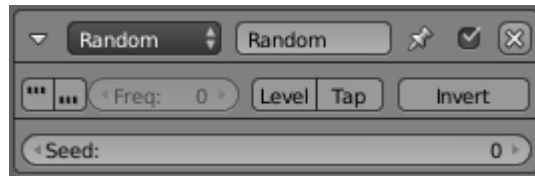


Какая игровая логика может обойтись без случайных чисел? Согласитесь, что игра становится не интересной, если противник движется предсказуемо. Именно случайность поведения и рождает интригу и спортивный интерес. Поэтому нам просто необходимо познакомиться с двумя моментами: модулем Python с названием Random и сенсором Random. Первый генерирует псевдо случайное число. А второй посылает положительный импульс на контроллер через псевдо случайное число кадров.

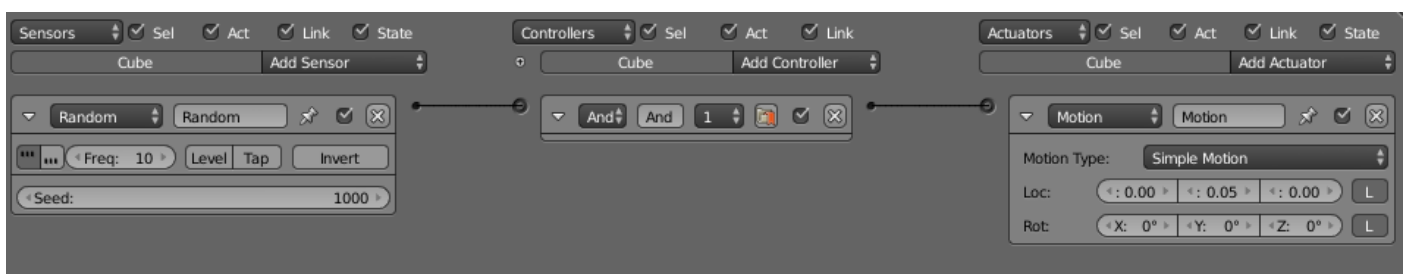
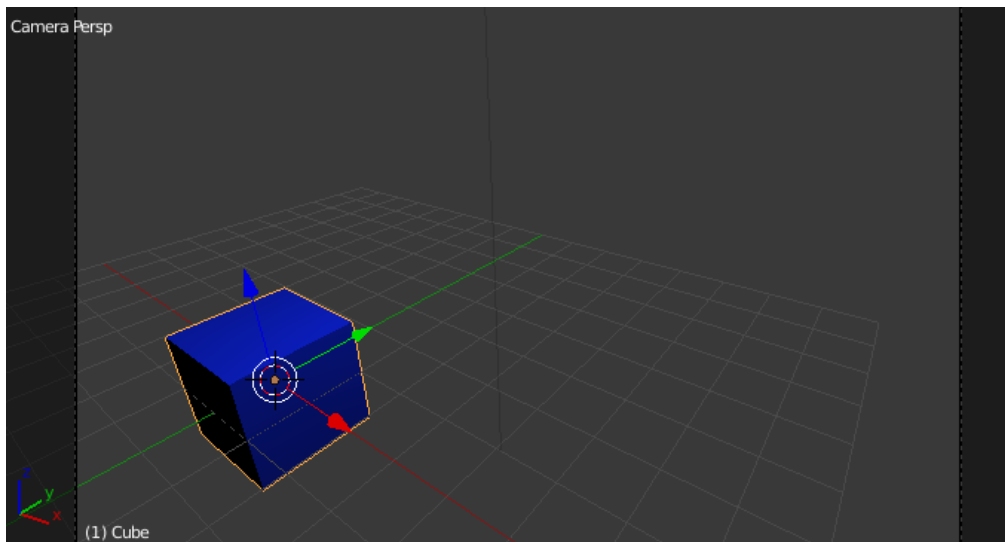
Начнём с сенсора. Выглядит он так:



Seed задаёт интервал между 0 и 1000. Если вы выберете 10, то случайное число будет генерироваться в пределах от 0 до 10 и т.д. Генерация происходит постоянно. Если активировать первую кнопку, то можно задать режим задержки в кадрах (Freq):



Создадим простую сцену и заставим двигаться наш куб с произвольным интервалом времени:



Если запустить проект, то куб будет прерывисто двигаться по оси Y. Интервалы в движении будут псевдо случайными (далее просто «случайными»). Почему «псевдо»? Да потому, что в компьютере даже генерация случайного числа вовсе не случайна, и подчиняется определённым законам.

Хорошо, а если мне нужен не случайный интервал, а случайное число? Проще всего этого добиться, написав скрипт на Python. Подключим модуль:

```
import bge
import random
```

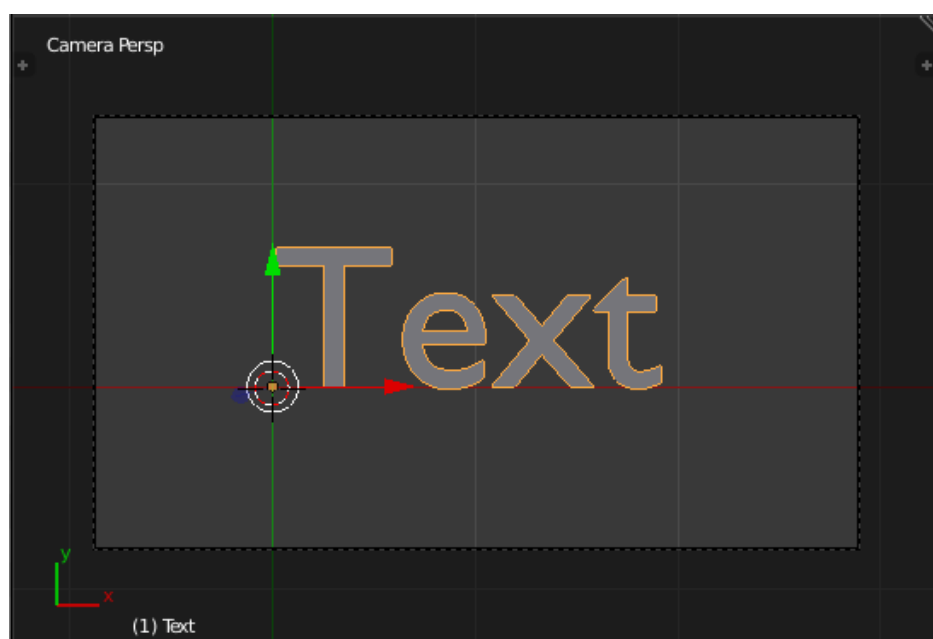
Получим случайное число... А вот здесь есть множество вариантов 😊. Откройте справку Python и убедитесь сами 😊. Хотя, на мой взгляд, нам достаточно и двух. В первом случае мы получим случайное число с плавающей точкой от 0.0 до 1.0 :

```
# Задаёт дробное число от 0.0 до 1.0
# Если есть переменная, то она должна быть Float
ran1 = random.random()
```

Во втором случае мы получим целое случайное число от 0 до 999 (если задать 1000):

```
# Задаёт целое случайное число от 0 до 999
# Если есть переменная, то она должна быть Integer
ran1 = random.randrange(1000)
```

Создадим простую сцену с текстовым объектом и камерой над ним:



Напишем скрипт, под названием Random. Все пояснения в нём есть:

```
import bge
import random # подключаем модуль случайных чисел
cont = bge.logic.getCurrentController() # Получить контроллер
scene = bge.logic.getCurrentScene()# получить текущую сцену
objList = scene.objects # получить список объектов
obj = cont.owner # получить объект, прикрепленный к контроллеру

text = objList["Text"] # получаем текстовый объект

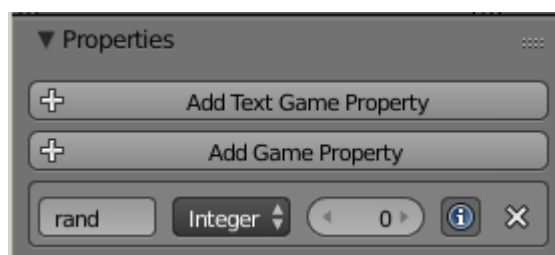
# Задаёт целое случайное число от 0 до 999
# Если есть переменная, то она должна быть Integer
ran1 = random.randrange(1000)

# Задаёт дробное число от 0.0 до 1.0
# Если есть переменная, то она должна быть Float
#ran1 = random.random()

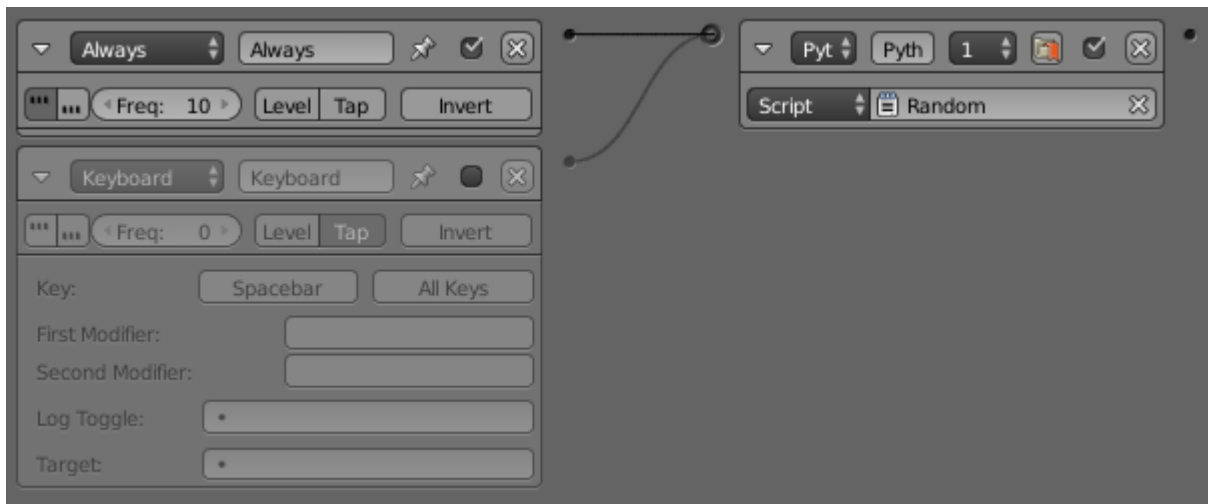
# Если переменная не нужна то её можно вообще удалить
obj["rand"] = ran1 # назначаем переменной значение ran1

text.text = str(ran1) # выводим значение в текстовом объекте
```

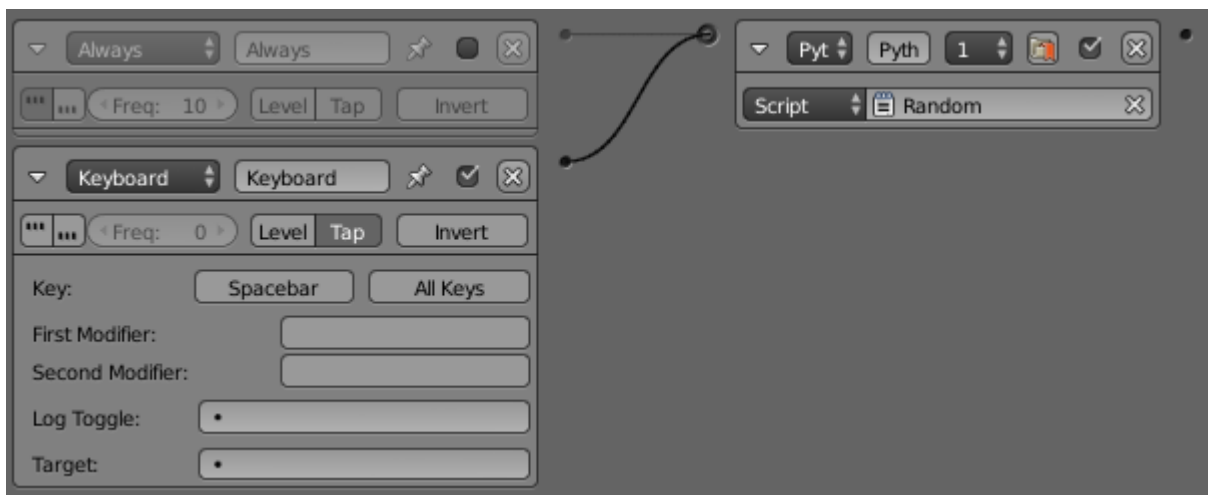
Выделив текстовый объект, в игровой логике добавим свойство (просто переменную):



Для дальнейших экспериментов нам понадобится два сенсора: Always и Keyboard. Первый необходим для проверки постоянной генерации случайных чисел либо с интервалом. Второй, для проверки генерации по нажатию клавиши. Естественно подключаем контроллер Python. В зависимости от надобности мы будем включать либо один сенсор, либо другой. Сначала включим Always:



После запуска на экране будут случайным образом меняться цифры, с интервалом в 10 кадров. Теперь выключим Always и включим Keyboard:



Смена сгенерированного числа будет происходить по нажатию клавиши пробела. Конечно, для демонстрации можно было вообще отказаться от свойства rand. Но в дальнейшем оно нам очень понадобится 😊. Поэтому я заранее показал, как его задействовать. Вот и всё. Исходник на странице «Готовые проекты». В следующем уроке попробуем устроить соревнования между кубиками 😊

13 июня 2015 года.

Составил [Niburiec](#) для сайта <http://blender-game.ucoz.ru>