

Лабораторная работа № 3. Проект – разработка простого AR-Приложения для Android-устройства (смартфон, планшет и пр.). Создание в графическом редакторе Unity 3D сцены дополненной реальности: визуализация управляемой анимации 3D-Модели с использованием плоского кнопочного меню. **Часть 1. Изучение возможностей по созданию анимации объектов контента в Проектах AR.**

Объекты ДР – это объекты проекта, создаваемого с помощью средств платформы Vuforia.

### Введение.

Работа по созданию приложений ДР заключается в создании проекта и объектов проекта (Контент ДР) в Vuforia, а разработка 3D-сцен для объектов этого проекта осуществляется в Unity 3D. При этом Vuforia отвечает за идентификацию проекта через License key (см. описания предыдущих ЛР), а привязка к будущей сцене виртуального 2D- или 3D-объекта (например, 3D-модели, плоских изображений, видеоклипов и пр.) будет осуществляться через определяемую в Vuforia метку (Target). Допустимые в используемой в Лабораторном практикуме версии Vuforia Engine типы таргетов были подробно рассмотрены в Описании ЛР№2, Часть 1.

**ВАЖНО!!** → вся работа с Vuforia (с проектом, объектами) осуществляется через web-интерфейс, иными словами, Vuforia является облачным приложением. А работа с Unity-3D осуществляется непосредственно на компьютере разработчика, т.е. локально.

Связь между облачным ведением проекта (в Vuforia) и локальной проработкой сцен Приложения ДР должна быть выполнена за счет импорта подготовленных объектов проекта из облака Vuforia в среду редактора Unity-3D и дополнения сцены 2D-изображениями или 3D-моделями.

**Задание на ЛР№3, часть 1** – разработка элементов виртуального интерфейса для работы в ДР, т.е. установление связей между поведением (behaviour) виртуальной 3D-модели (у нас - «анимация») и состоянием элемента управления («кнопка» → события: кнопка отжата/кнопка нажата).

Предлагается разработать приложение ДР для Android-устройств, в котором при наведении камеры устройства на реальную метку (таргет – изображение, например, на бумаге, или на дисплее) пользователь в области воспроизведения на экране мобильного устройства (МУ) увидит заранее подготовленную и размещенную в сцене в рамках выполненного задания ЛР№2, Часть3, **3D – модель, которая будет анимирована. А управление анимацией будет осуществляться с помощью кнопки плоского меню.**

**В первой части ЛР№ 3 студенты** осваивают способы задания анимации объектов контента ДР.

#### Предварительные условия для начала работы:

- Интернет-соединение локального компьютера;
- Наличие аккаунта пользователя Vuforia (результат успешного выполнения ЛР №1);
- Установленная на компьютере разработчика система Unity 3D (результат успешного выполнения ЛР №1);

- Заранее подготовленные изображения для метки (**таргета**) и объектов контента (**3D-модель в одном из допустимых форматов**).

**В результате** успешного выполнения **ЛР №1**, получены из облака **Vuforia** и сохранены в локальной файловой системе для локальной работы в **Unity 3D** следующие ресурсы:

- **установочный** специфический объект **Vuforia** – файл в формате **.unitypackage**
- **лицензия**
- **БД таргетов** - файл в формате **.unitypackage**
- Кроме того, в рекомендуемом для использования проекте размещена **3D-модель в одном из допустимых форматов**.

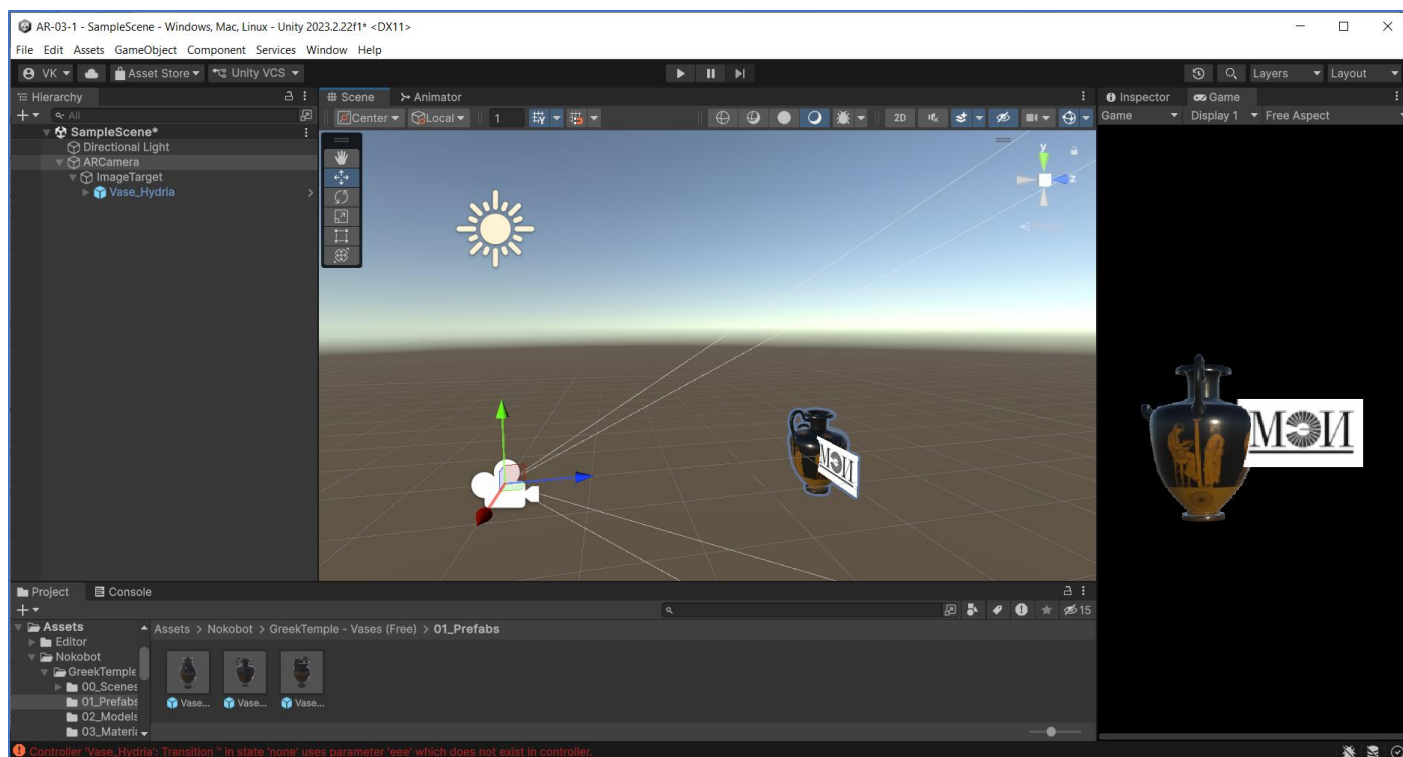
**ВАЖНО!!!** Проект уже создан, все необходимые настройки для этого выполнены и описаны в пунктах 1., 2. в описаниях ЛР №№1 и 2 (все части).

**Т.о. – переходим сразу к пункту 3. Описания →**

3. **Переходим в проект в Unity 3D.** Продолжаем работать в проекте, заведенном в **ЛР№2, часть 3** (с геометрической моделью, в данном описании – это **3D-модель греческой амфоры**). В нашем примере Проект имеет название **ARProj003**.

Вы можете завести новый проект, с новой моделью – опыта у вас уже достаточно. Все необходимые шаги были проделаны вами в предыдущей ЛР. Если вы решили использовать новую модель, повторите эти шаги. В нашем случае на данном этапе выполнения ЛР состав сцены будет таким, как представлено на рисунке ниже.

**Постарайтесь** добиться визуально привлекательного и однозначно распознаваемого соотношения таргета и модели в сцене, чтобы в дальнейшем наблюдать разрабатываемые анимации наилучшим образом.



**ВАЖНО!!** Обратите внимание - на этом шаге у нас уже создан объект контента, связанный с таргетом – **Vase\_Hydria** → Смотрите, как это отображено в **Hierarchy**.

4. Для управления виртуальным объектом **Vase\_Hydria** создадим поведение (**behaviour**) 3D-модели, которым будем управлять с помощью элемента управления – «кнопка плоского меню». Например, при нажатии на кнопку виртуальный объект должен:

- вращаться,
- перемещаться,
- масштабироваться и т.д.

Для этого воспользуемся базовой функциональностью **Unity 3D** и выберем достаточно сложное, multifunctional поведение (**behaviour**) - **Animation**.

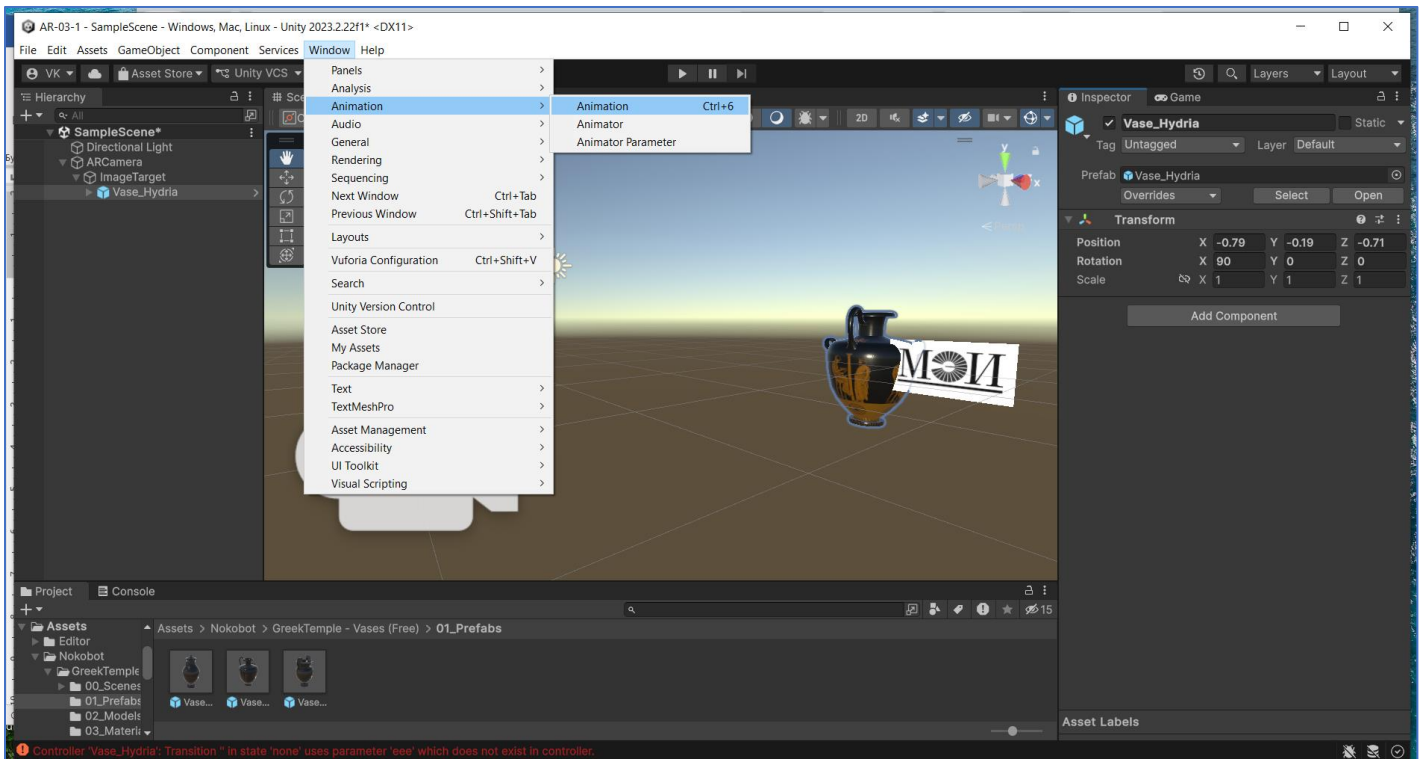
\*\*\*

Соответствующий раздел описания **Unity 3D** (выберите одну из последних версий) находится по адресу:

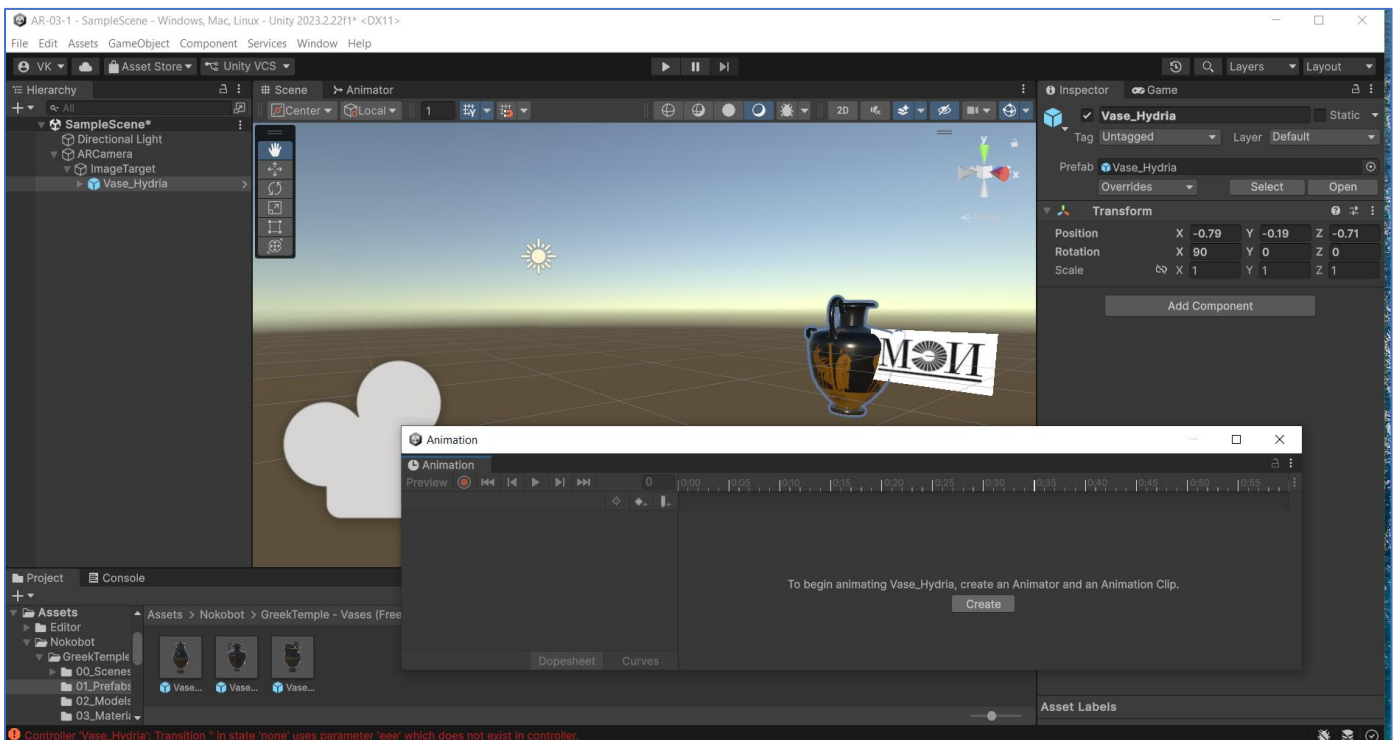
<https://docs.unity3d.com/ru/current/Manual/AnimationSection.html>

\*\*\*

Для создания анимации – выбираем в **Hierarchy** объект **Vase\_Hydria**, а затем выбираем в падающем меню **Window**→**Animation**→**Animation**:



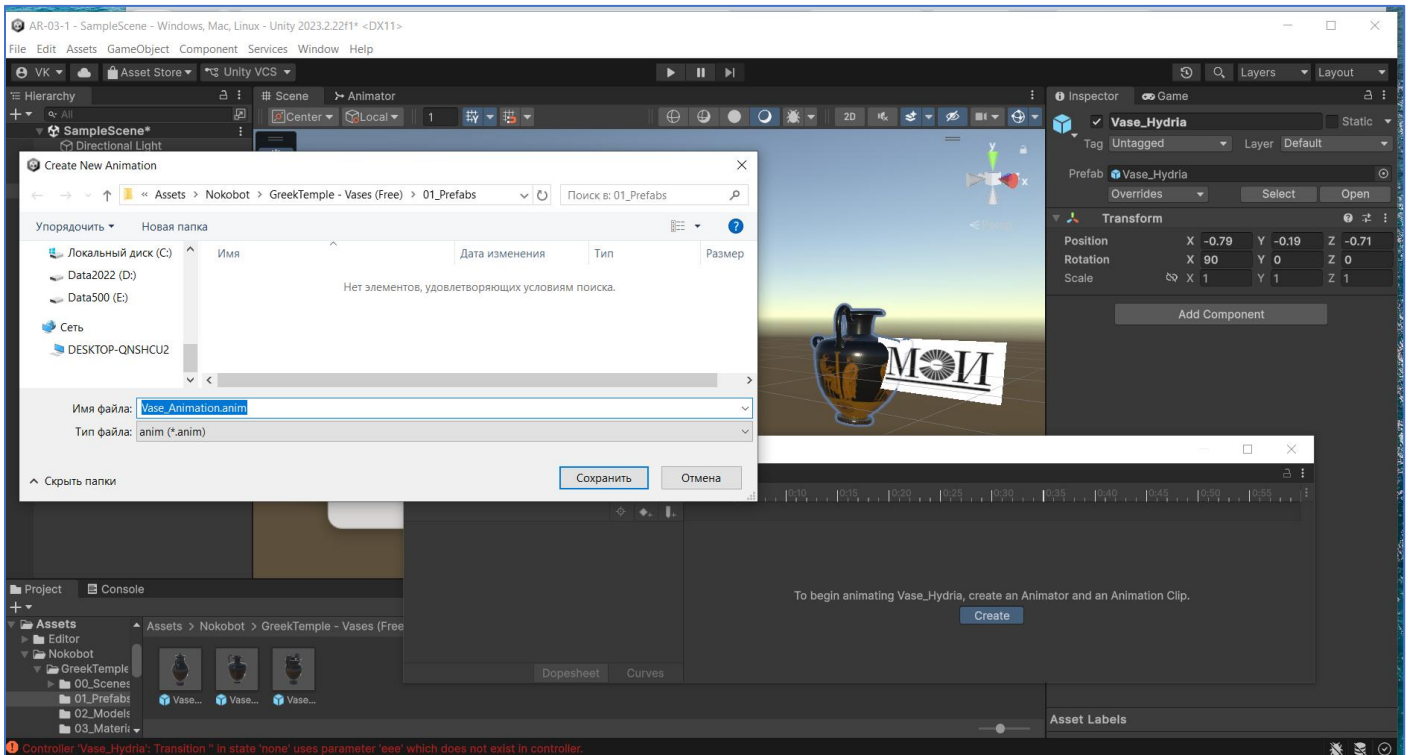
В результате появляется окно создания анимации:



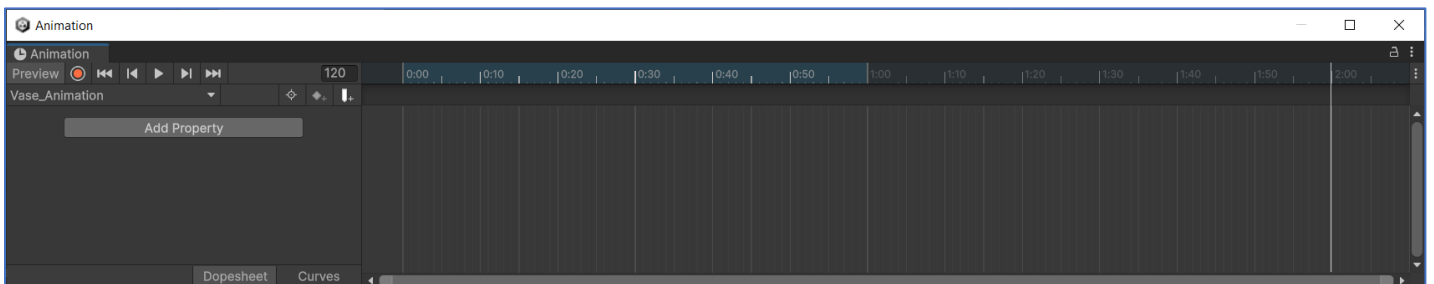
По клавише **Create** переходим к созданию анимации:

«Анимация» (сочетание известных вам стандартных трансформаций объектов в трехмерном пространстве – **поворот/сдвиг/масштабирование**) в данной системе представляет из себя набор данных, который сохраняется в локальной файловой системе в файле с расширением **.anim**. Присвоим ему пользовательское имя **Vase\_Animation.anim**.

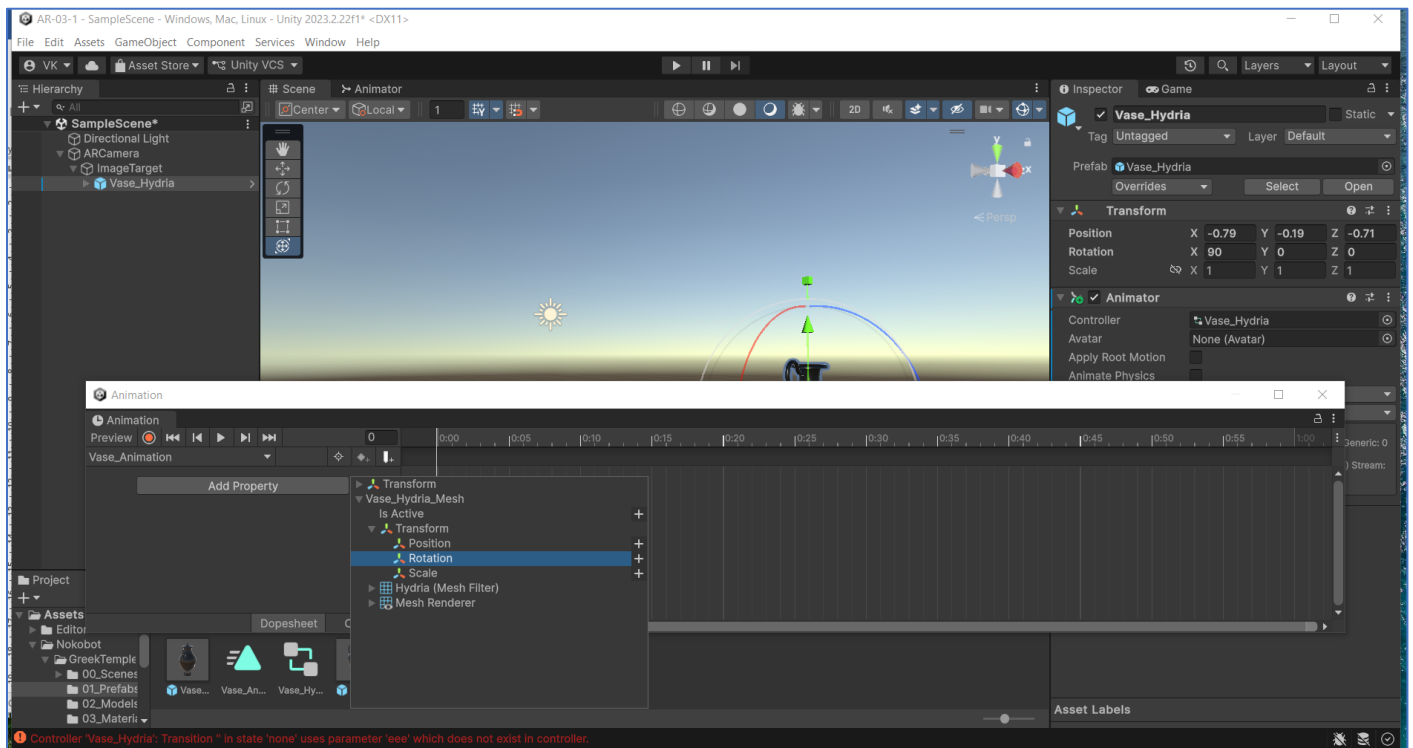
**Зафиксируйте это имя для дальнейшего использования в организации управления с помощью элементов плоского меню, например в скриптах и сценариях управляемой анимации.**



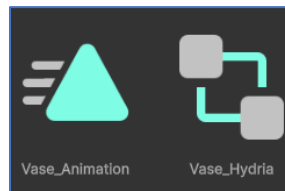
После сохранения мы попадаем в окно создания **Animation**:



В данной ЛР для простоты создадим **Animation** в виде простого вращения объекта контента – **3D-модели Vase\_Hydria** вокруг **оси Z** (в нашем случае это и есть вертикальная ось вращения) на **360 градусов** за определенное время (**1сек**). Для этого в окне **Animation** выбираем **Add Properties**→**Vase\_Hydria**→**Transform**→**Rotation**:

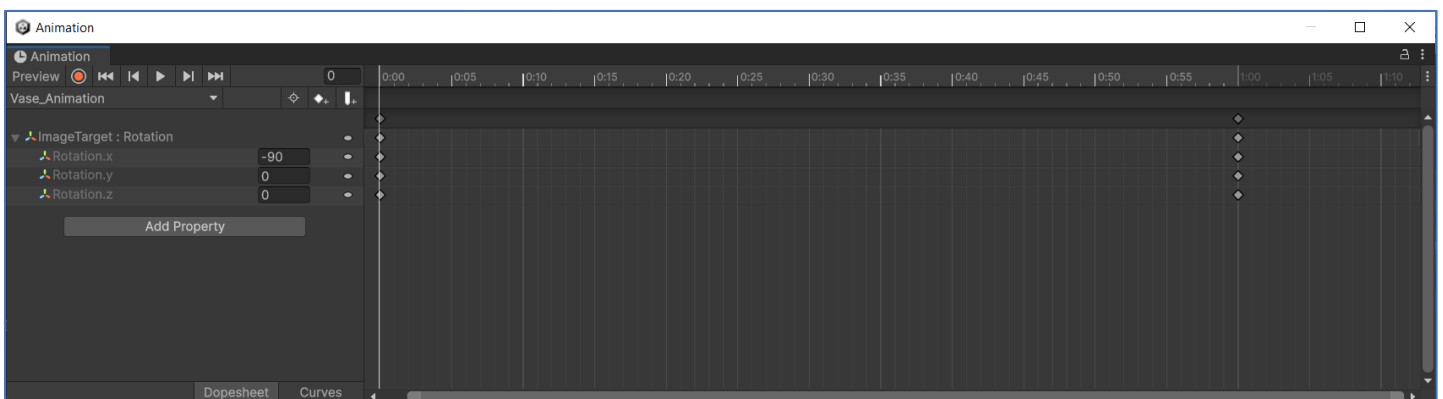


**ВАЖНО!!** Обратите внимание – в области Assets у нас появились два объекта – сама анимация с выбранным нами названием **Vase\_Animation** и контроллер анимации с именем анимируемого объекта **Vase\_Hydría**.



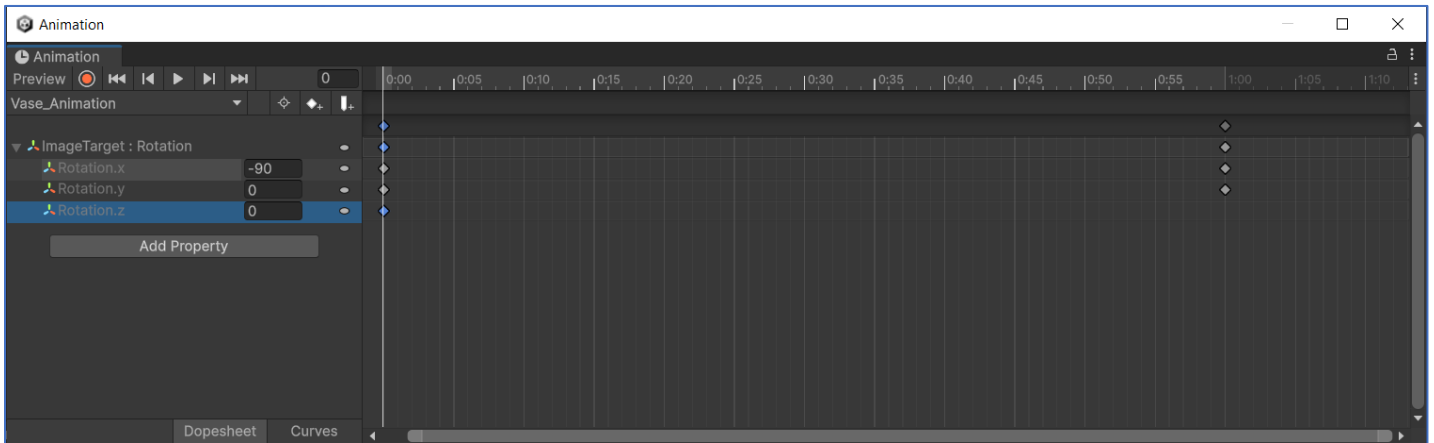
Как было рекомендовано выше, из всех возможных типов изменений для выполнения анимации выберем известный вам из других курсов стандартный тип изменения - **преобразование** – **Transform**, который распадается на **поворот/сдвиг/масштабирование**.

Добавляем **Rotation (+)**:

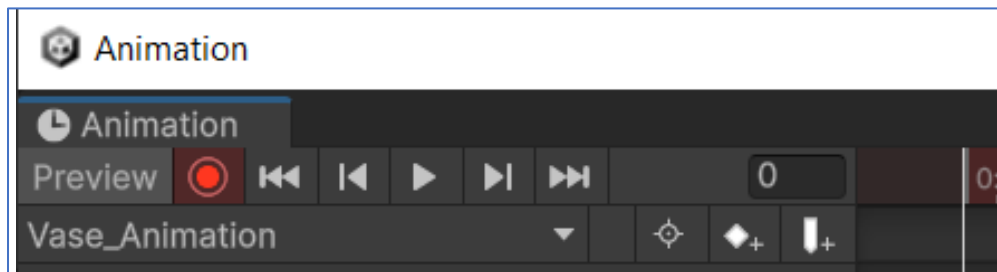


Выбираем нужный тип поворота (**Rotation**) – **вокруг оси Z**. Для этого раскрываем

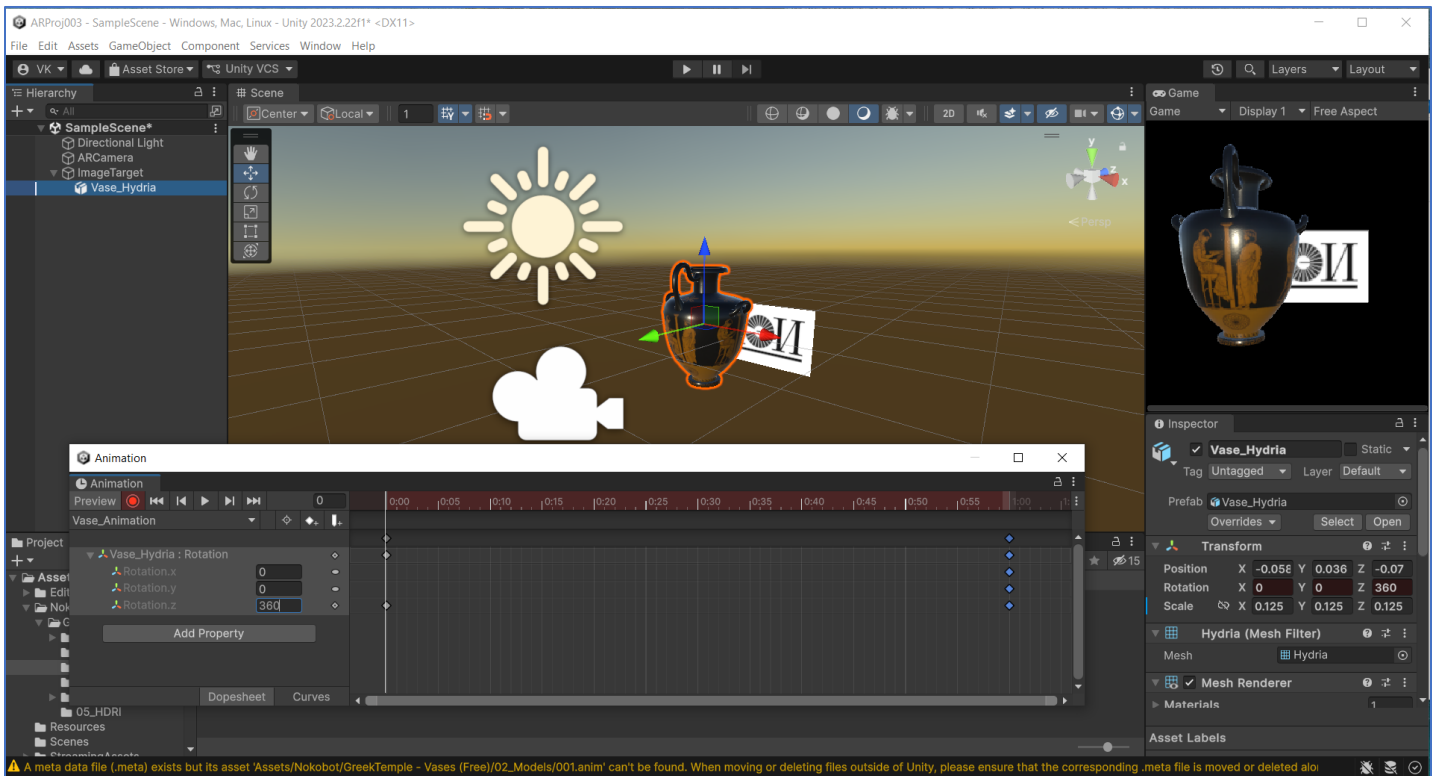
**Vase\_Hydria : Rotation** и выбираем соответствующее вращение:



Для того, чтобы сформировать необходимую нам анимацию – вращение вокруг вертикальной оси (**Z**) – нажимаем красную клавишу записи (**Red Button**)

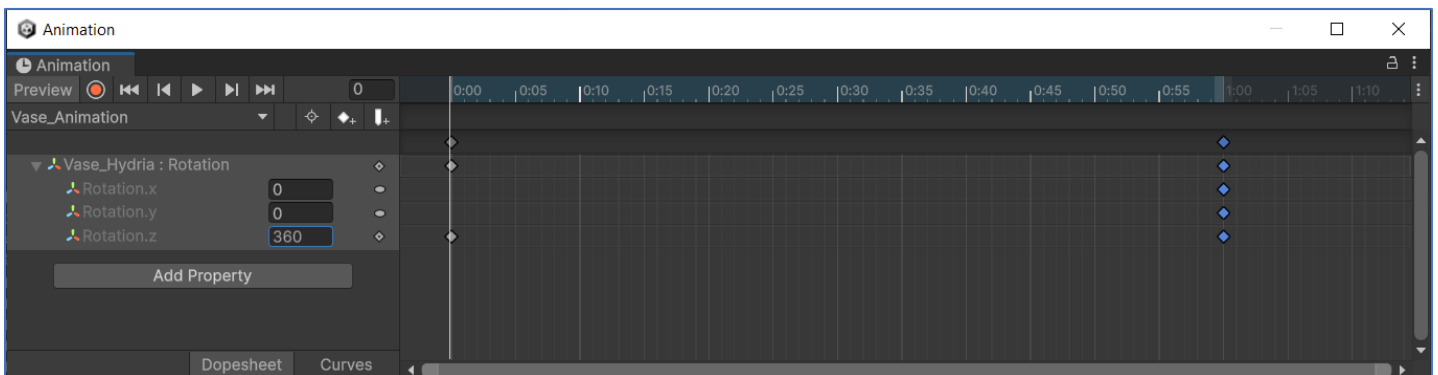


Затем в **таймлайне окна Animation** (верхняя горизонтальная шкала окна), при выбранном вращении вокруг оси Z, перемещаем белый вертикальный ползунок в позицию «**1 сек**» (крайняя правая), и в позиции **Rotation Z (в любой!!!)** устанавливаем угол поворота (**360 градусов**). Это означает – **полный оборот вазы за 1 секунду вокруг вертикальной оси**.

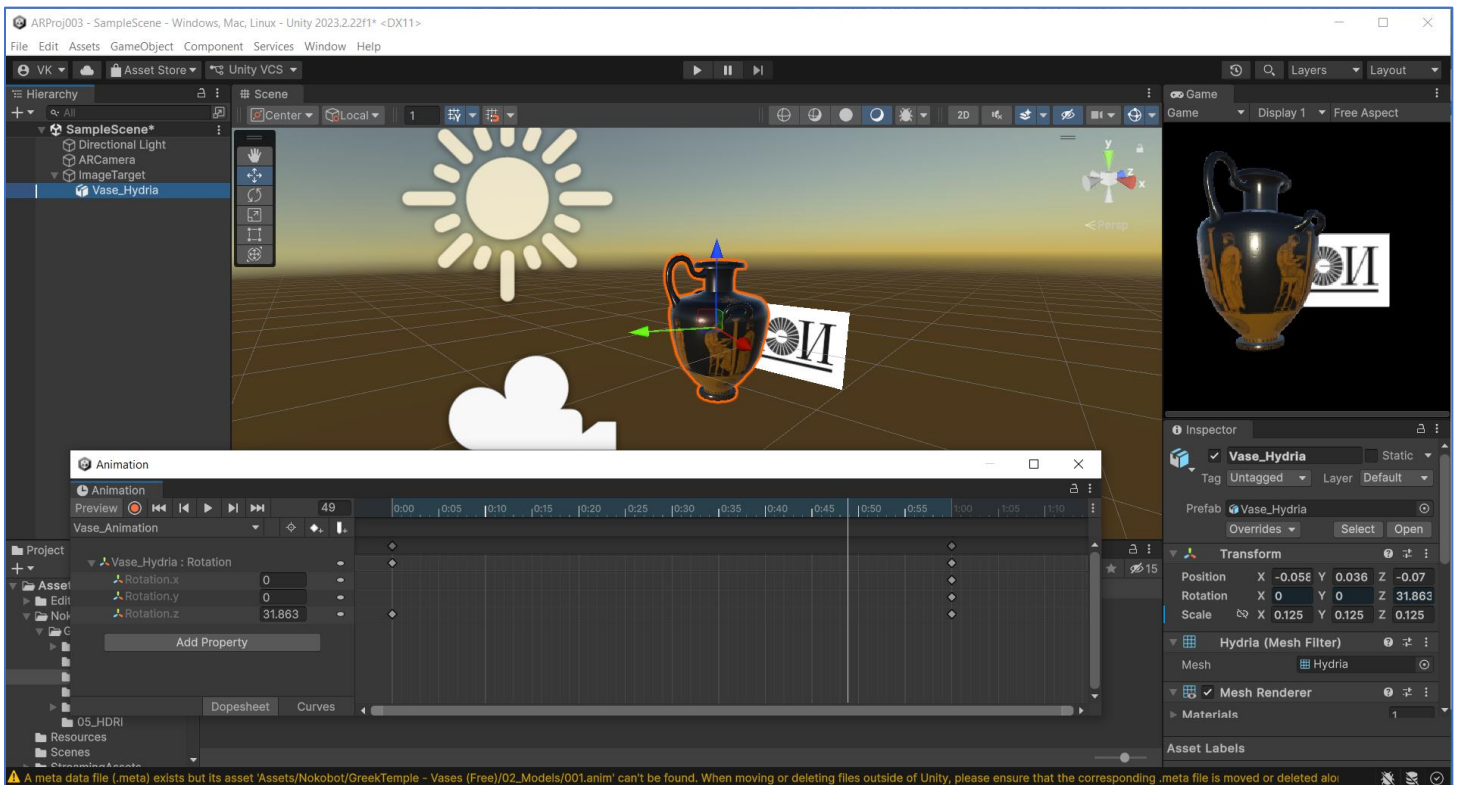


Убедитесь, что в позиции **Rotation Z** в **Inspector**'е установлено нужное значение угла поворота (**360**).

Оценим текущее состояние полученной анимации после записи. Для этого отжимаем кнопку **Записи (Red Button)** и включаем режим проигрывания – клавишу **Play**:

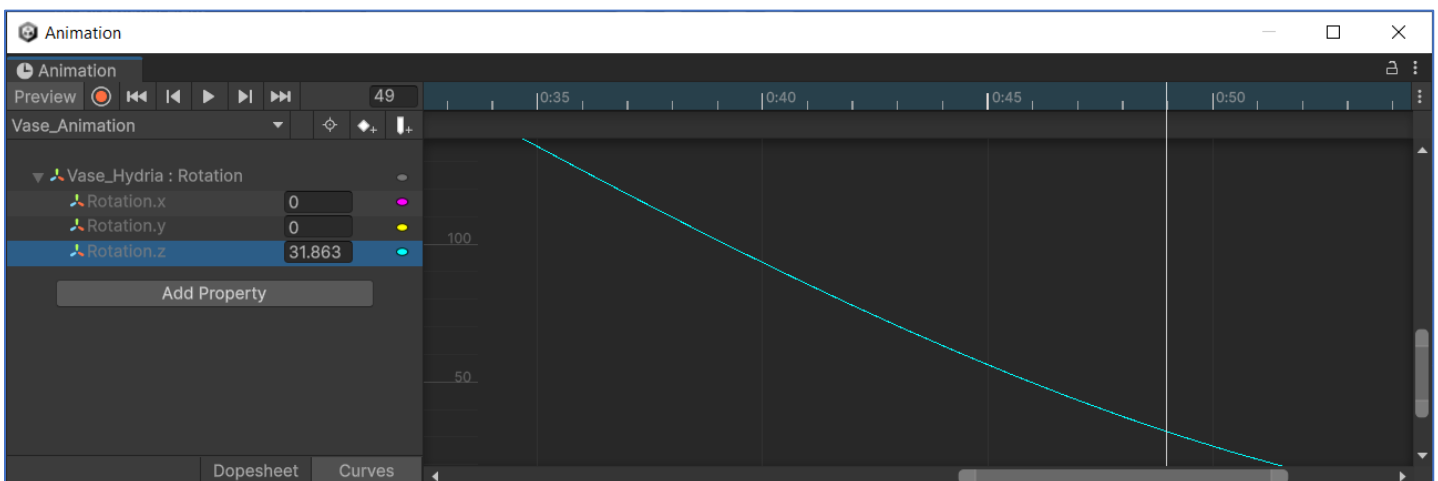


В результате в окне сцены **Editor**'а мы будем наблюдать записанную анимацию **Vase\_Hydría** – вращение вокруг вертикальной оси (ось **Z**).



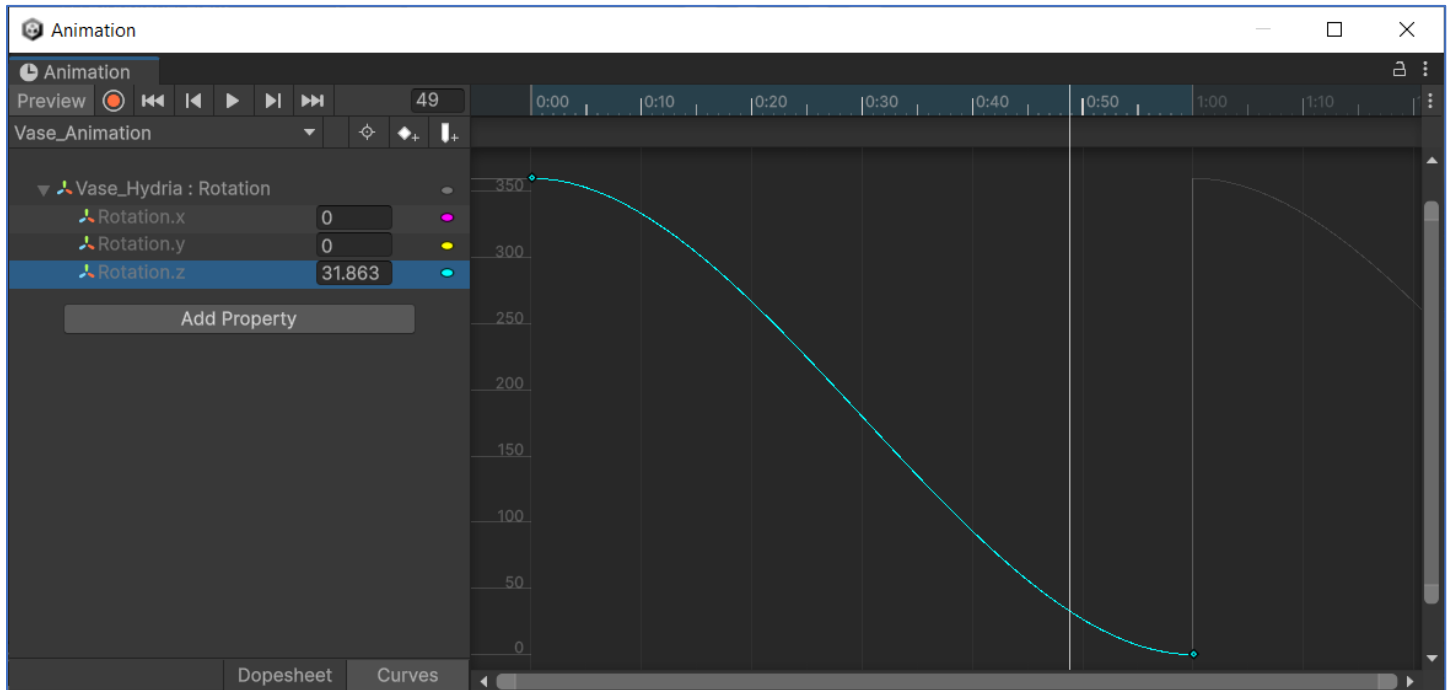
Теперь можно обработать еще один параметр анимации – скорость трансформации – у нас скорость вращения вокруг оси Z. Для этого в окне **Animation** переключимся в режим **Curves**  . В результате получаем график изменения параметра **скорость вращения**. График свидетельствует о том, что скорость вращения неравномерна.

В первый момент вы можете увидеть в окне отображения нечто подобное:



Вертикальная шкала у левой границы серого поля экрана – это градусы. При этом вы можете увидеть только часть кривой, описывающей скорость вращения. Для того, чтобы работать со всей кривой, вам нужно поменять масштаб, для этого у вас есть ползунок у правой вертикальной границы окна, горизонтальный ползунок, и среднее колесико мыши, изменение границ окна, с

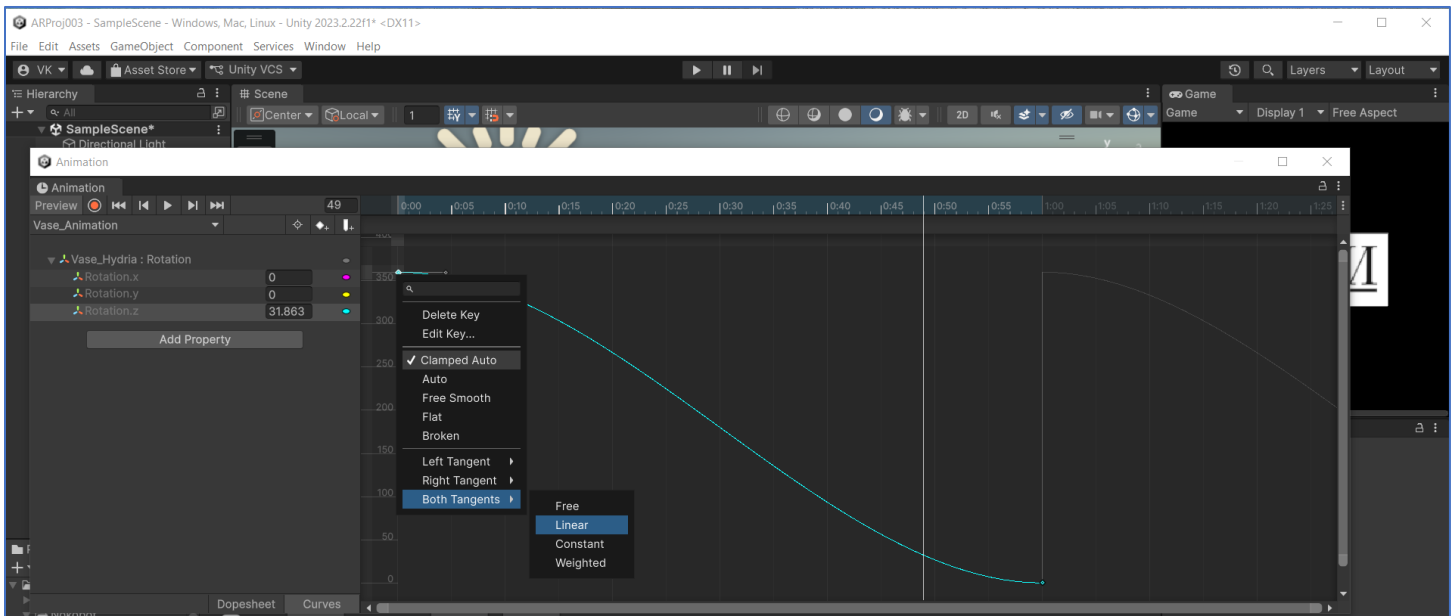
помощью которых вы можете менять масштаб внутри окна, по вертикали и по временной шкале. Добейтесь того, чтобы график скорости изменения был виден целиком. На кривой могут быть т.н. «ключи», как минимум два – в начале и конце кривой. Управлять кривизной и характером кривой можно с помощью этих ключей – добавляя/уменьшая их число, меняя их положение (с помощью клавиш мыши) на кривой и т.д. При этом можно наблюдать изменение анимации объекта в режиме предпросмотра (клавиша **Play**).



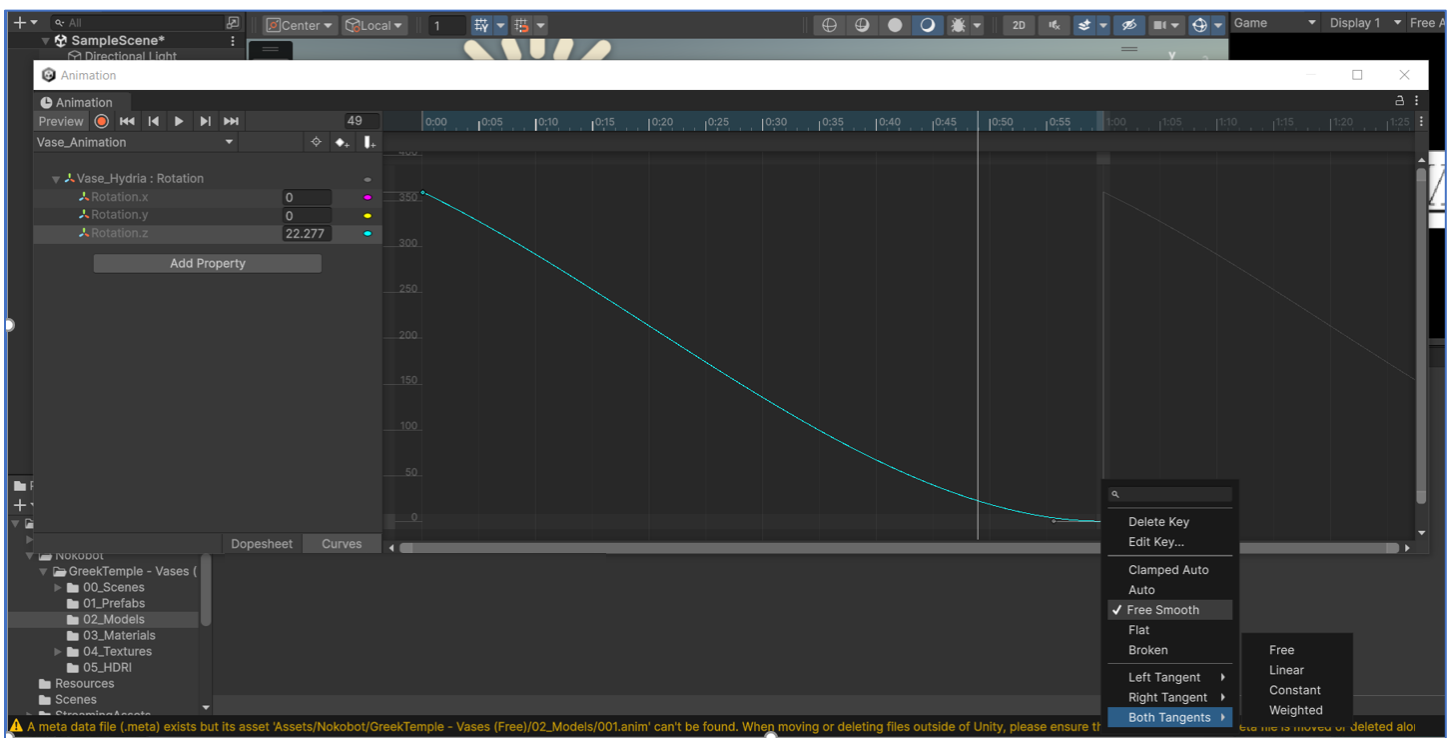
Теперь можно добиться равномерности скорости вращения. Равномерное вращение соответствует линейному характеру поведения кривой. Обратите внимание, что любая нелинейность означает ускорение или замедление вращения, изменение знака первой производной кривой – изменение направления вращения объекта контента. У разработчика есть масса инструментов, с помощью которых он может добиться различного характера анимации. Остается только освоить их все и поэкспериментировать с удовольствием.

Наша задача в данном примере – добиться равномерного вращения вазы. Для этого в начале кривой поместим указатель и по правой клавише мыши вызовем меню коррекции графика:

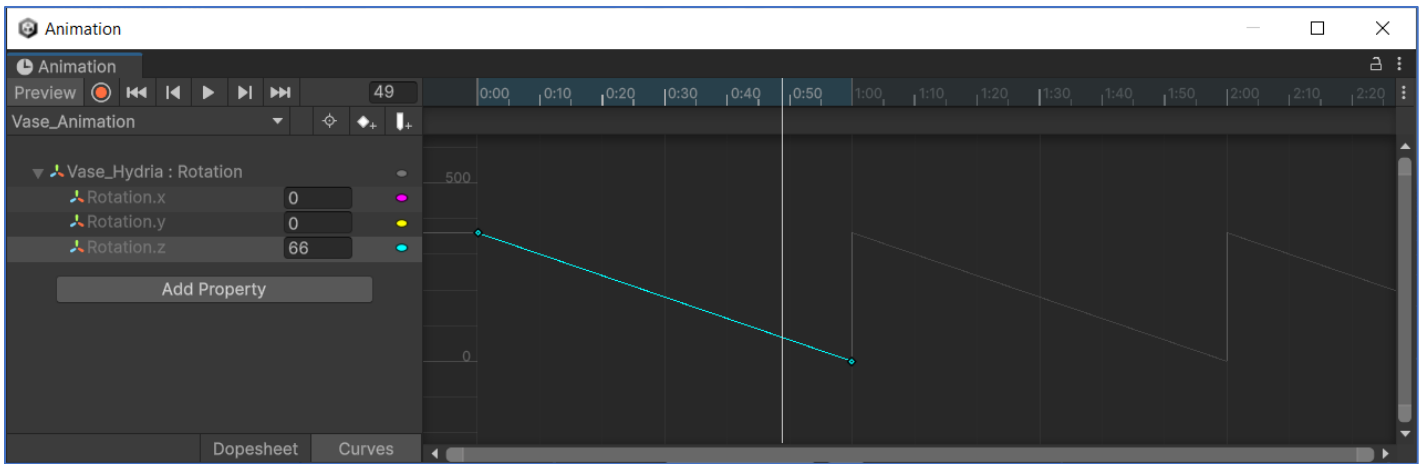
**Both Tangents→Linear**



Выполним те же самые действия на конечной точке **TimeLine**.

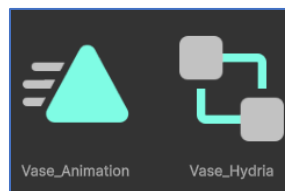


В результате мы добились равномерности скорости вращения →



**Работа** по созданию простейшей анимации виртуального объекта **3D Модели** завершена. Элементы меню окна **Animation**, с которым вы познакомились на этом шаге, **позволят вам создавать трансформации любой сложности на любом временном интервале.**

Закроем окно **Animation** и убедимся, что в **Assets окна Project** появились два новых **Assets** – сама анимация **Vase\_Animation** и контроллер анимации модели – **Vase\_Hydris**.



Запомним эти имена для дальнейшей работы.

На данном этапе вы можете протестировать свою сцену с анимированным объектом контента – **3D-моделью** - известным вам способом в редакторе **Unity 3D**, воспользовавшись камерой локального компьютера и выбранным таргетом.

Собранный файл **.apk** позволит продемонстрировать в режиме ДР ваш анимированный объект.

**Задание для самостоятельной индивидуальной работы. ЛР №3, Часть1:**

**Разработать 3 Приложения ДР для:**

- Вращения **3D-модели** вокруг выбранной оси (в зависимости от модели);
- Перемещения **3D-модели** по диагонали, например – из нижнего левого угла в верхний правый;
- «Наезда»/«Удаления» вращающейся модели на зрителя.

**Можно все три анимации разработать в одном проекте.**

Разработанные в рамках ЛР № 3, часть 1 AR-Приложения (Приложение) необходимо продемонстрировать преподавателю. Креатив приветствуется!

Одну (а может быть и все три) из разработанных в рамках этого задания анимаций вы сможете использовать во **второй части ЛР №3** для связывания с кнопкой плоского меню.