

Присваиваем модификатор сразу нескольким объектам с помощью Python

В данной заметке рассказывается, как назначить и настроить модификатор сразу нескольким объектам, что может быть удобно, когда этих объектов OVER9000 (не присваивать же каждому модификатор, настраивать и повторять эту процедуру ещё 8999 раз?).

Работа с Python рассматривается на примере симулятора дыма, но эти принципы позволят работать практически со всеми свойствами объектов.

Естественно, предполагается наличие хотя бы базовых знаний языка Python. Поехали.

Для начала импортируем модуль **bpy**:

```
import bpy
```

Модуль bpy (от **B**lender **P**Ython) необходим для взаимодействия с Блендером.

Далее нужно получить список выделенных в данный момент объектов. Делается это с помощью контекста — он как раз и позволяет получить доступ не к определённому объекту по имени, а ко всем объектам, выделенным в данный момент. Создаём переменную **objects** и присваиваем ей список всех выделенных объектов:

```
objects = bpy.context.selected_objects
```

Теперь запускаем цикл, который перебирает все выбранные объекты (напомню, их список — в переменной **objects**) и с каждым из них производит нужные действия.

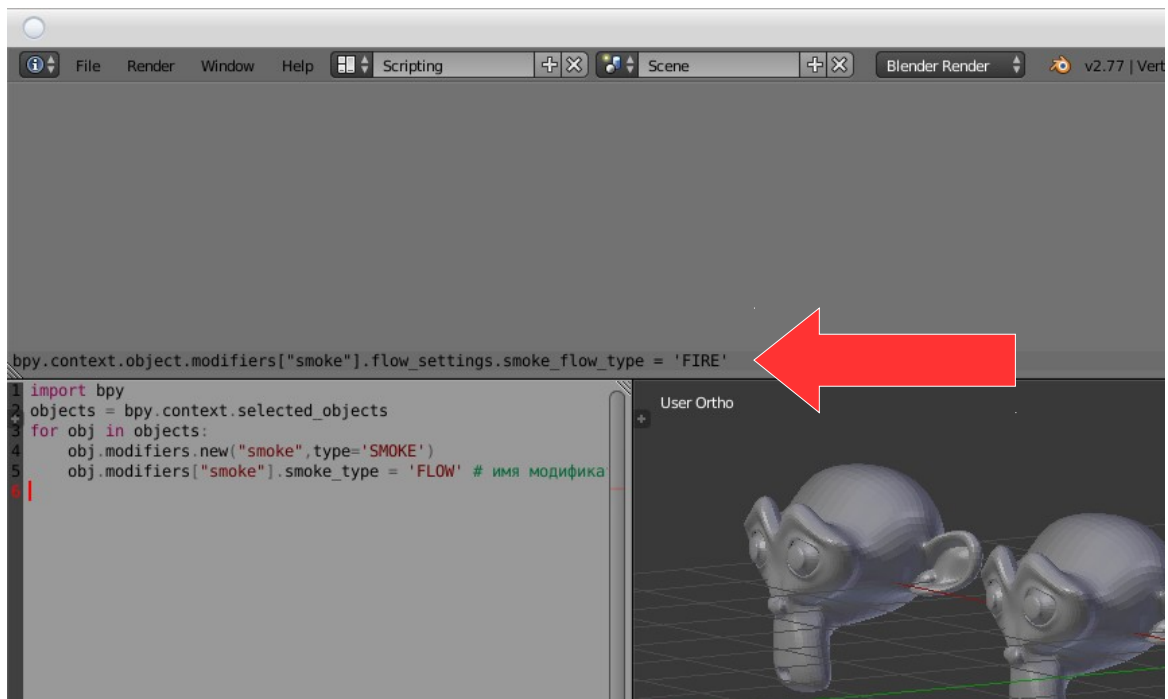
```
for obj in objects:  
    obj.modifiers.new("smoke", type='SMOKE')  
    obj.modifiers["smoke"].smoke_type = 'FLOW'
```

Что делает этот код:

0. Сначала каждому объекту добавляется модификатор дыма. Параметры в скобках — имя и тип. Заметим, что имя чувствительно к регистру.
1. Потом устанавливается тип объекта, в данном случае это FLOW — то есть объект испускающий дым (есть ещё, например, домен).

Вопрос: а как добавить объекту некое свойство, о котором в том или ином уроке подробно не написано? Например, если надо, чтоб вместо дыма был огонь?

Чтобы узнать, что прописывать в коде, потребуется окно информационных сообщений Блендера. Открыть оно можно, оттянув вниз верхний header (где располагаются меню «File», «Render», «Window», «Help») или переключив раскладку (layout) в режим «Scripting», где всё нужное будет открыто сразу. Попробуем переключить «Smoke» на «Fire» — как обычно, мышкой. Смотрим в окно сверху и видим следующее:



Теперь понятно, что происходит при переключении режима дыма. Следовательно, в цикл нужно добавить

```
obj.modifiers["smoke"].flow_settings.smoke_flow_type = 'FIRE'
```

Таким образом можно узнать код для очень многих операций.

В результате весь код будет выглядеть следующим образом:

```
import bpy
objects = bpy.context.selected_objects
for obj in objects:
    obj.modifiers.new("smoke", type='SMOKE')
    obj.modifiers["smoke"].smoke_type = 'FLOW'
    obj.modifiers["smoke"].flow_settings.smoke_flow_type = 'FIRE'
```

После нажатия Run Script на все выделенные объекты будет повышен симулятор дыма с заданными свойствами.