**Государственное бюджетное общеобразовательное учреждение Республики Крым «Керченская школа-интернат с усиленной физической подготовкой»**

**РАЗРАБОТКА УРОКА**

**«БУДЕМ ЗНАКОМЫ! АЛИСА»** 

**5 КЛАСС. ОСНОВЫ ОБЪЕКТНО-ОРИЕНТИРОВАННОГО ПРОГРАММИРОВАНИЯ НА JAVA**

**Учитель Романец К.С.**

**Керчь, 2018 год**

**Цели урока:** 

***предметные***

1. Познакомить со средой программирования «Alica 3»
2. Научить создавать новый проект в изучаемой среде программирования. Формирование навыков программирования

***метапредметные –***

умение планировать пути достижения целей; соотносить свои действия с планируемыми результатами; осуществлять контроль своей деятельности; определять способы действий в рамках предложенных условий; корректировать свои действия в соответствии с изменяющейся ситуацией; оценивать правильность выполнения поставленной задачи; ИКТ компетентность

***личностные*** –

понимание роли информационных процессов в современном мире;

**Оборудование урока:** компьютеры, мультимедийный проектор, экран,

# программное обеспечение

**План урока**   
1. Сообщение темы и постановка целей урока (организация начала занятия). – 2 мин.

1. Контроль выполнения домашнего задания -10 мин
2. Изучение новой темы – 25 мин
   1. идентификация компонентов сцены
   2. создание и сохранение нового проекта
   3. добавление объекта в сцену
   4. сохранение разных версий одной сцены
   5. простые инструкции по программированию
   6. использование команд «копировать» и «отменить»
   7. необходимость тестирования и отладки
3. Подведение итогов. 3 мин.
4. Домашнее задание. – 3 мин.

## **Ход урока**

## 1**.Сообщение темы и постановка целей и задач урока**

## Учитель сообщает тему урока обучающимся

## 2. **Контроль выполнения домашнего задания**

(Выбрать любой жанр компьютерной игры и подготовить презентацию, в которой необходимо дать определение выбранному жанру игры, описать его особенности и характеристики. Привести примеры игр. В презентации необходимо использовать не менее 8 слайдов, обязательно наличие иллюстраций в презентации ссылок на используемый материал).

Вопросы:

• Дайте определение понятию «компьютерная игра».

• Назовите классификацию игр.

• Сформулируйте особенности каждого жанра компьютерной игры.

## **3. Изучение новой темы:**

## Начальная сцена

Начальная сцена – отправная точка вашей анимации. Она содержит 3 компонента:

* задний фон, который предусматривает небо, землю и свет
* неподвижные объекты для декорации
* движимые объекты

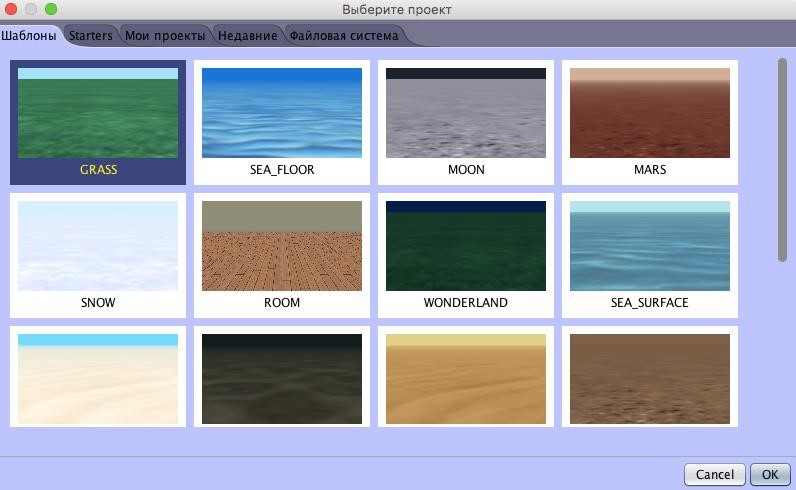
Начальная сцена – это первая сцена вашей анимации. Здесь вы выбираете задний фон и позиции объектов.

## Компоненты начальной сцены



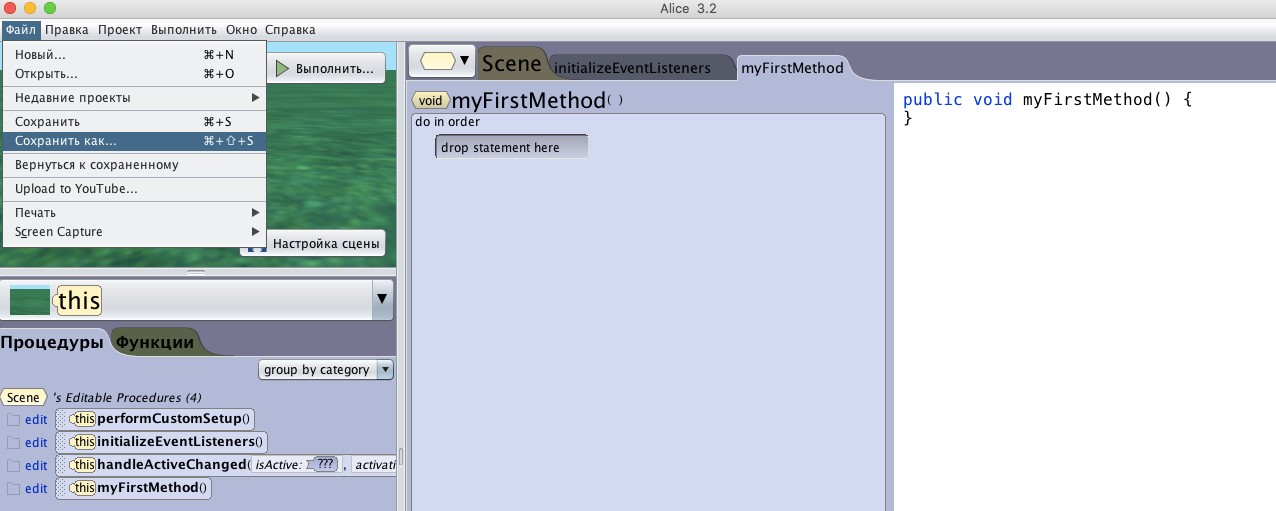
## Создание нового проекта

* Запустите Alice 3
* Выберите вкладку «Шаблоны»
* Выберите задний фон и нажмите «ОК»



## Сохранение проекта

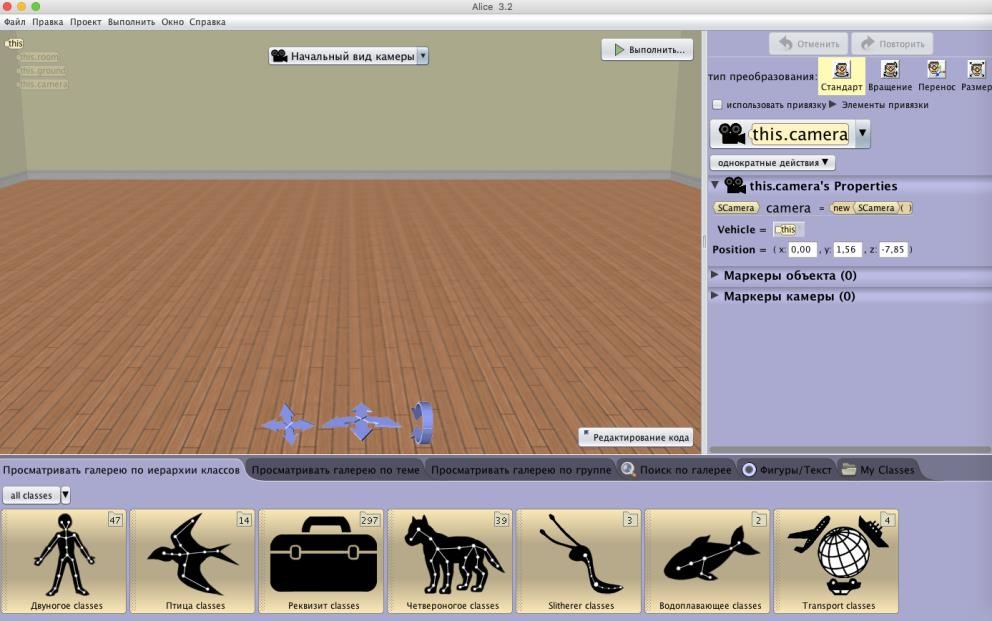
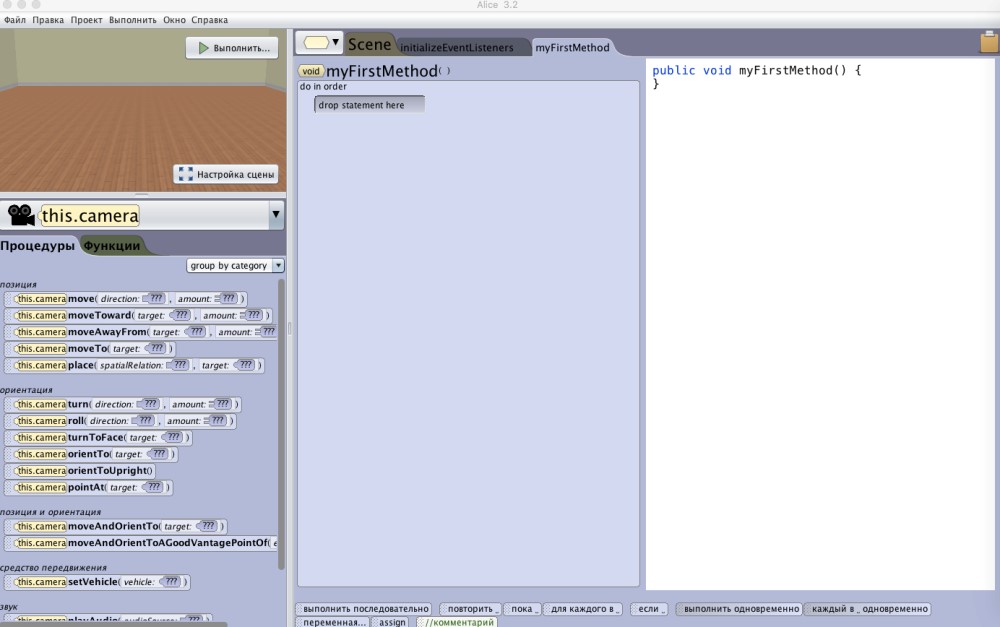
* В верхнем меню выбираем «Файл» – «Сохранить как…»
* Выбираем папку для сохранения (например, «Рабочий стол»)
* Вводим название проекта
* Нажимаем «Сохранить»
* Регулярно сохраняйте проект во избежание потери внесенных изменений



## Переключение режимов редактирования

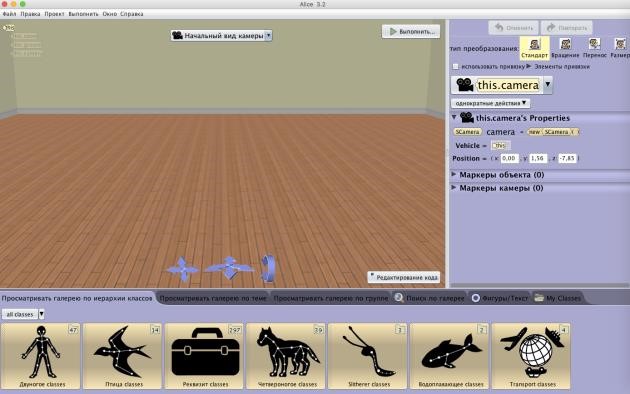
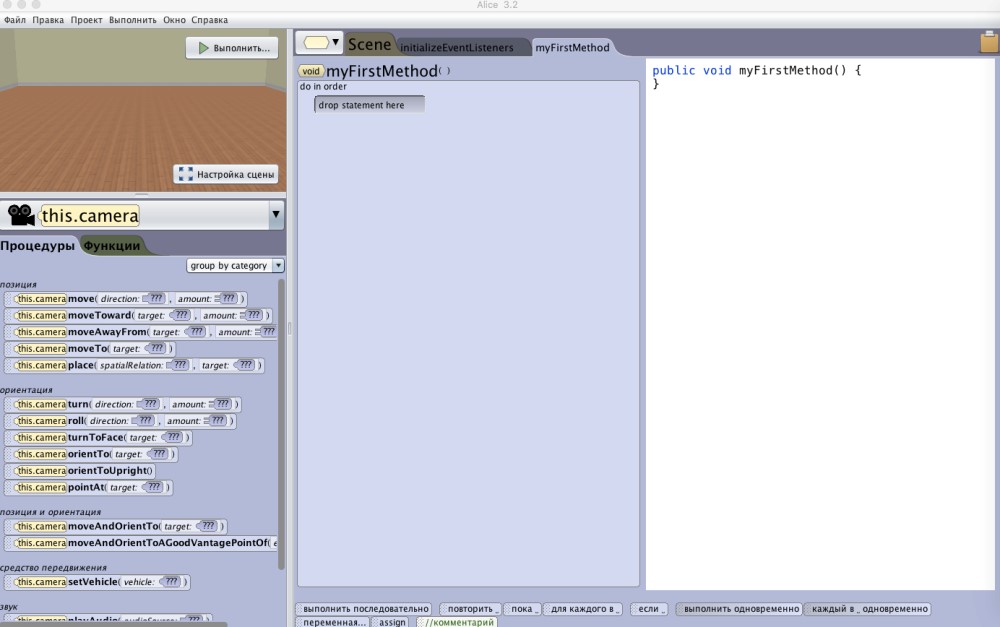
В Alice доступны 2 режима редактирования, которые вы будете регулярно переключать в процессе работы над проектом:

* редактирование кода (показано на скриншоте слева)
* настройка сцены (показано на скриншоте справа)



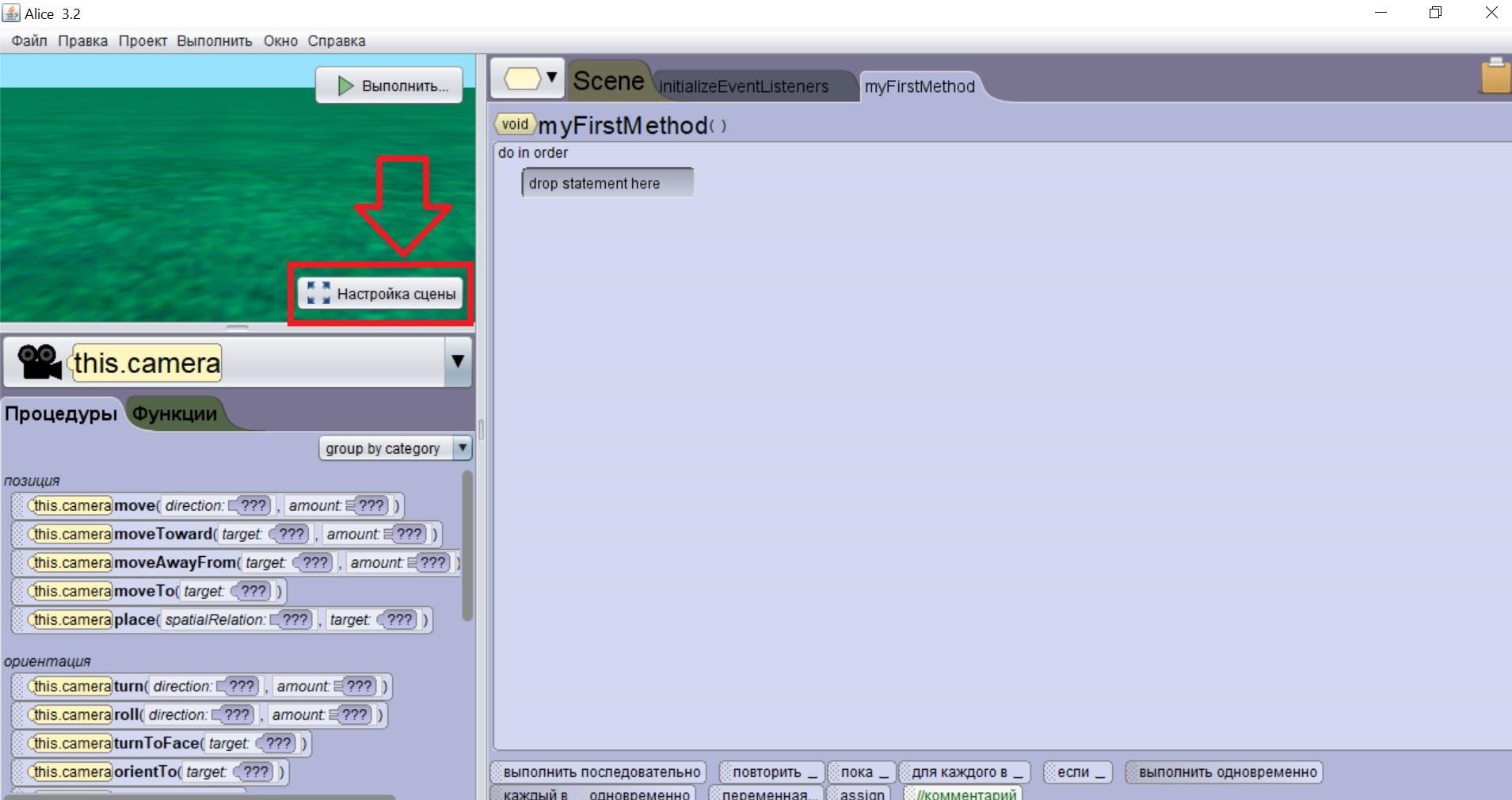
## Переключение режимов редактирования

* Переключать режимы редактирования вы можете с помощью кнопок «Редактирование кода» и «Настройка сцены»
* Если на экране слева вы видите набор команд, значит, вы находитесь в режиме редактирования кода
* Если на экране снизу вы видите галерею объектов, значит, вы находитесь в режиме настройки сцены



## Стартовый режим редактирования

В качестве стартового в Alice используется режим редактирования кода. Для перехода в режим настройки сцены нажмите на кнопку «Настройка сцены»:



## Добавление объекта в сцену

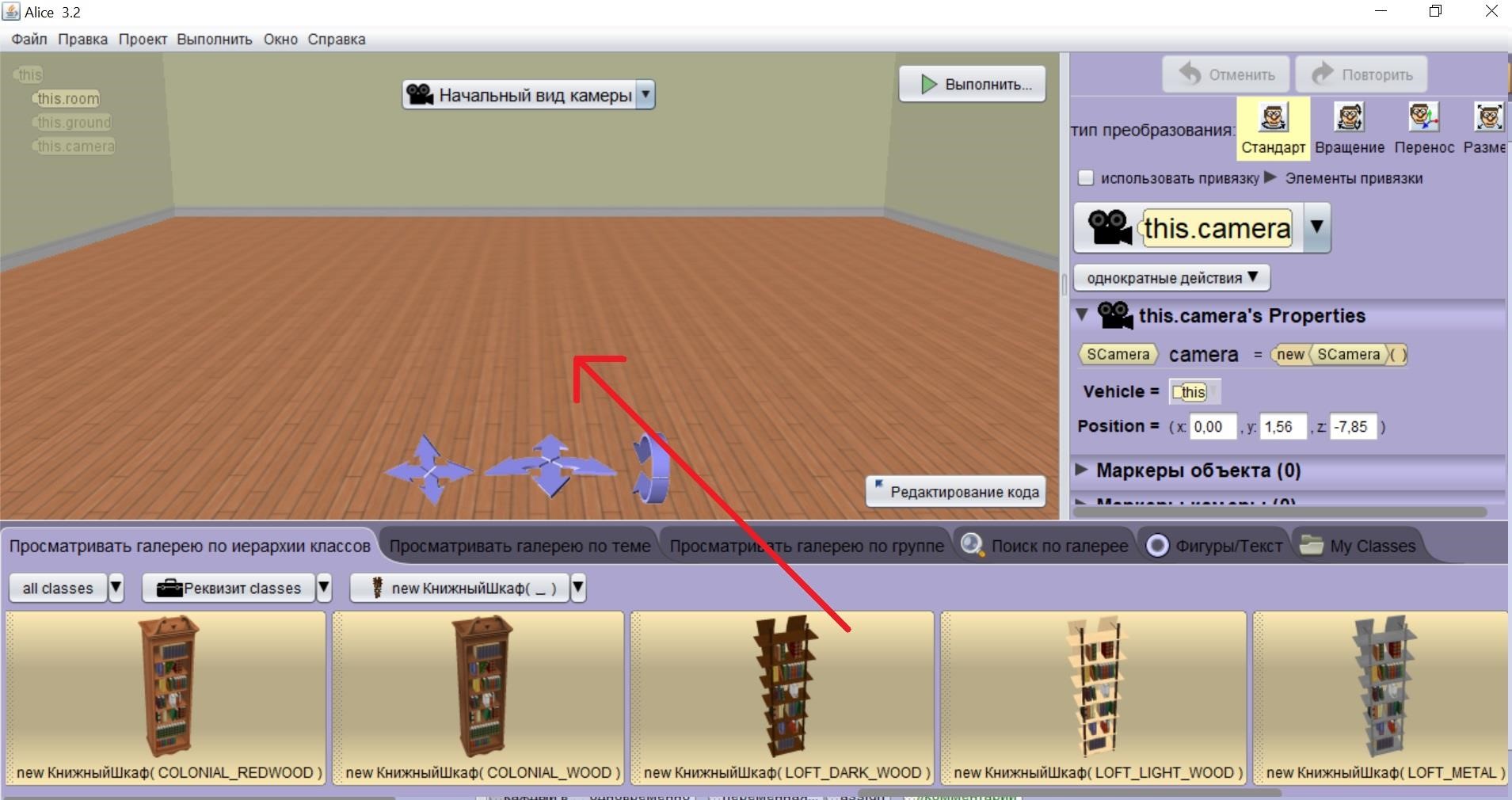
В режиме настройки сцены существует 2 способа добавить объект:

* мышкой перетащите объект из галереи в сцену и заполните диалоговое окно. Объект появится там, где вы его разместили
* кликните на объект мышкой, заполните диалоговое окно. Объект будет размещен по центру сцены

В программировании класс – это модель, используемая для создания объекта, а объект – это частный случай («экземпляр») класса. После того, как объект добавлен в сцену, он становится частным случаем объекта. Вы можете добавить в одну сцену множество экземпляров одного объекта (например, множество кораллов в воду). Каждый такой экземпляр должен иметь уникальное наименование.

## Добавление объекта в сцену

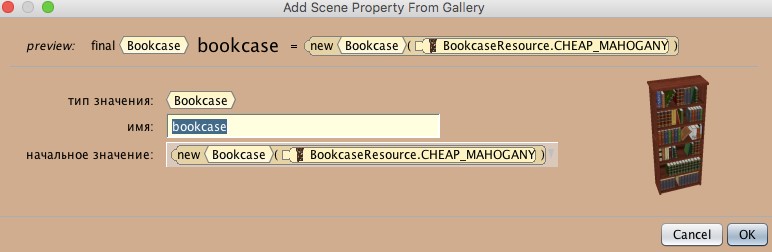
Нажмите на объект мышкой и перетащите его из галереи в сцену



## Название объекта

Если предложенное в Alice наименование объекта вам не

подходит, измените его и нажмите «ОК» Вы можете написать название на русском языке



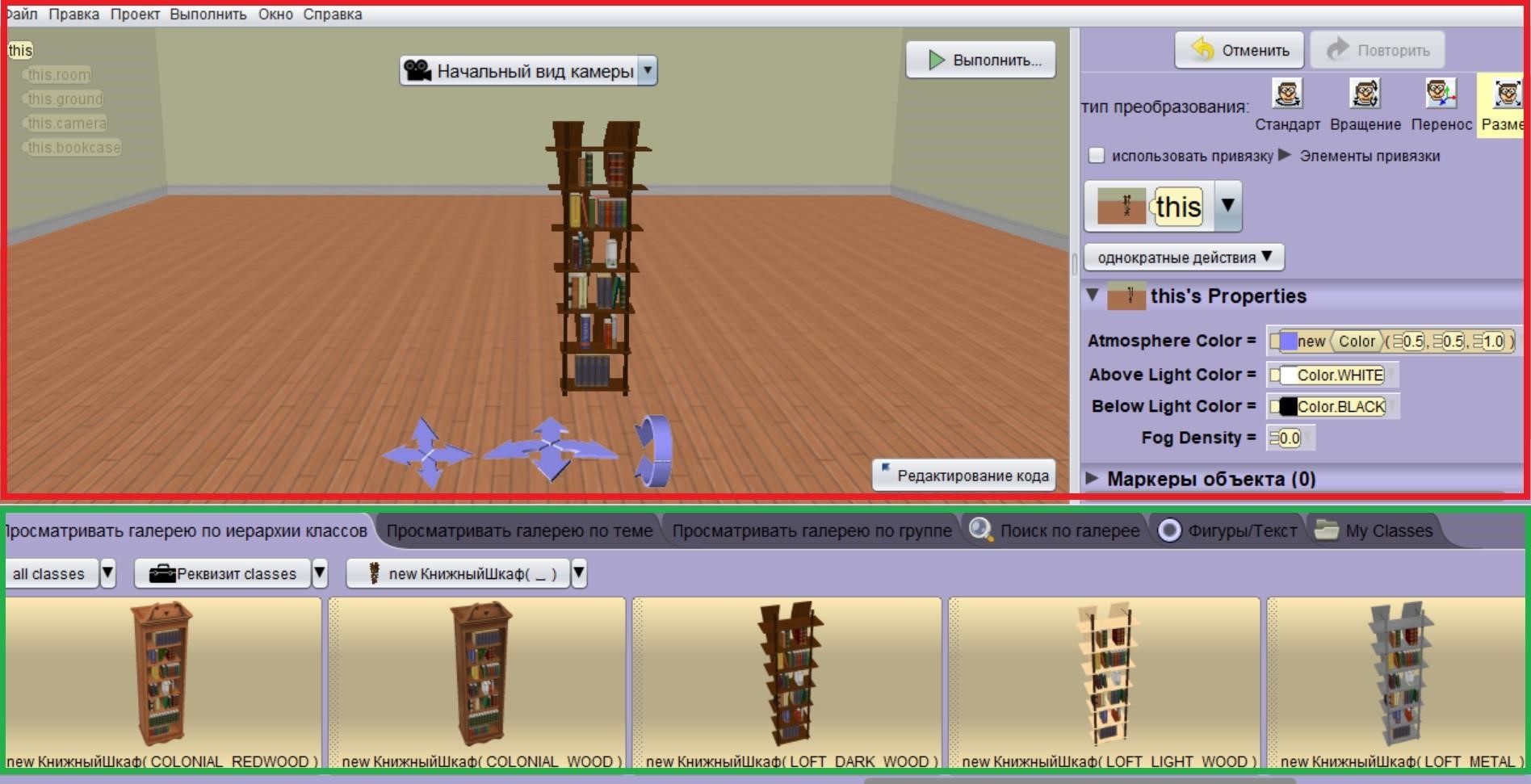
## Настройка сцены

В режиме «Настройка сцены» вы можете:

* выбрать объект из галереи, чтобы добавить его в сцену
* выбрать позицию объекта, используя панель инструментов снизу
* изменить свойства объекта, используя панель инструментов справа
* открыть режим редактирования кода
* запустить созданный проект путем нажатия кнопки «Выполнить» Режим настройки сцены состоит из двух панелей: – настройка сцены сверху

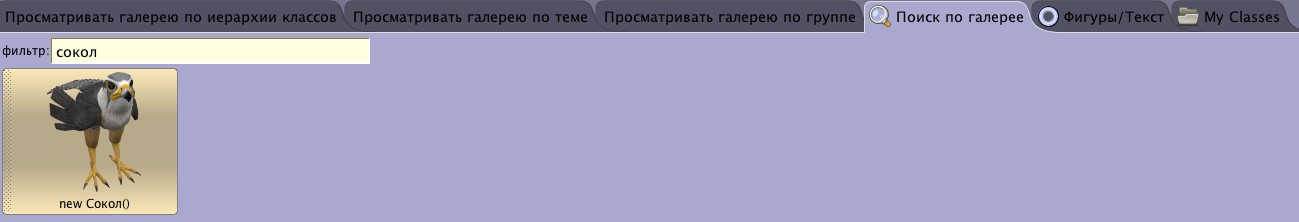
–

галерея снизу



## Галерея

* Галерея – это коллекция трехмерных объектов, которые вы можете добавлять в сцену
* В галерее есть несколько вкладок
* Чтобы найти объект, переходите по вкладкам или воспользуйтесь поиском по галерее (поиск можно осуществлять в том числе на русском языке)



Вкладки в галерее

|  |  |
| --- | --- |
| **Вкладка** | **Функция** |
| Просматривать галерею по иерархии классов | Расположение объектов по мобильности (способу перемещения) |
| Просматривать галерею по теме | Расположение объектов по местности или фольклорному контексту |
| Просматривать галерею по группе | Расположение объектов по категориям |
| Поиск по галерее | Позволяет искать объект по названию |
| Фигуры/текст | Объекты по форме, 3D-текст, доски для объявлений |
| My classes | Позволяет добавить внешние классы в ваш проект |

## Выбор класса

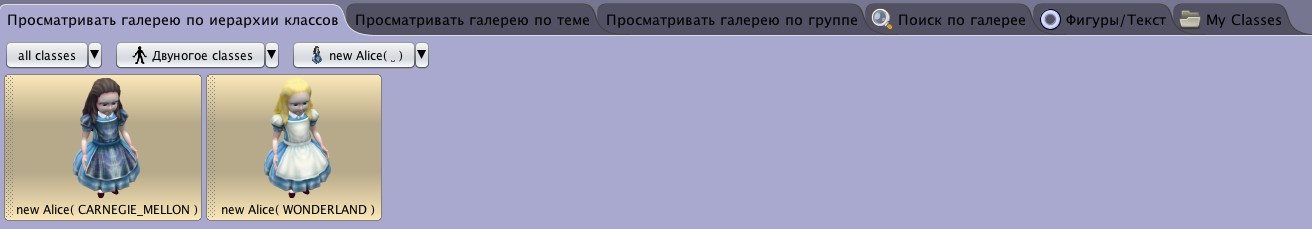
Во вкладке «Просматривать галерею по иерархии классов» объекты сгруппированы по способу передвижения (двуногое, четвероногое, птица и т.д.)

Класс содержит инструкции, которые определяют внешний вид и манеру передвижения объекта. У всех объектов внутри одного класса есть общие свойства. В Alice 3 у каждого класса есть инструкции для создания объекта и его отображения, когда объект добавляют в сцену.



## Примеры классов

* Класс может содержать подклассы (пример на скриншоте)
* Класс «Alice» – это подкласс класса «Двуногие». В галерее два подкласса для класса «Alice»
* Каждая Алиса, добавленная в сцену, наследует общие свойства объектов «Двуногие»: две ноги, подвижные суставы и т.д.



## Сохранение нового проекта

* Создание разных версий одного проекта поможет вам сэкономить время
* После того, как объекты добавлены в сцену, сохраните несколько версий вашего проекта, дав каждой версии разное название

Преимущества создания разных версий проекта:

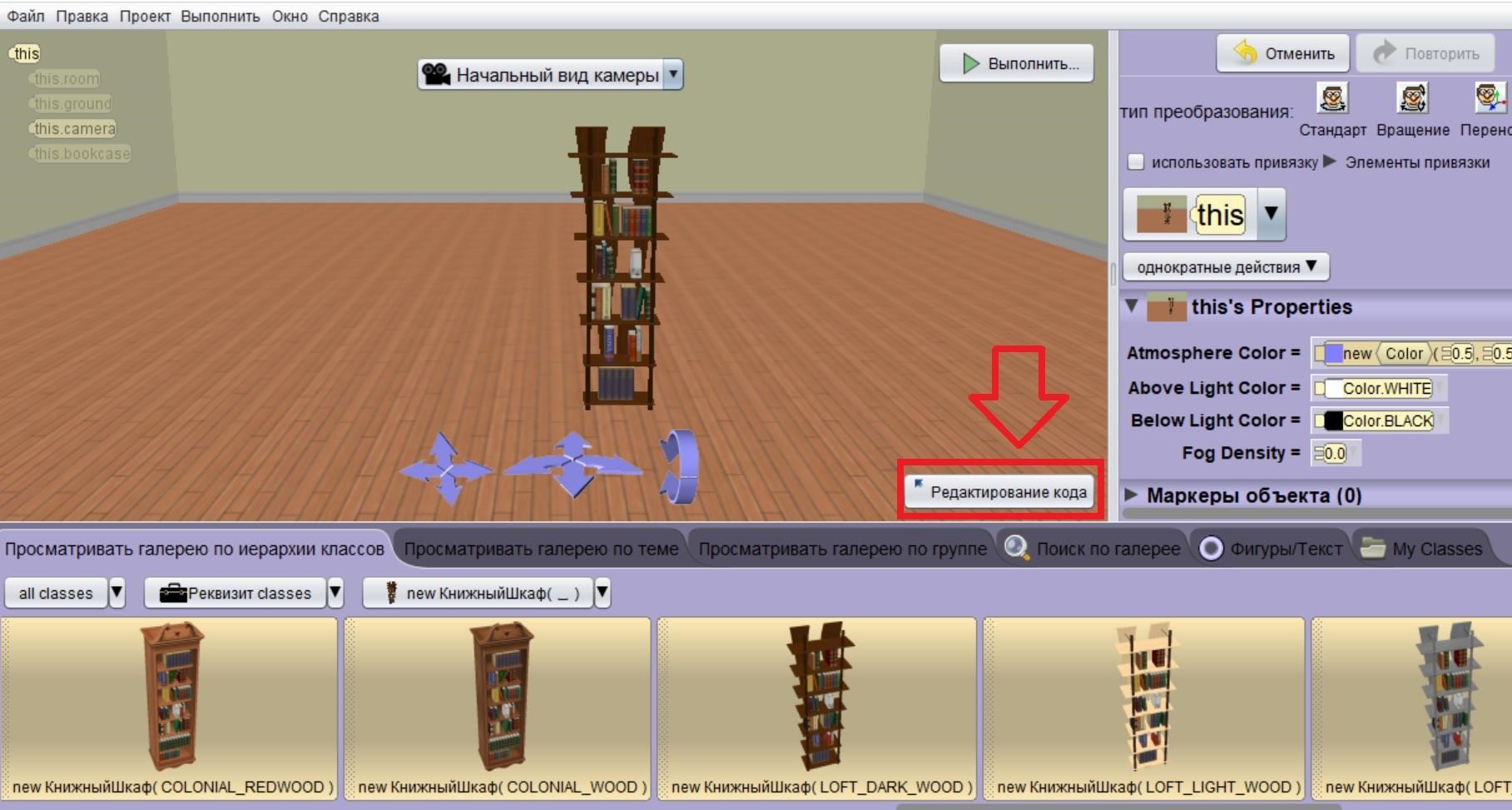
* возможность использования одной сцены для создания разных анимаций;
* экономия времени в случае возникновения программных ошибок.

## Как сохранить проект

* В верхнем меню выберите «Сохранить как…»
* Выберите папку для сохранения (рабочий стол, диск D и т.п.)
* Укажите название проекта
* Нажмите «Сохранить»
* Сохраняйте проект регулярно на случай возникновения системных ошибок компьютера или программного обеспечения

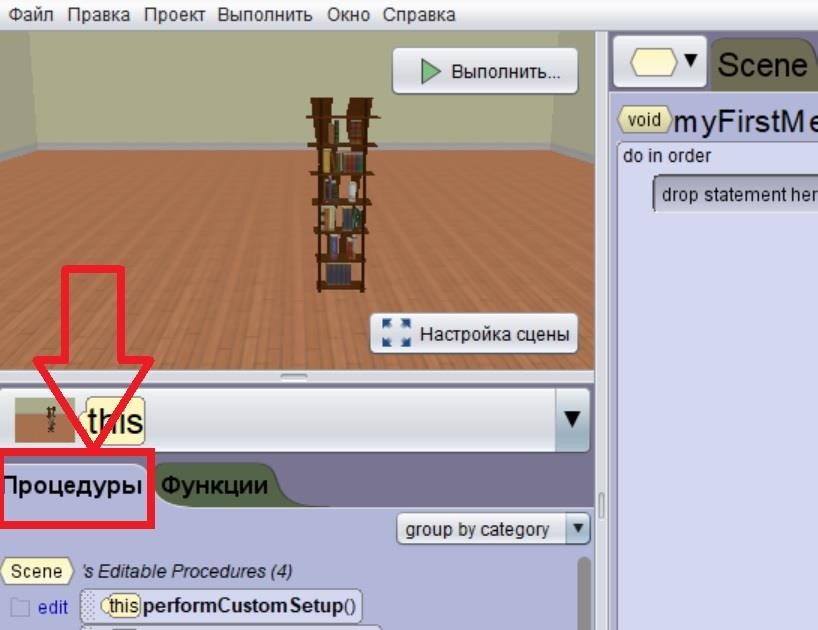
## Режим редактирования кода

Нажмите кнопку «Редактирование кода», чтобы перейти в этот режим. Здесь вы можете вводить инструкции, чтобы программировать вашу анимацию.



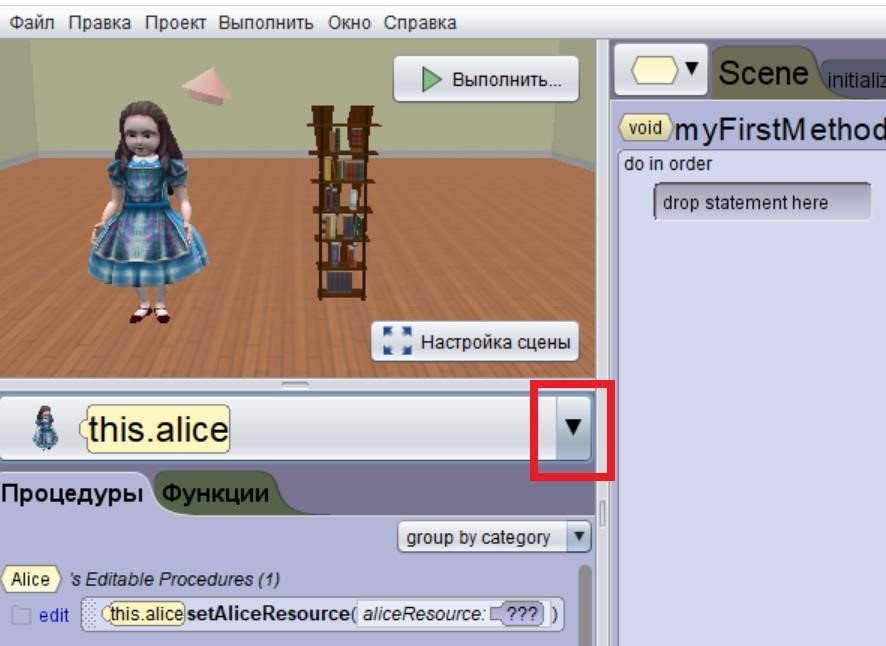
## Панель «Процедуры»

Панель «Процедуры» отображает предварительно определенные критерии для выбранного объекта (экземпляра), а также критерии всего класса объектов данного типа. Процедуры – это кусок программного кода, который определяет, как объект должен выполнять задачу. В Alice 3 существует набор процедур для каждого класса, а также пользователь может создать новые процедуры.



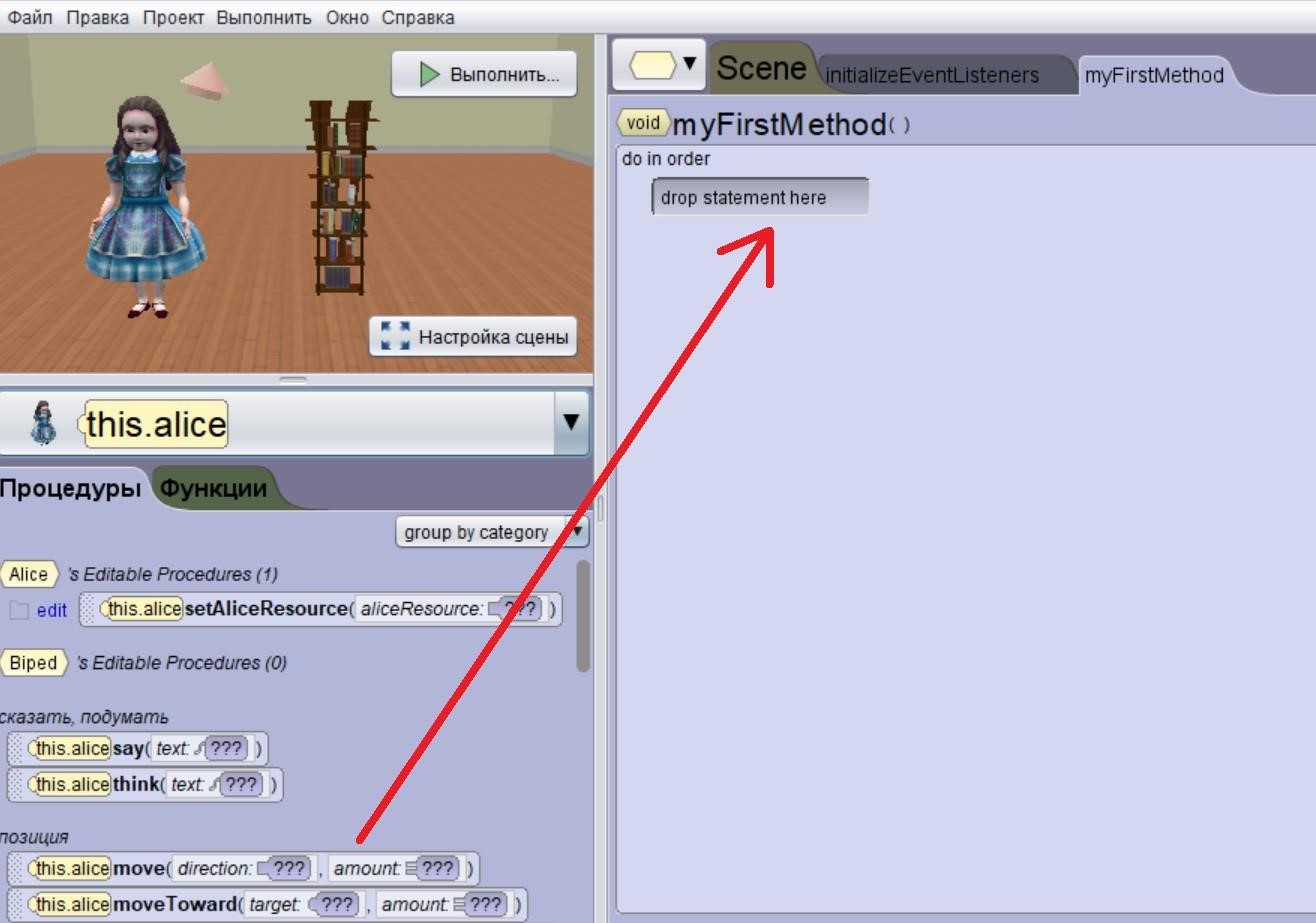
## Меню объектов

Здесь вы можете выбрать нужный объект нажатием на значок в виде перевернутого треугольника



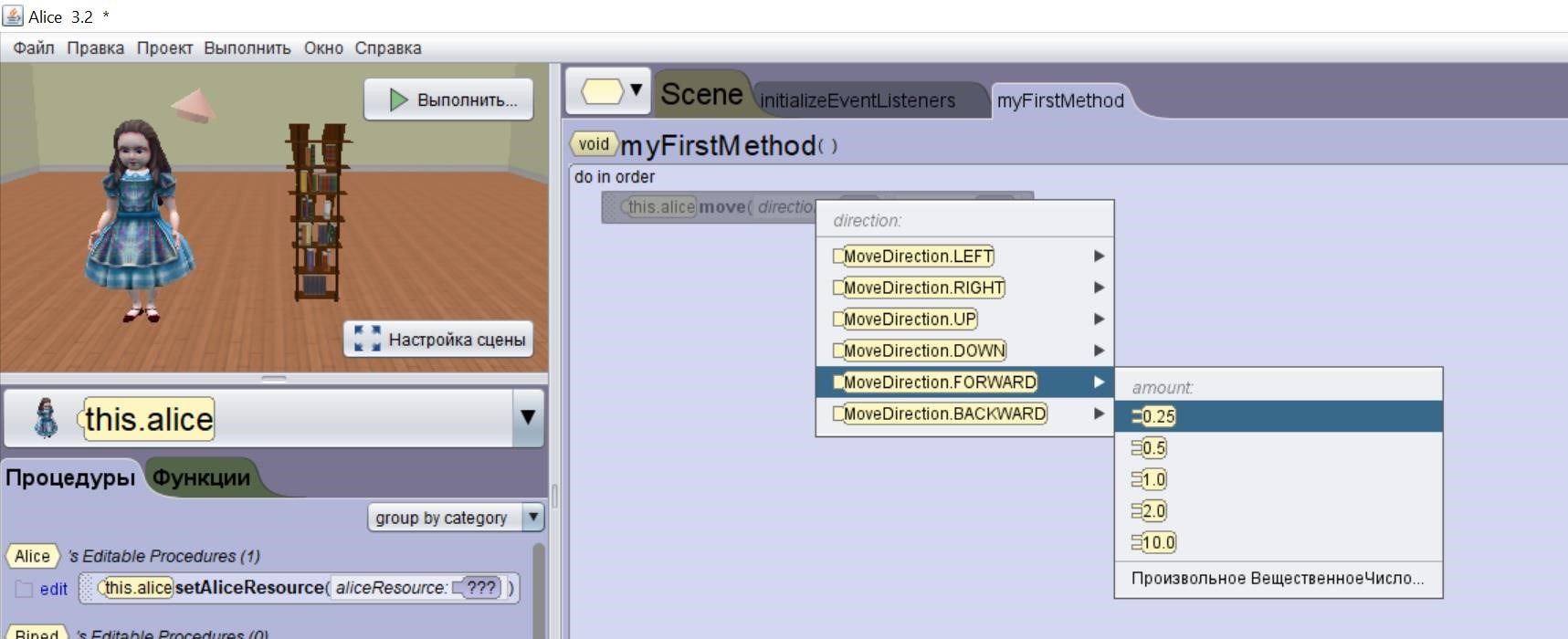
## Создание программных инструкций

Нажмите и перетащите из левой панели нужную команду во вкладку «My first method»



## Выбор значений

После того, как вы перетащили программную инструкцию во вкладку «My first method», используйте выпадающее меню, чтобы выбрать нужное значение для каждого используемого аргумента.



## Типы аргументов

Аргументы могут быть следующих типов:

* направление
* количество
* длительность
* текст

В Alice 3 вы можете увидеть, как много аргументов необходимо для каждой программной инструкции.

## Копирование программных инструкций

Чтобы скопировать программные инструкции, вы можете использовать один из следующих методов:

Ctrl + перетащить

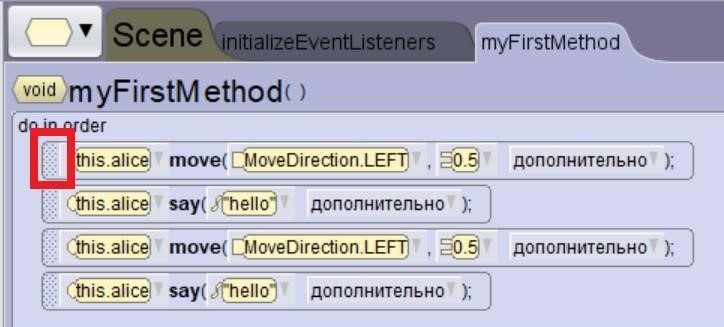
Клик правой кнопкой мышки, выбрать «Скопировать в буфер обмена»

Кликнуть и перетащить в буфер обмена

## Ctrl + перетащить

Зажмите кнопку Ctrl на клавиатуре

Нажмите на квадратик слева от программного кода и удерживайте

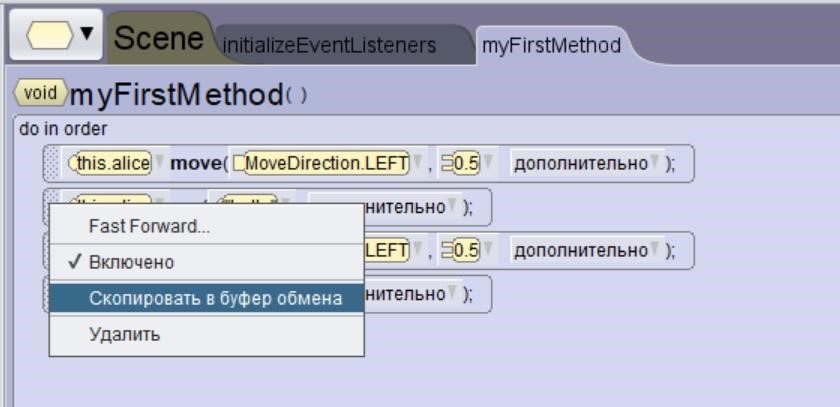


Перетащите квадратик в нужное место в коде. Отпустите кнопку мыши до того, как отпустите кнопку Ctrl

## Команда «Скопировать в буфер обмена»

Нажмите правой кнопкой мыши на команду и выберите

«Скопировать в буфер обмена»



## Как перетащить программную инструкцию в буфер обмена

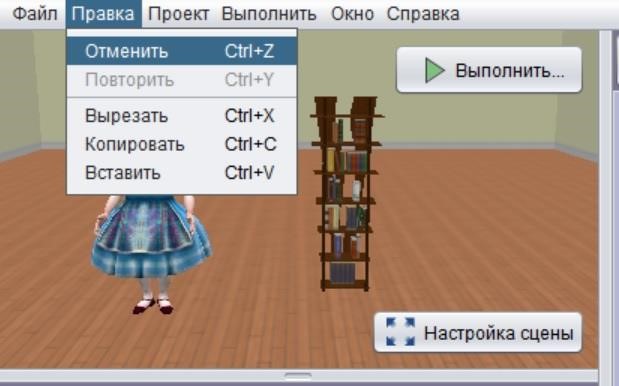
Зажмите и перетащите программную инструкцию на значок буфера обмена. Значок изменит цвет, когда вы перетащите на него программную инструкцию. Используйте этот метод для переноса программных инструкций между вкладками.

Буфер обмена может содержать несколько программных инструкций. На значке отображается количество сохраненных программных инструкций.



## Отмена действия

Отменить действие можно нажатием клавиш Ctrl + Z или выбрав в меню «Правка» команды «Отменить»

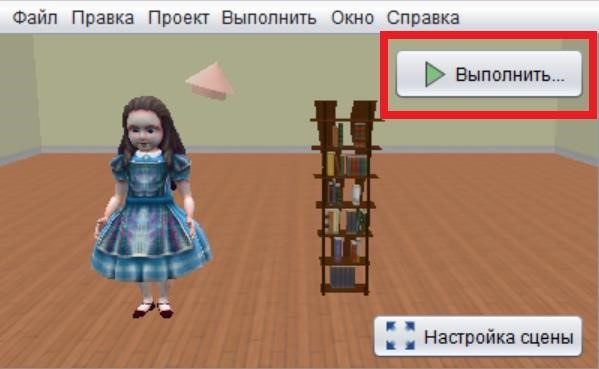


## Тестирование и внесение правок

Как только вы создали программную инструкцию, можно посмотреть результат.

Чтобы посмотреть результат, нажмите на кнопку «Выполнить». Посмотрите созданную анимацию, чтобы убедиться, что все выглядит так, как вы планировали и без ошибок.

Во время создания анимации рекомендуем проверять результат как можно чаще.



## Тестирование и внесение правок

Отладка программы – это цикл, который включает в себя: тестирование программы, идентификация ошибок или непредусмотренного вами результата, внесение правок в код, повторное тестирование.

Программирование ПО, такого как анимация, тестируется путем внесения непредусмотренных команд и попыток «сломать» код. Если что-то ломается или начинает работать неправильно, это называется баг «bug». Debugging – это процесс поиска багов в работе ПО.

## Тестирование и внесение правок

Используйте следующие методы в процессе создания анимации в

Alice 3:

* уточните такие аргументы, как направление движения, дистанция и продолжительность движения объекта.
* уточните или измените аргументы, если что-то работает не так, как задумано.

4. Подведение итогов. 7 мин.

5. Домашнее задание. – 3 мин.