ЛЕКЦИЯ 12

Блюпринты в действии 3

ЦЕЛИ И ИТОГИ ЛЕКЦИИ

Goals

Цели этой лекции:

- Представить разнообразие функций Blueprint
- Объяснить функцию Set Input Mode Game And UI
- Показать, как создать случайную точку в ограниченном боксе
- Объяснить ноду Select

Outcomes

К концу этой лекции вы сможете

- Использовать операцию по модулю (%) и функцию Clamp
- Изменять материал с помощью функции Blueprint
- Изменять режим ввода
- Использовать ноду Select с разнообразными типами индексов

МОДУЛЬ (%)

Операция **modulo** представленная символом процента возвращает остаток от деления.

Ввод

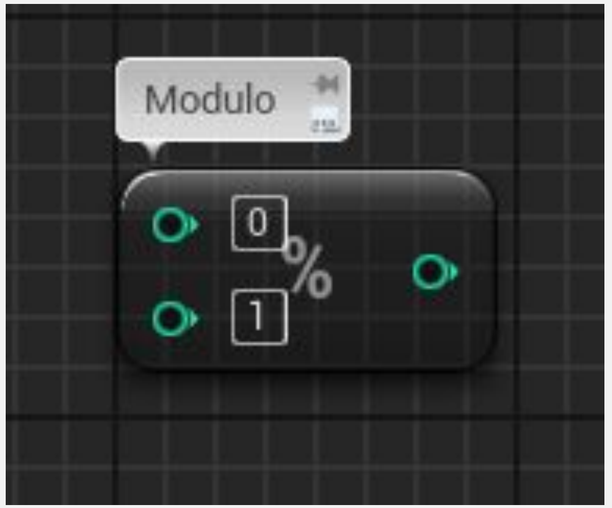
- **A**: Значение, которое будет делиться.
- **B**: Значение, которое будет использоваться как делитель.

Вывод

- **Return Value**: Остаток деления.

Пример

1 % 3 = 1
2 % 3 = 2
3 % 3 = 0
4 % 3 = 1
5 % 3 = 2
6 % 3 = 0

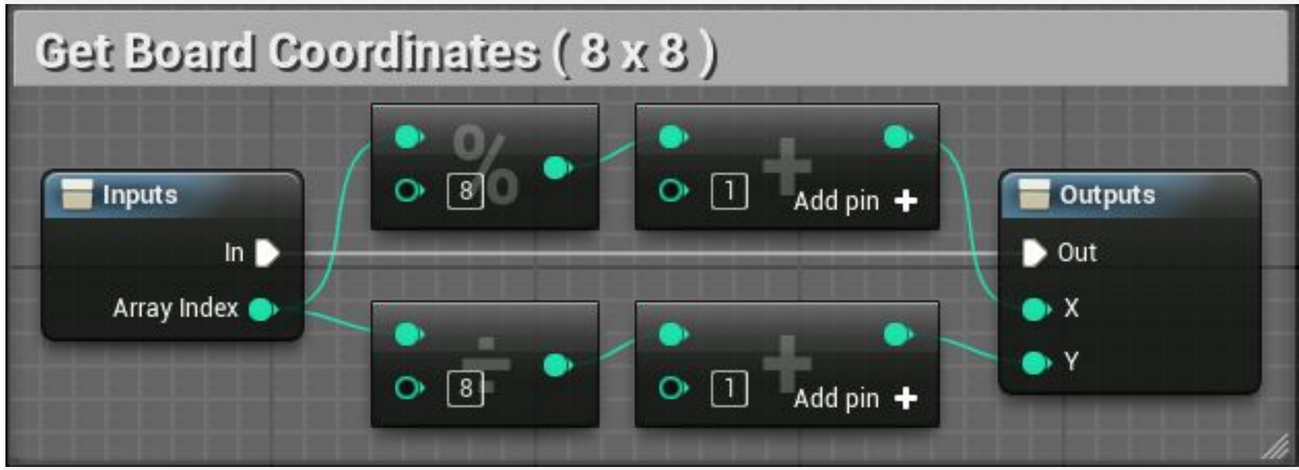


МОДУЛЬ (%): ПРИМЕР

Операция по модулю обычно используется для преобразования индекса одномерного массива в двумерные координаты.

В примере справа массив целых чисел используется для представления шахматной доски. Массив содержит 64 позиции, поэтому его индексы варьируются от «0» до «63». Каждая позиция доски представлена координатой (X, Y).

Первая позиция имеет координаты (1, 1), а последняя позиция имеет координаты (8, 8). Макрос, показанный в примере, преобразует индекс массива в координаты X и Y доски.



ФУНКЦИЯ CLAMP

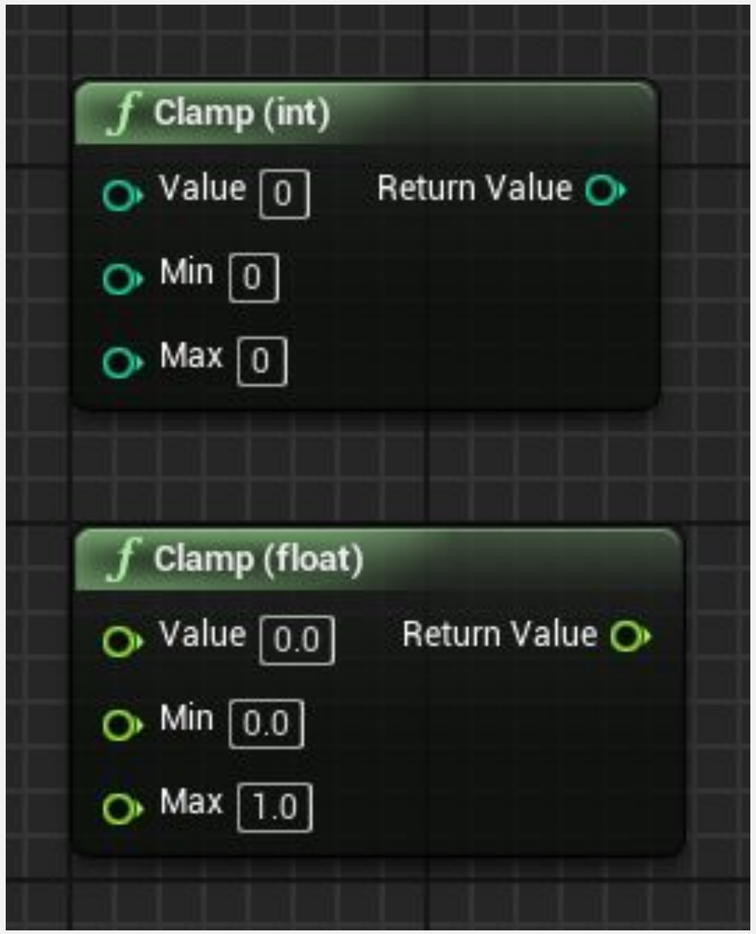
Функция **Clamp** получает входное значение, а также минимальное и максимальное значения. Если входное значение находится между минимальным и максимальным значениями, оно возвращается без изменений.Если оно ниже минимального, возвращается минимальное значение. Если оно выше максимального, возвращается максимальное значение.

Ввод

- **Value**: Входное значение, которое будет сравниваться со значениями **Min** и **Max**.
- **Min**: Максимальное значение, которое может быть возвращено.
- **Max**: The highest value that can be returned.

Вывод

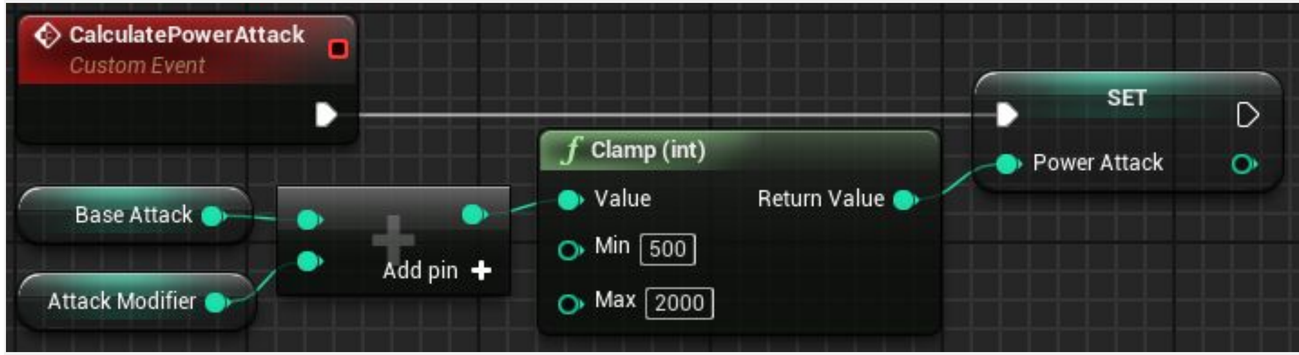
- **Return Value**: значение между значениями **Min** и **Max**.



ФУНКЦИЯ CLAMP: ПРИМЕР

В примере справа есть событие, которое вычисляет значение силовой атаки персонажа. Это значение основано на сумме значений переменных **Base Attack** и **Attack Modifier**. Переменная **Attack Modifier** может отображать бонусы атаки или иметь отрицательное значение, отображающее какое-то плохое состояние персонажа.

Но независимо от того, насколько плохи штрафы за атаку персонажа, минимальное значение силовой атаки составляет «**500**». Точно так же максимальное значение силовой атаки составляет «**2000**», даже если сумма значений базовой атаки и бонуса атаки выше.



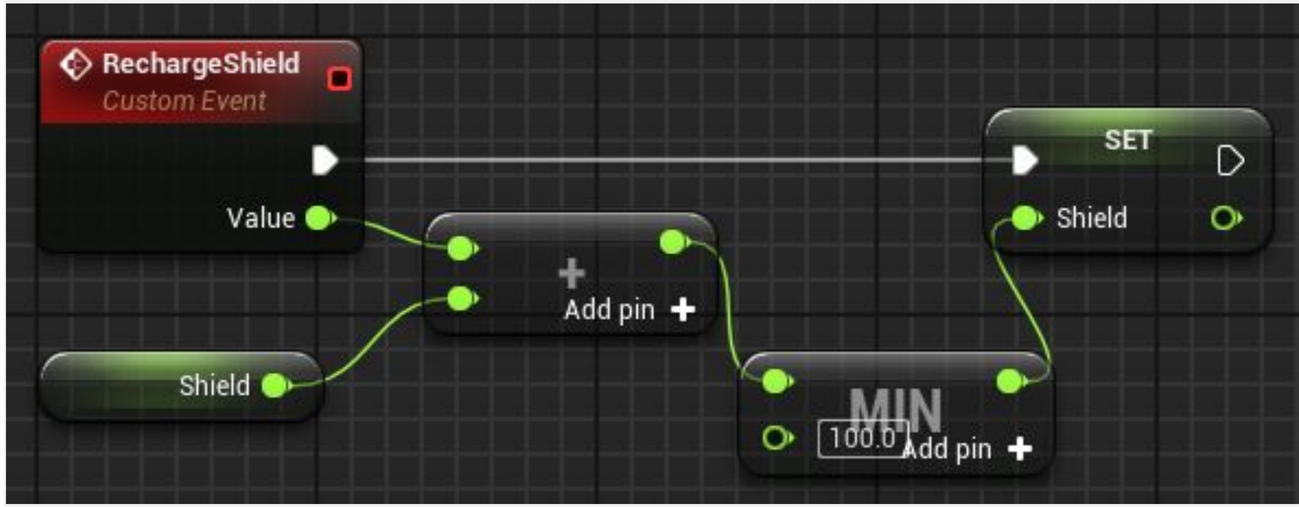
ФУНКЦИИ MIN И MAX

Функция **Min** возвращает минимальное значение между входными значениями. Функция **Max** возвращает максимальное значение между входными значениями.

Многие входные значения могут быть добавлены в обе функции с помощью кнопки **Add pin**.

Пример

На нижнем изображении справа функция **Min** используется для обеспечения того, чтобы значение переменной **Shield** никогда не превышало «**100.0**»..

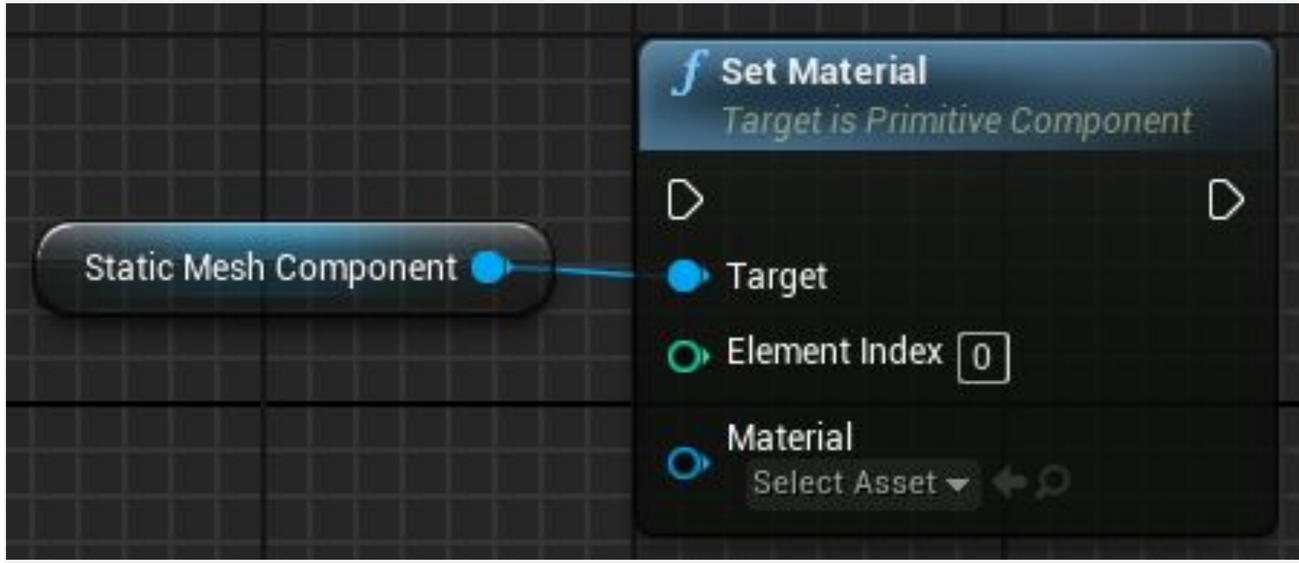


ФУНКЦИЯ SET MATERIAL

Функция **Set Material** меняет материал, используемый компонентом.

Ввод

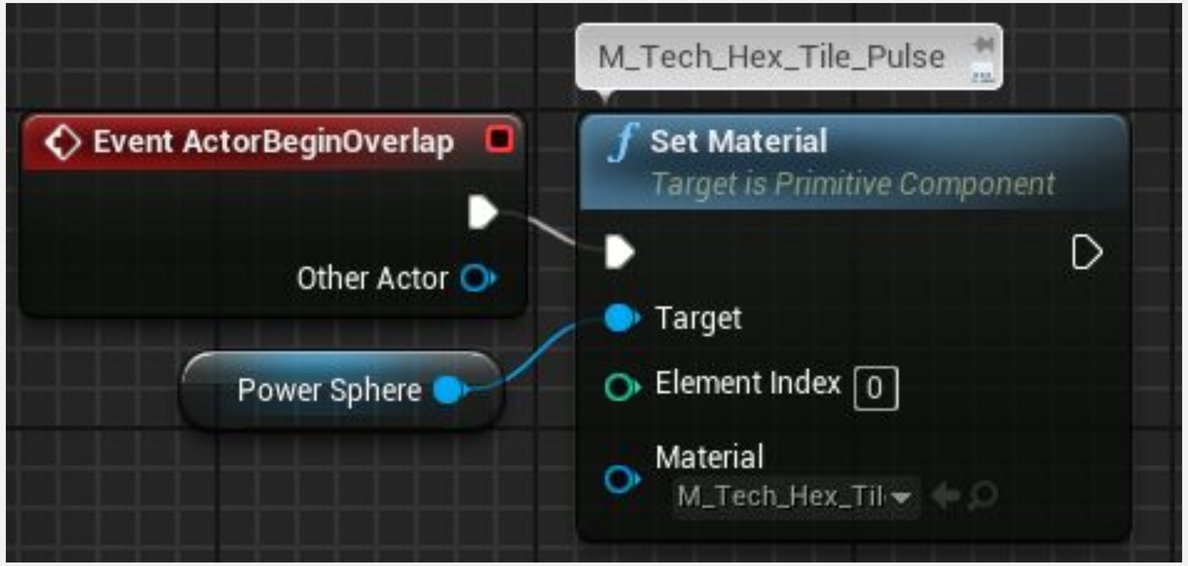
- **Target**: Примитивный компонент, материал которого будет меняться.
- **Element Index**: Отображает какой материал будет изменен у компонента.
- **Material**: Ссылка на новый материал, который будет использоваться.

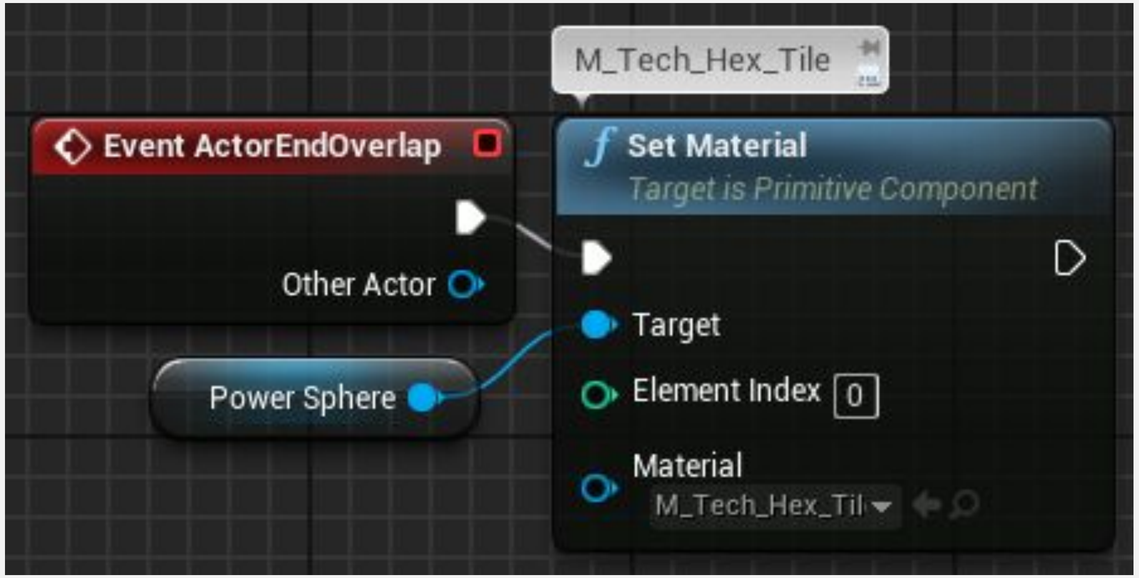


ФУНКЦИЯ SET MATERIAL: ПРИМЕР

В примере справа есть Blueprint, который содержит компонент Static Mesh под названием «**Power Sphere**». Когда Актор приближается к чертежу, материал **Power Sphere** заменяется на **T_Tech_Hex_Tile_Pulse**, чтобы указать, что он активен.

Когда Актор уходит, Материал заменяется **M_Tech_Hex_Tile**, указывая, что компонент отключен.



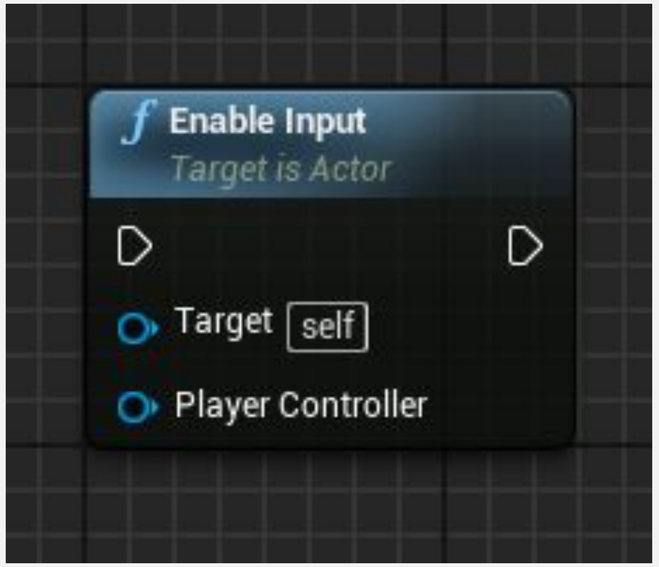


ФУНКЦИЯ ENABLE INPUT

Функция **Enable Input** используется, чтобы позволить Blueprint реагировать на события ввода, такие как клавиатура, мышь и геймпад.

Ввод

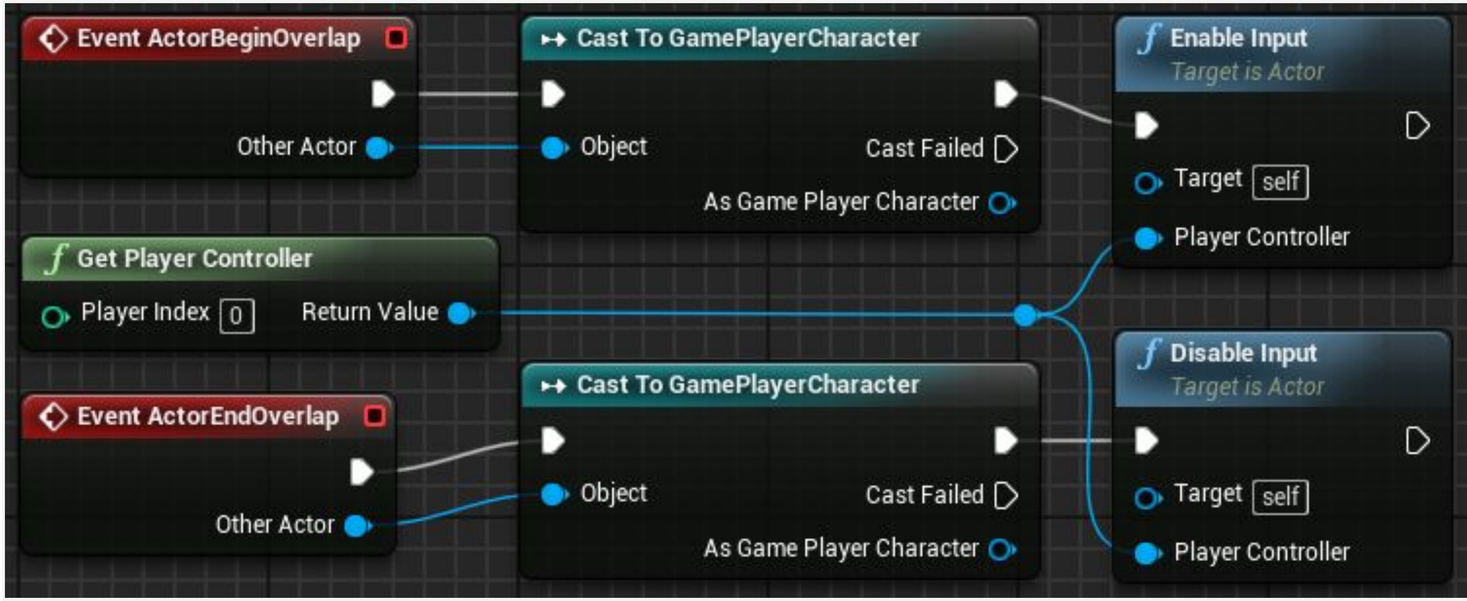
- **Target**: Актор, для которого будет разрешен вход. Значение «**self**» используется для нацеливания на Blueprint, содержащий ноду.
- **Player Controller**: Ссылка на используемый Player Controller.



ФУНКЦИЯ ENABLE INPUT: ПРИМЕР

В примере справа Blueprint реагирует на события ввода, только когда игрок находится рядом с ним. Игрок представлен Blueprint с именем **GamePlayerCharacter**. Когда игрок находится в пределах границ столкновения Blueprint, вызывается функция **Enable Input**.

Когда игрок находится за пределами границ столкновения Blueprint, вызывается функция **Disable Input**, чтобы этот Blueprint больше не реагировал на события ввода.

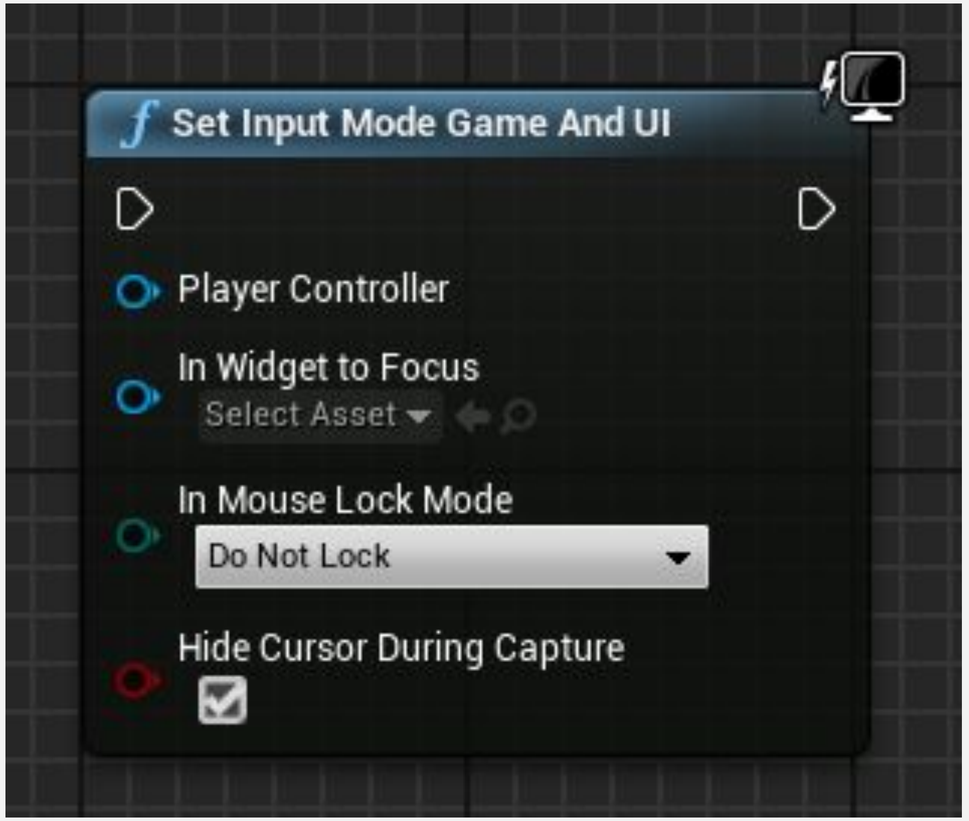


ФУНКЦИЯ SET INPUT MODE GAME AND UI

Функция **Set Input Mode Game And UI** определяет режим ввода, в котором пользовательский интерфейс (UI) имеет приоритет при обработке событий пользовательского ввода. Если пользовательский интерфейс не знает, как обрабатывать конкретное событие, входные данные передаются Player Controller.

Ввод

- **Player Controller**: Ссылка Player Controller.
- **In Widget to Focus**: Ссылка на виджет UMG который будет получать ввод игрока, прежде чем передать его Player Controller.
- **In Mouse Lock Mode**: Отображает, как будет держаться мышка.
- **Hide Cursor During Capture**: Указывает, должен ли быть скрыт курсор мыши.

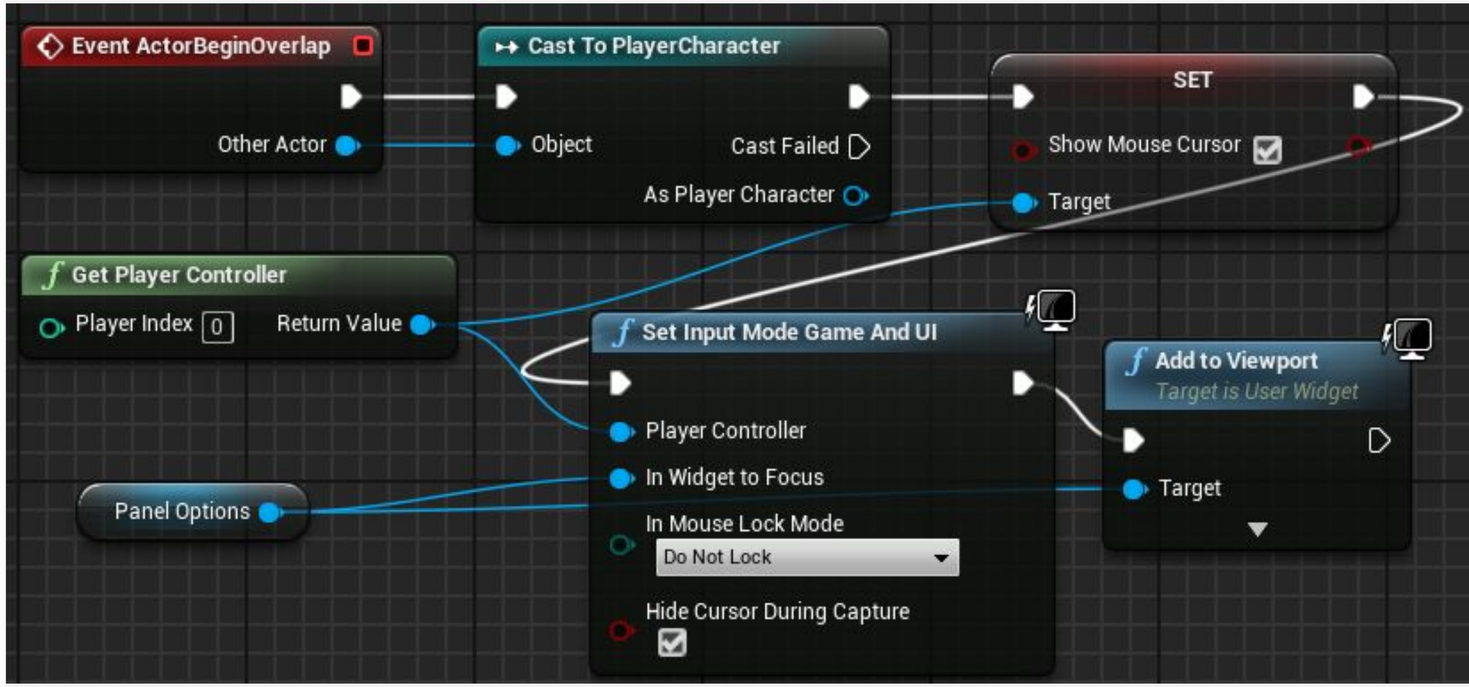


ФУНКЦИЯ SET INPUT MODE GAME AND UI: ПРИМЕР

В примере справа есть Blueprint, который взаимодействует с игроком в игре от первого лица. Когда игрок перекрывает этот Blueprint, отображается пользовательский интерфейс с опциями, которые игрок может выбрать с помощью мыши.

Для свойства **Show Mouse Cursor** Player Controller установлено значение «true», чтобы курсор отображался на экране.

Функция **Set Input Mode Game And UI** заставляет мышь управлять курсором, чтобы он мог взаимодействовать с панелью, а не изменять вид игрока.



ФУНКЦИЯ TELEPORT

Функция **Teleport** перемещает Актора в указанное место. Если в локации есть препятствие, Актор перемещается в ближайшее место, где нет коллизии.

Ввод

- **Dest Location**: Место назначения.
- **Dest Rotation**: Вращение, примененное к Актору.

Вывод

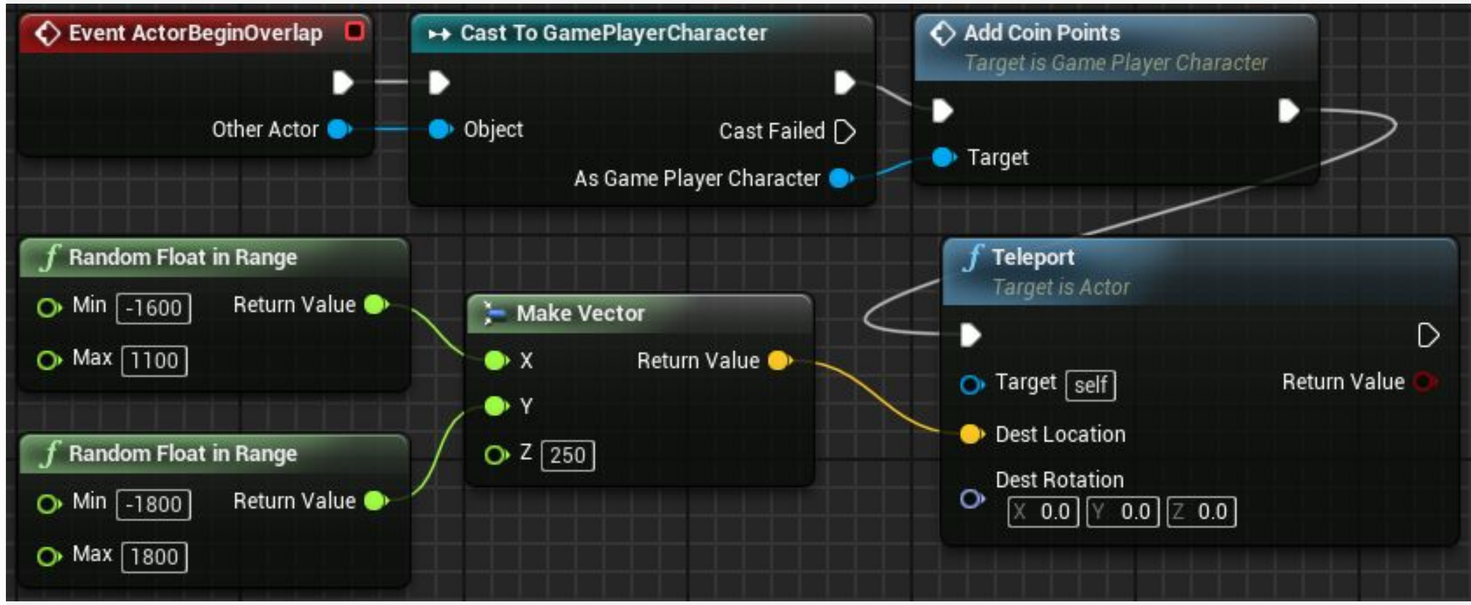
- **Return Value**: Логическая переменная. Если значение “**false**”, Актор не сможет двигаться.

ФУНКЦИЯ TELEPORT: EXAMPLE

В примере справа показана игра, в которой игрок собирает монеты. На изображении показано событие **ActorBeginOverlap** Blueprint, которое представляет монеты.

Когда игрок собирает монету, вызывается функция **Add Coin Points** для регистрации очков, и после этого монета телепортируется в другое место в игровой зоне.

На игровой площадке есть несколько препятствий, но функция телепортации поместит монету в свободное место.



ФУНКЦИЯ RANDOM POINT IN BOUNDING BOX

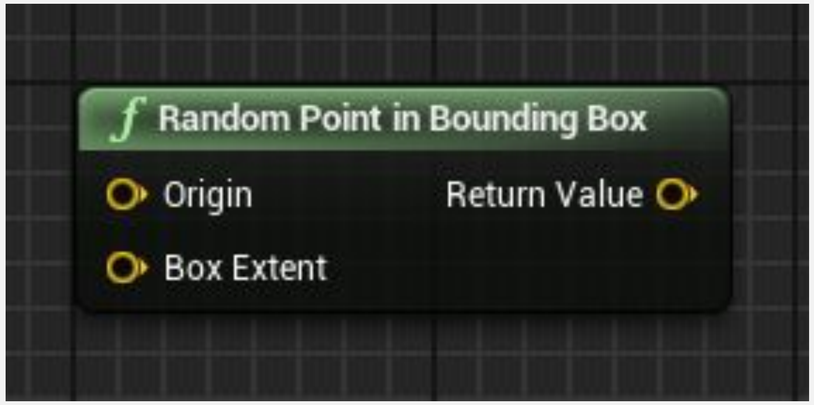
Функция **Random Point in Bounding Box** возвращает вектор, представляющий случайную точку, расположенную в объеме, заданном исходной точкой и Box Extent.

Ввод

- **Origin**: Вектор, представляющий центральную позицию box extent.
- **Box Extent**: Вектор, содержащий размеры бокса, определяющего трехмерный объем.

Вывод

- **Return Value**: Вектор, представляющий точку, которая была случайно сгенерирована внутри заданного Box Extent.

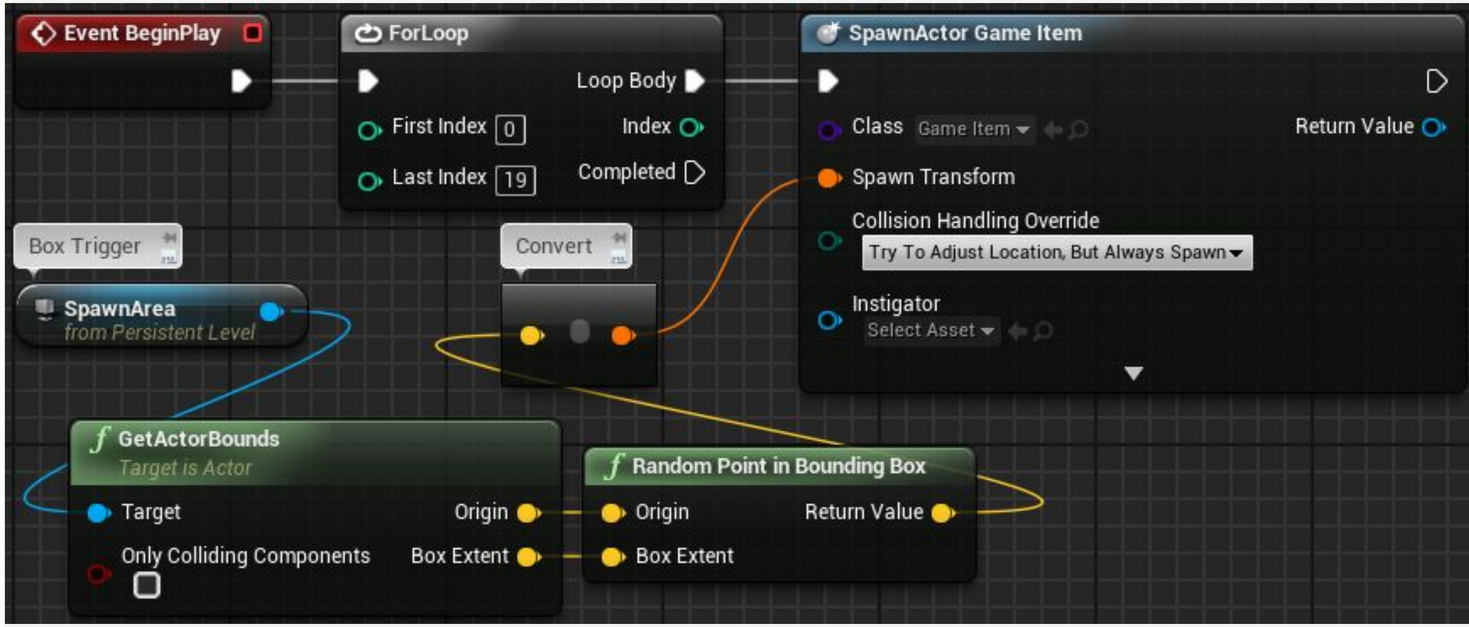


ФУНКЦИЯ RANDOM POINT IN BOUNDING BOX: ПРИМЕР

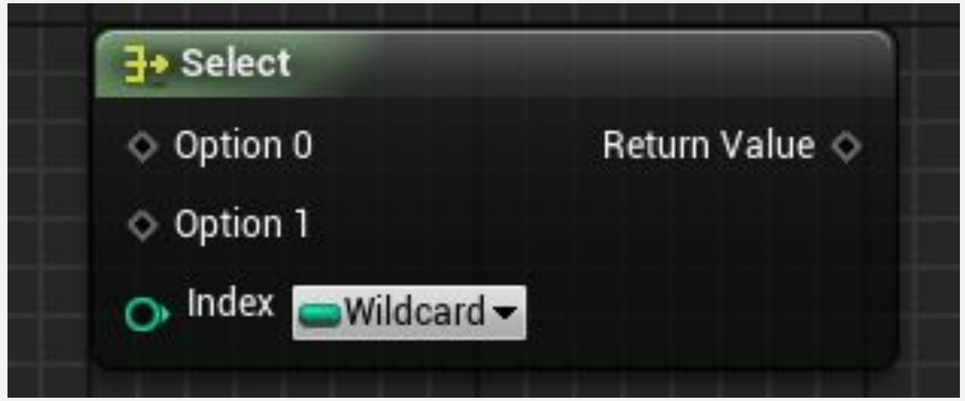
Изображение справа взято из Level Blueprint. Поле-триггер с именем «**SpawnArea**» был помещен на уровень, чтобы определить область, в которой элементы будут создаваться в случайных местах.

Когда игра начинается, нода **ForLoop** выполняет ноду **SpawnActor** 20 раз, чтобы создать 20 элементов в случайных положениях.

Положение каждого элемента определяется функцией **Random Point in Bounding Box** с использованием исходной точки и box extent, указанных для области появления.



НОДА SELECT



Нода **Select** возвращает значение, связанное с параметрами, которое соответствует индексу, переданному в качестве входных данных. Нода **Select** является универсальной и может работать с несколькими типами переменных для индекса и значений параметров. Чтобы изменить тип параметров или индекса, щелкните параметр правой кнопкой мыши и выберите “**Change Pin Type**”. Вы также можете установить тип вывода, перетащив ссылку на переменную или провод на выводы. Чтобы добавить дополнительные параметры, щелкните узел правой кнопкой мыши и выберите “**Add Option Pin**”.

Ввод

- **Option 0, 1,** …: Значения, которые могут быть возвращены нодой **Select.** Варианты могут быть любого типа.
- **Index**: Значение, определяющее возвращаемый параметр. Тип может быть “**Boolean**”, “**Byte**”, “**Integer**”, или “**Enum**”.

Вывод

- **Return Value**: Опция, указанная параметром Index.

НОДА SELECT: ПРИМЕР 1

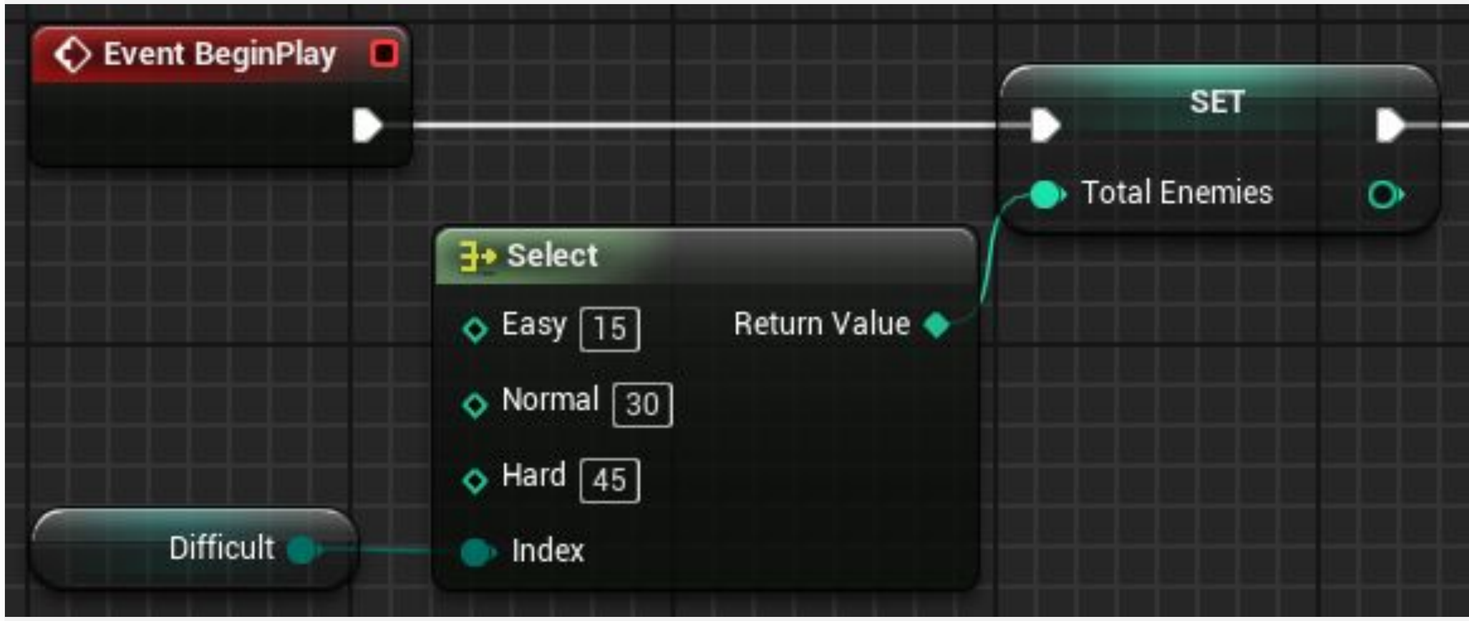
В событии **Respawn** на картинке справа, нода **Select** используется для получения места точки спавна, основанная на значении целочисленной переменной **Spawn Point Index**.

Если значение **Spawn Point Index** является “**0**”, оно вернет “**(0,0,0)**”. Если значение равно “**2**”, оно вернет значение переменной **Spawn Point 2**.

НОДА SELECT: ПРИМЕР 2

На примере справа, индекс это перечисление с именем “**Difficult**” которое содержит значения **Easy**, **Normal**, и **Hard**.

Тип опций это “**Integer**” и представляет количество врагов в соответствии со значением перечисления переменной **Difficult**.



ИТОГ

Эта лекция рассказала о различных функциях Блюпринтов.

Она показала как менять режим ввода, создавать случайные точки, менятяь материал, и использовать ноду Select.