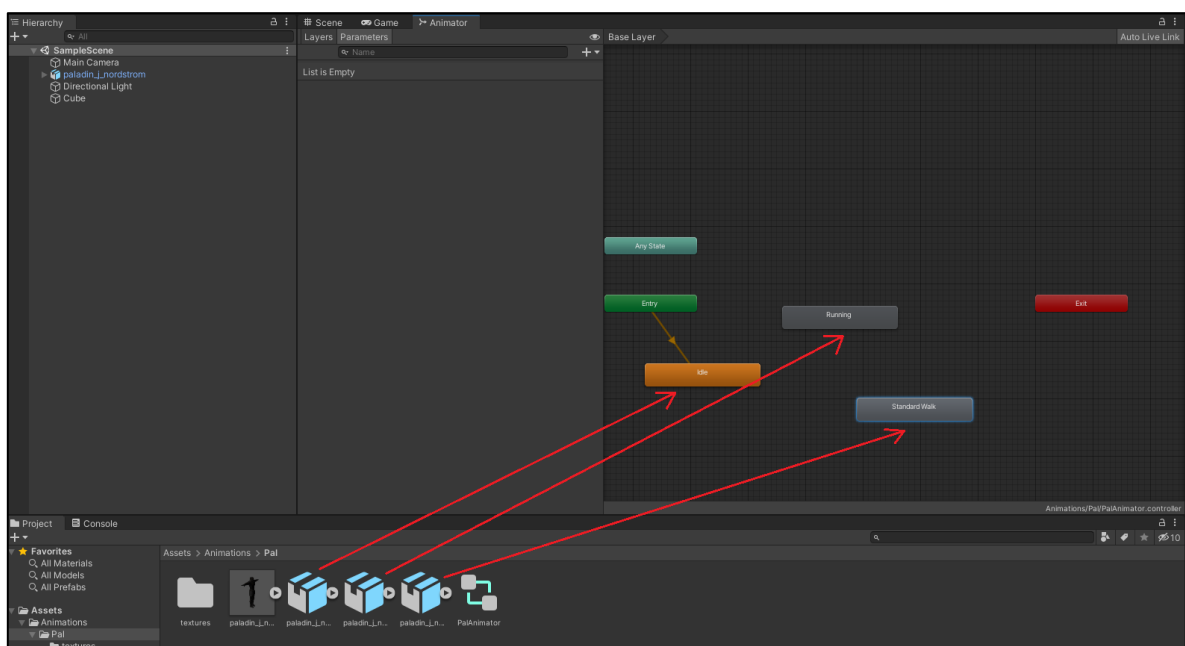
Загруженная модель является префабом и добавить в сцену её можно так же, как и любой префаб:



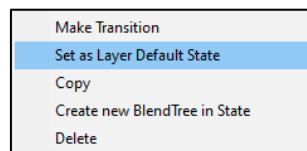
Для добавления анимаций к модели, необходимо создать контроллер анимаций (аниматор):



Для начала работы с аниматором, кликните по нему дважды, а затем, перетащите анимации в открывшееся поле:

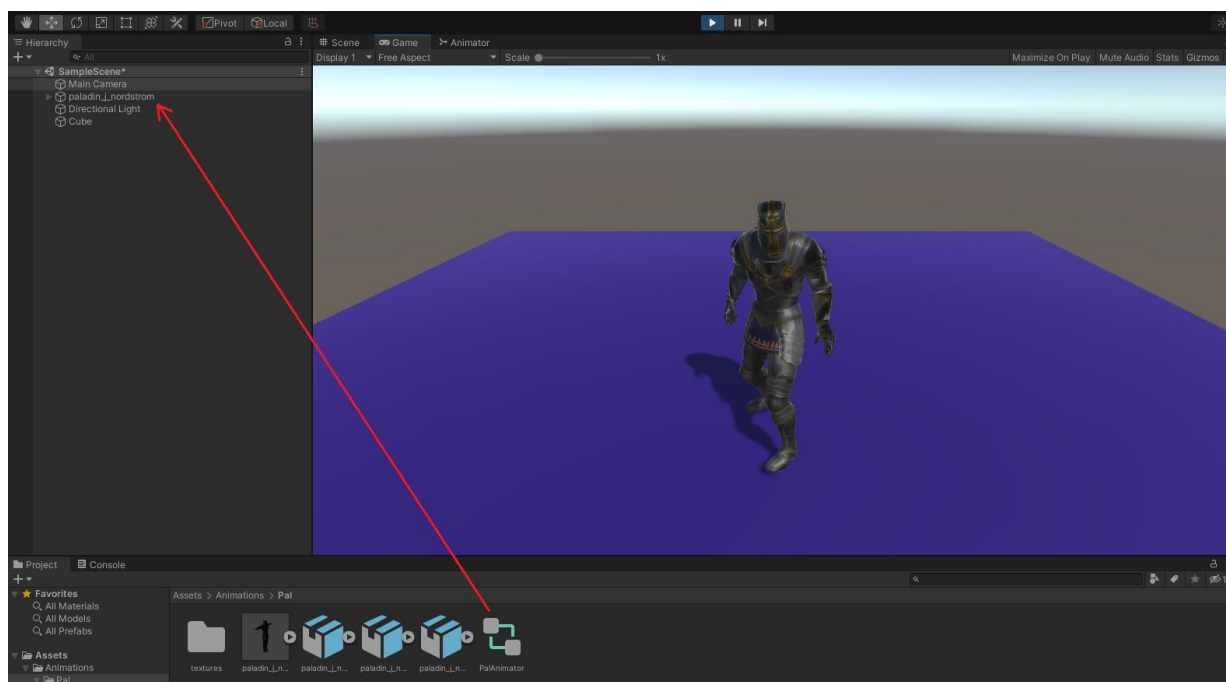


Что бы установить анимацию в качестве анимации по умолчанию, кликните по ней правой кнопкой мыши и выберите Set as Layer Default State:



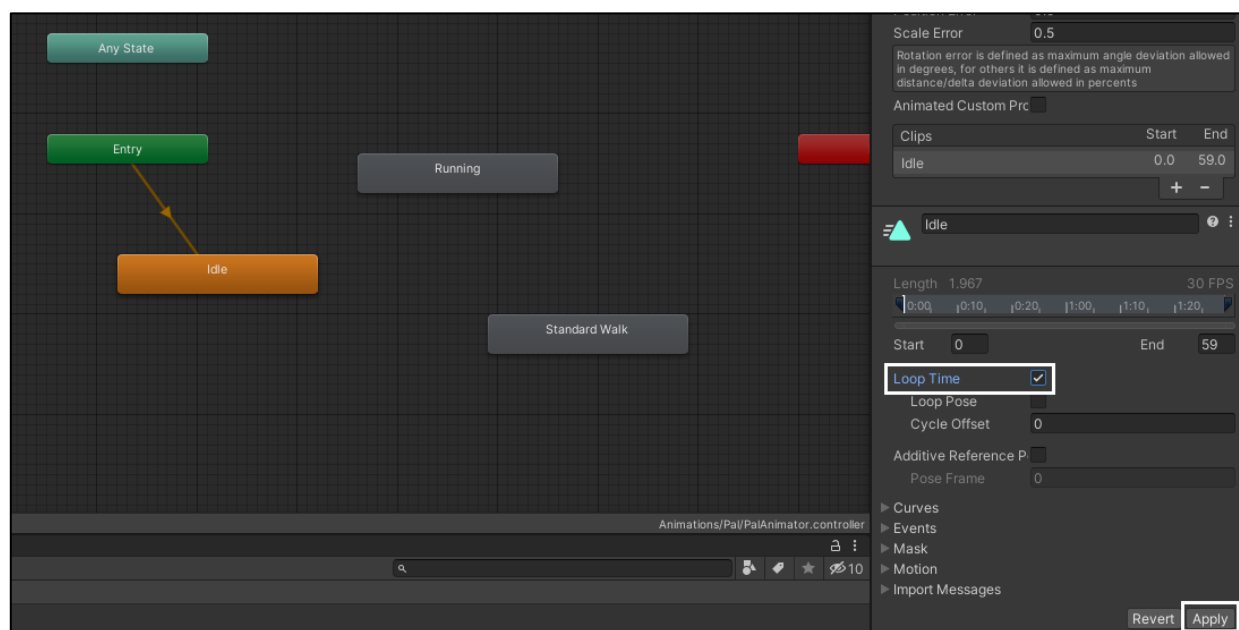
Анимация по умолчанию означает, что при старте работы аниматора, будет проигрываться именно эта анимация.

После установки анимации по умолчанию, перетащите аниматор на модель и запустите проект:



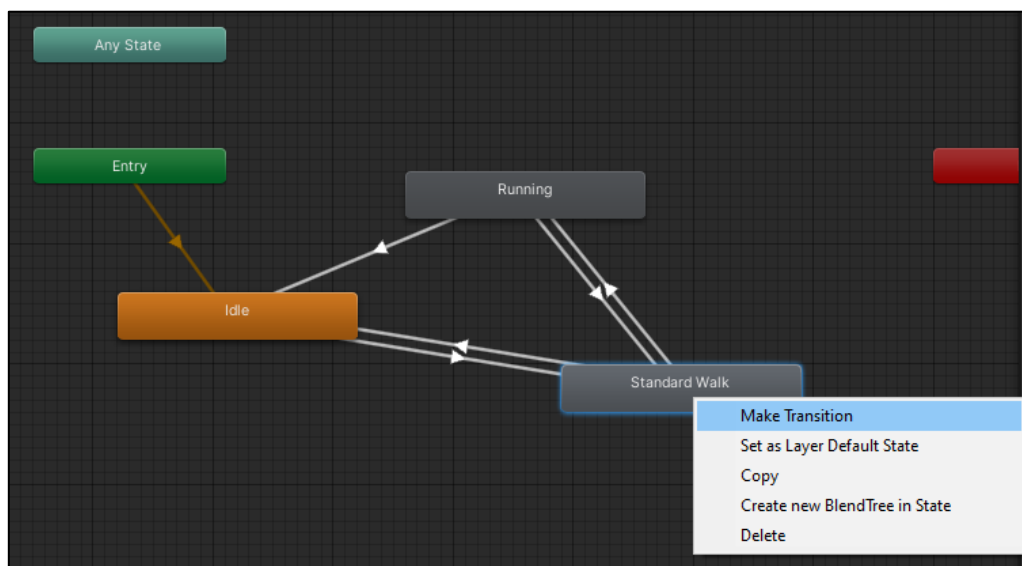
Если всё было сделано правильно, модель проиграет стартовую анимацию один раз.

Если требуется, чтобы анимация проигрывалась циклично, кликните по ней в поле аниматора дважды, отметьте поле Loop Time и нажмите Apply:

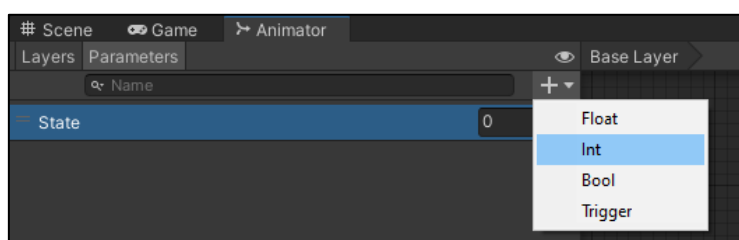


Запустите проект, теперь анимация по умолчанию должна проигрываться циклически.

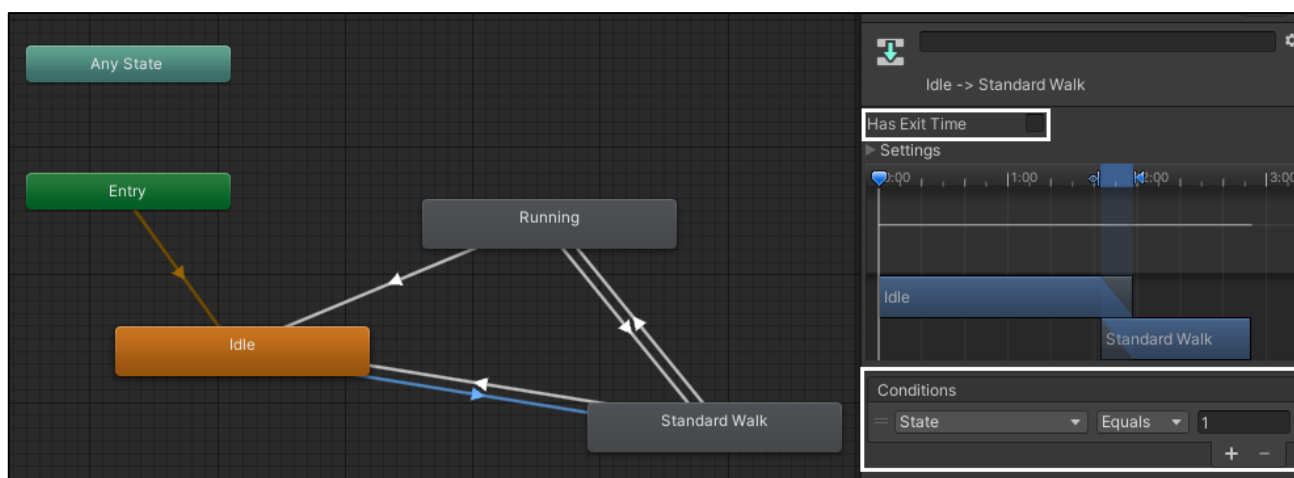
По сути, аниматор предоставляет инструментарий для создания конечных автоматов, определяющих список состояний и возможные переходы между ними. Что бы добавить переход между анимациями, кликните по анимации из которой осуществляется переход правой кнопкой мыши, затем выберите Make Transition, и кликните левой кнопкой мыши по анимации, в которую необходимо выполнить переход. Ниже приведён пример переходов между анимациями простоя, ходьбы и бега:



По умолчанию, переход из одной анимации в другую будет осуществляться автоматически, при завершении предыдущей анимации. Для управления переходами, можно создать переменную, которая будет частью переходов по условию. Для этого кликните на вкладку Parameters аниматора, затем знак “+” и выберите тип переменной:



Для создания условного перехода, кликните на переход, затем в разделе Conditions нажмите “+” и задайте значение переменной, при которой будет выполняться данный переход:



Обратите внимание на то, что каждый переход имеет параметр Has Exit Time. Если этот параметр отмечен, то переход будет выполняться только после завершения текущей анимации. Если нет, то текущая анимация будет прервана.

Доступ к переменной отвечающей за условные переходы можно получить из скрипта. Для этого, добавьте скрипт следующего вида к объекту, содержащему аниматор:

```
public class PalScript : MonoBehaviour
{
    Animator anim; //ссылка на аниматор
    Ссылка: 0
    void Start()
    {
        anim = GetComponent<Animator>();
    }

    Ссылка: 0
    void LateUpdate()
    {
        if (Input.GetAxis("Vertical") > 0) //если нажата клавиша вперёд
        {
            anim.SetInteger("State", 1); //установка состояния в "Ходьба"

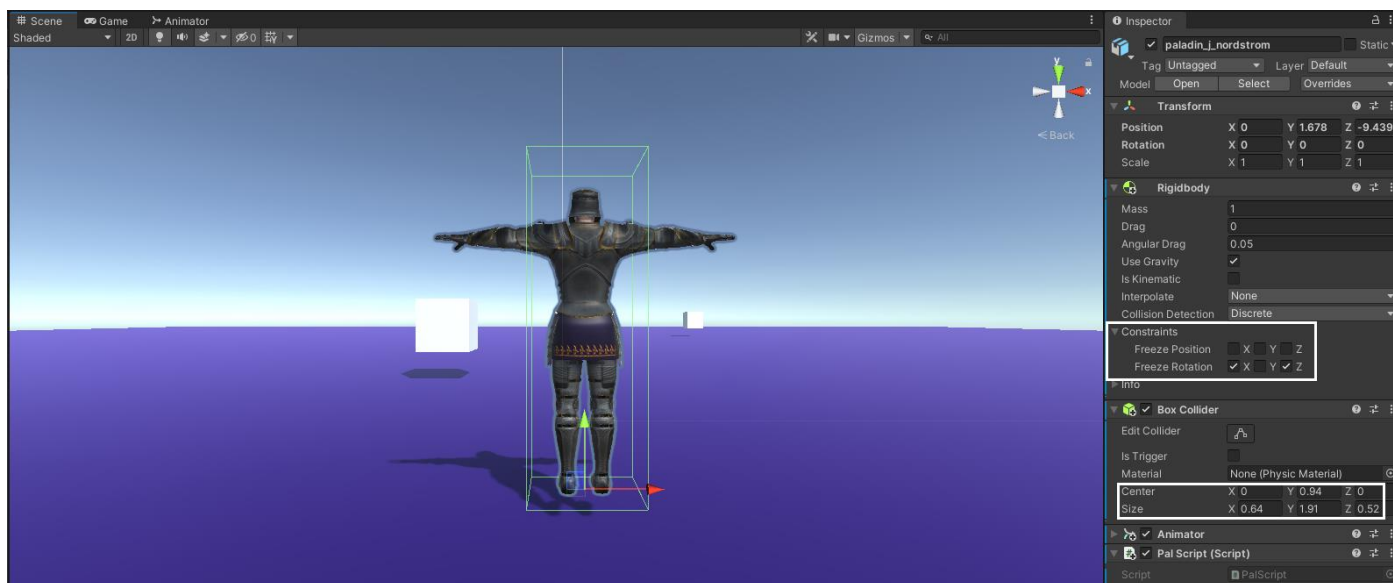
            if (Input.GetKey(KeyCode.LeftShift)) //если зажат левый шифт
            {
                anim.SetInteger("State", 2); //установка состояния в "Бег"
            }
        }
        else
        if (Input.GetAxis("Vertical") <= 0) //если клавиша движения вперёд не нажата
        {
            anim.SetInteger("State", 0); //установка состояния в "Стоит"
        }
    }
}
```

Перемещение анимированного объекта осуществляется при помощи методов, рассмотренных в лабораторной работе №1. Повороты объекта можно осуществлять при помощи кода следующего вида:

```
transform.Rotate(Vector3.up, 90 * Time.deltaTime, Space.Self); //поворот вправо относительно локальной оси Y
```

Добавление коллайдера и физического тела

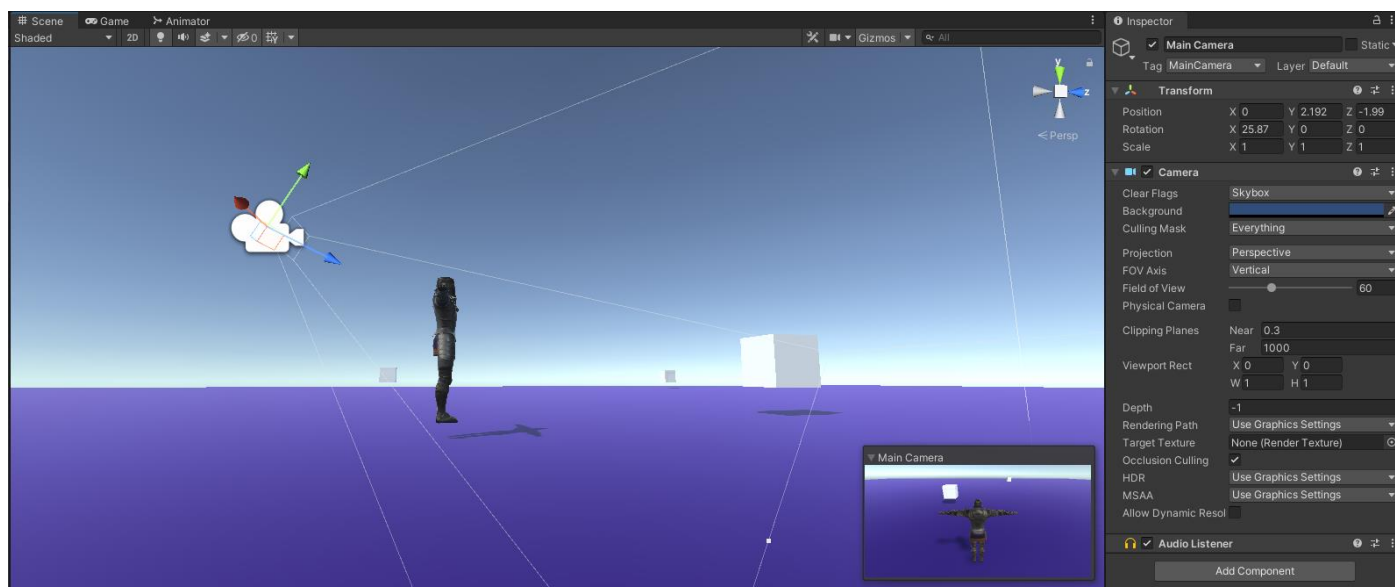
Не все анимированные модели полученные из различных источников имеют коллайдеры и физические тела сразу же. Добавить недостающие элементы можно используя меню Add Component:



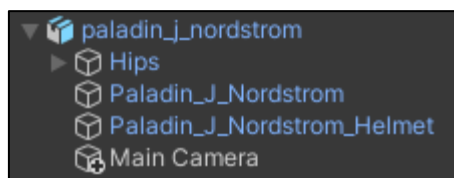
Во избежание странностей в физическом взаимодействии анимированной модели и объектов сцены, можно зафиксировать повороты объекта по осям X и Z, используя меню Constraints. Размеры и положение коллайдера подбирается вручную, изменением позиции центра и масштабов по осям.

Следящая камера

Для реализации базового функционала следящей камеры от третьего лица, разместите камеру за спиной модели в желаемой позиции:



А затем, в разделе Иерархия, переместите камеру на модель так, чтобы она стала её частью:



Если всё было сделано правильно, то камера будет наследовать перемещения и вращения модели, то есть будет следовать за ней и поворачиваться вместе с ней.

Задание:

Создать приложение, содержащее игровой уровень, модель персонажа и набор анимаций. Пользователь приложения должен иметь следующие возможности:

1. Перемещать персонажа при помощи клавиш W (движение вперёд), A (поворот вправо), S (поворот влево), D (движение назад). При зажатой клавише Shift, персонаж должен переходить на бег. Каждое движение (кроме поворотов) должно иметь свою анимацию. При отсутствии движения, должна воспроизводиться анимация простая.
2. Совершать прыжки при помощи клавиши Spacebar. Прыжок должен состоять из двух анимаций – отталкивание и падение. Анимация падения должна воспроизводиться при любом перемещении персонажа вниз. Анимация падения должна проигрываться до тех пор, пока персонаж не окажется на какой-либо поверхности.
3. Выполнять атаку/наносить удар при нажатии клавиши Control. Переход к другим анимациям (кроме падения) должен быть не возможен до завершения анимации удара/атаки.

Пример выполнения первой части задания: <https://h-anim.github.io/animation/>