EasyShelf

Краткое руководство пользователя



Что это такое

Плавающая панель кнопок поверх редакторов Blender, в духе Maya shelf. У каждого режима и редактора — свой шелф со своим набором кнопок: Object Mode, Edit Mode, Sculpt Mode, UV Editor, Shader Editor, Geometry Nodes. Каждый шелф можно показывать панелью, радиальным пай-меню на хоткей или и тем и другим. Каждая кнопка по клику выполняет свой Python-код.

Установка и обновление

Ставится прямо из Blender: Edit > Preferences > Get Extensions, найти EasyShelf и нажать Install. Страница аддона: extensions.blender.org/add-ons/easy-shelf. Нужен Blender 4.2 или новее.

Обновления тоже приходят через Blender. Если включён Allow Online Access (Preferences > System > Network), Blender показывает, что вышла новая версия, и обновляет её в один клик. Кнопки и настройки сохраняются.

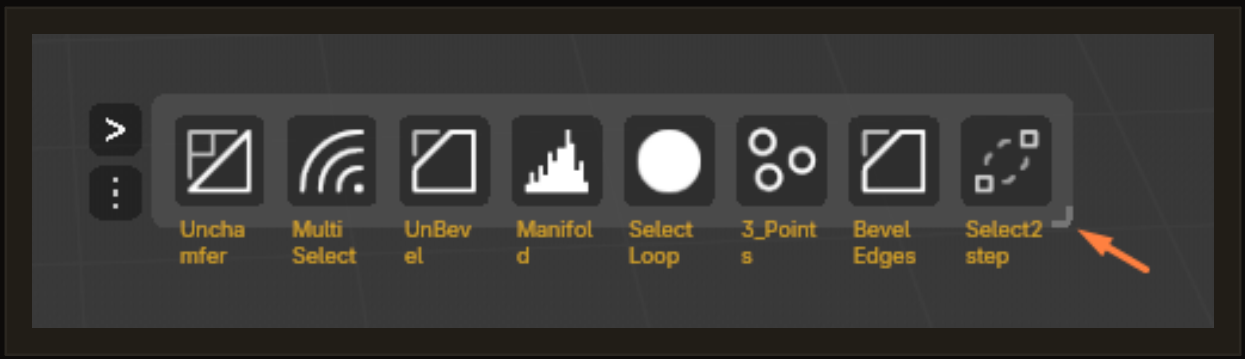
Ориентация и перемещение



«>» и ручка перетаскивания слева от панели

Слева от панели — «>» (переключает горизонтальный/вертикальный режимы) и ручка из трёх точек (перетаскивание мышью). Позиция сохраняется автоматически. При переключении переключатель остаётся под курсором, поэтому шелф не «убегает» от мыши.

Размер шелфа



Уголок для изменения размера (отмечен стрелкой)

У правого нижнего угла панели (в вертикальном режиме — у левого нижнего) проходит едва заметная дуга по контуру угла. При наведении она подсвечивается, а если зажать ЛКМ и потянуть, шелф меняет размер: от угла панели — больше, к нему — меньше. Дуга при этом становится оранжевой. Размер сохраняется — это тот же ползунок Shelf Size, что и в настройках.

Разделители и сворачиваемые секции



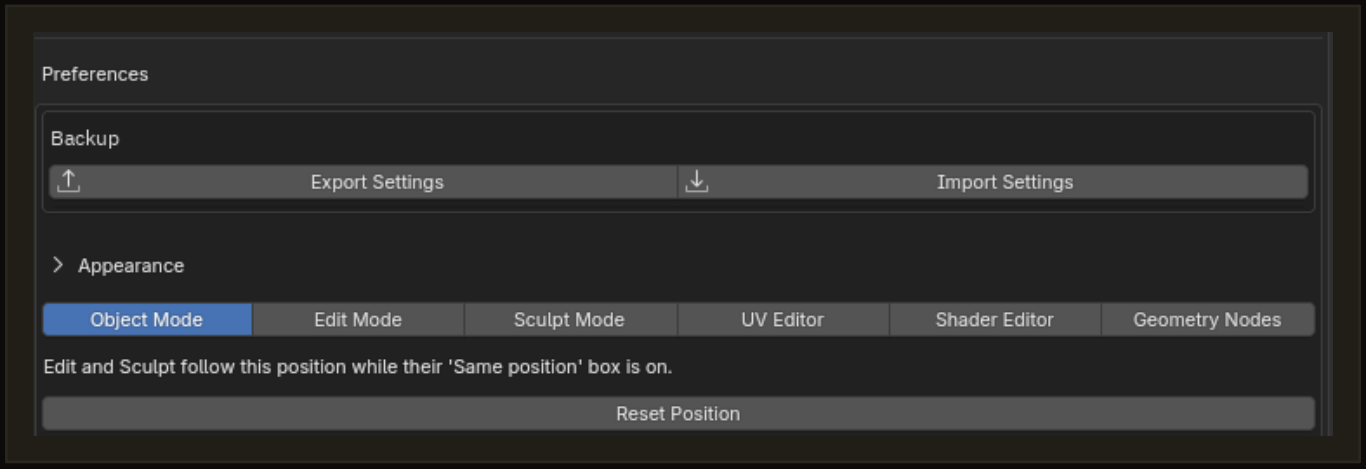
Секции раскрыты; одна свёрнута; обе свёрнуты

Разделитель — тонкая черта между группами кнопок. ПКМ по кнопке шелфа, Add Separator, затем клик в промежуток, куда его поставить. Move переставляет кнопку или разделитель так же, Delete удаляет.

Разделитель работает и как «спойлер»: клик по нему прячет все кнопки после него до следующего разделителя. Разделитель при этом становится короче, показывая, что внутри что-то скрыто. Ещё клик — кнопки возвращаются. Состояние сохраняется.

Разделители — не кнопки: у них нет номера, поэтому нумерация идёт 1, 2, 3... сквозь них, и в пай-меню они не попадают. Сворачивание действует только на панель, пай-меню показывает все включённые кнопки.

Свой шелф для каждого контекста



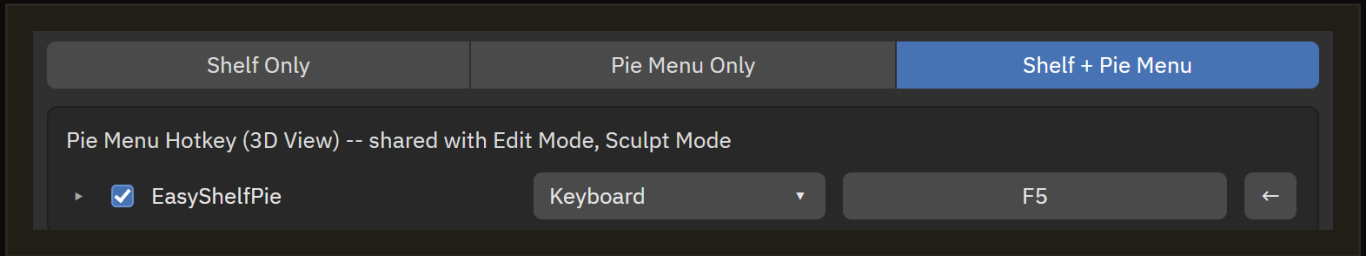
Переключатель контекстов, Enable, Reset Position

В Preferences переключатель контекстов выбирает, какой шелф вы настраиваете: общий Shelf (Object Mode) или Sculpt, UV, Shader, Geometry Nodes. У каждого контекста своя галочка включения, свой список кнопок, свой режим отображения и своя позиция.

Для Edit Mode и Sculpt Mode по умолчанию включено «Same position as Object Mode»: шелф не прыгает при переключении режимов в 3D-вьюпорте. Снимите галочку, если хотите разместить его отдельно.

«Reset Position» возвращает шелф на место по умолчанию — вверх по центру — и показывает его, если он был скрыт хоткеем.

Режим отображения и хоткей



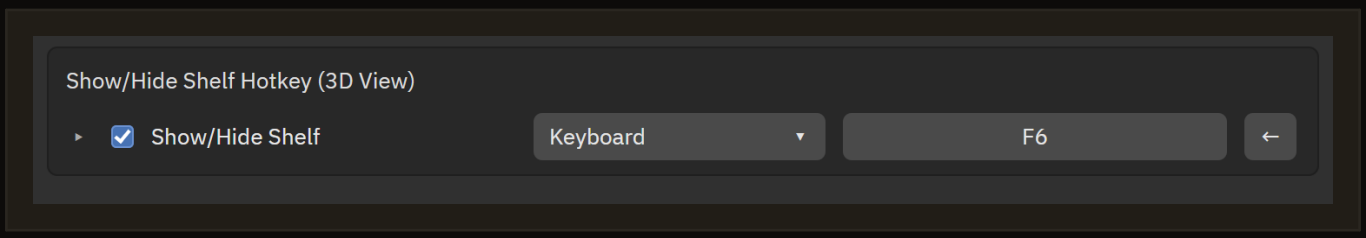
Режим Shelf / Pie / Both и хоткей пай-меню

Для каждого контекста выбирается режим: Shelf (только панель), Pie (только пай-меню) или Both (и то и другое).

Пай-меню всегда повторяет шелф текущего контекста: в Edit Mode — кнопки Edit-шелфа, в UV Editor — UV-шелфа. Порядок функций совпадает с порядком в списке настроек.

Хоткей назначается отдельно для каждого редактора: 3D-вьюпорт (общий для Object, Edit и Sculpt Mode), UV Editor и Node Editor (общий для Shader Editor и Geometry Nodes). Вы можете использовать один и тот же хоткей для всех пай-меню либо назначить разные.

Скрытие шелфа хоткеем

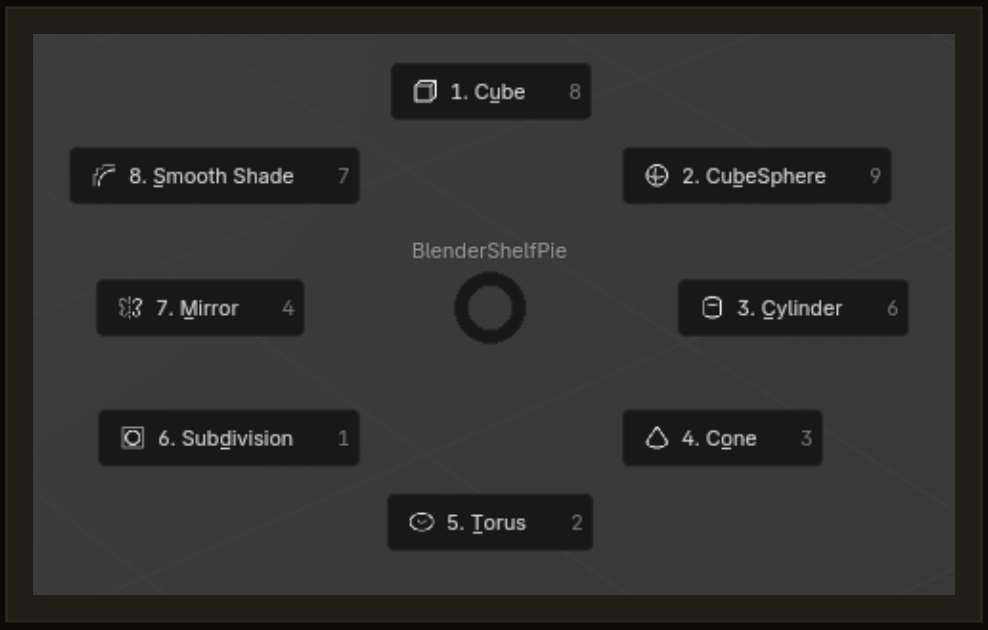


Хоткей Show/Hide Shelf рядом с хоткеем пай-меню

Назначьте клавишу для Show/Hide Shelf в блоке под режимом отображения, рядом с хоткеем пай-меню. Одно нажатие прячет шелф во всех редакторах, следующее возвращает. Пока шелф скрыт, пай-меню продолжает работать.

По умолчанию клавиша не назначена. После перезапуска Blender шелф всегда виден, и Reset Position тоже возвращает его.

Пай-меню во вьюпорте



Пай-меню EasyShelf (Object Mode)

Назначенный хоткей открывает радиальное меню кнопок того шелфа, в чьём контексте оно вызвано.

Preferences: общие настройки

Блок «Backup»: Export/Import Settings сохраняют и загружают все настройки аддона одним JSON-файлом — удобно перед переустановкой или при переходе на другой компьютер.

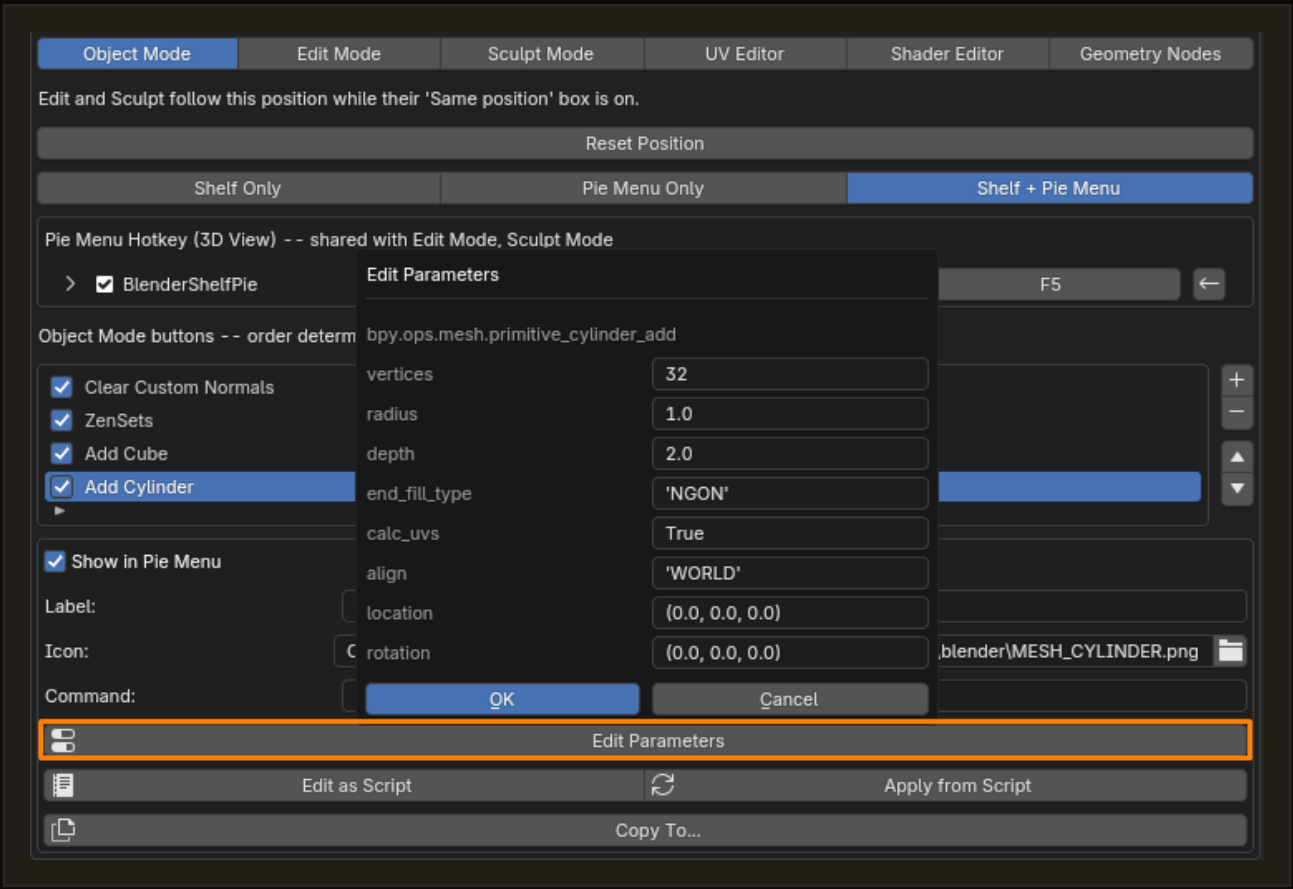
Сворачиваемый блок «Appearance» действует на все шелфы: размер кнопок и прозрачность иконок, надписи (показ, размер шрифта, цвет, положение относительно кнопки), цвета кнопок и фона, Show Number — показ порядковых номеров.

Надписи, расположенные снаружи кнопок (Above, Below, Left, Right), сами переходят на противоположную сторону, если у края вьюпорта они бы не поместились, например когда шелф прижат к панели инструментов. Выбранная настройка не меняется, а надписи возвращаются, как только шелф отодвигается от края.

Список кнопок

Под настройками контекста — список его кнопок. У каждой кнопки своя карточка: подпись, иконка, команда. Стрелками меняется порядок, «+» и «–» добавляют и удаляют кнопки, галочка Show in Pie исключает кнопку из пай-меню.

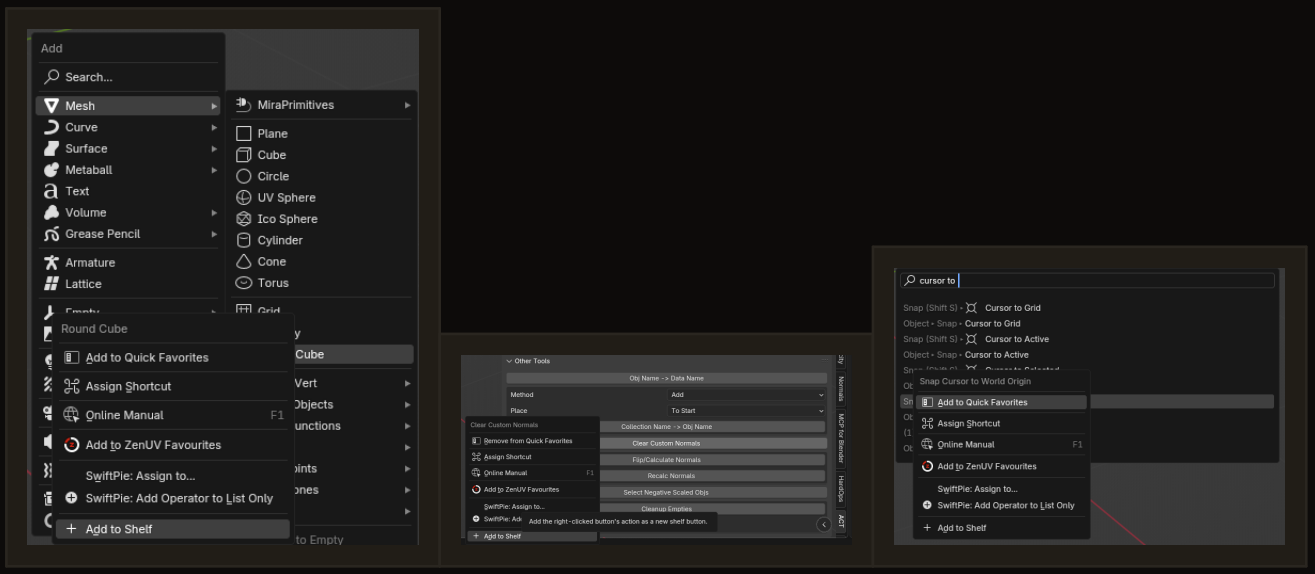
Настройка параметров без Python



Edit Parameters: правка параметров цилиндра без Python

«Edit Parameters» открывает окно, позволяющее редактировать доступные для функции параметры — например, количество граней, радиус и высоту у цилиндра — без необходимости писать или редактировать Python-скрипт вручную.

Способы добавить функцию (кнопку) на шелф



«Add to Shelf» работает из любого контекстного меню кнопки: встроенный оператор, кнопка стороннего аддона, результат поиска (F3)

1. «+» в списке настроек — пустая заготовка, команду и подпись впишите вручную.

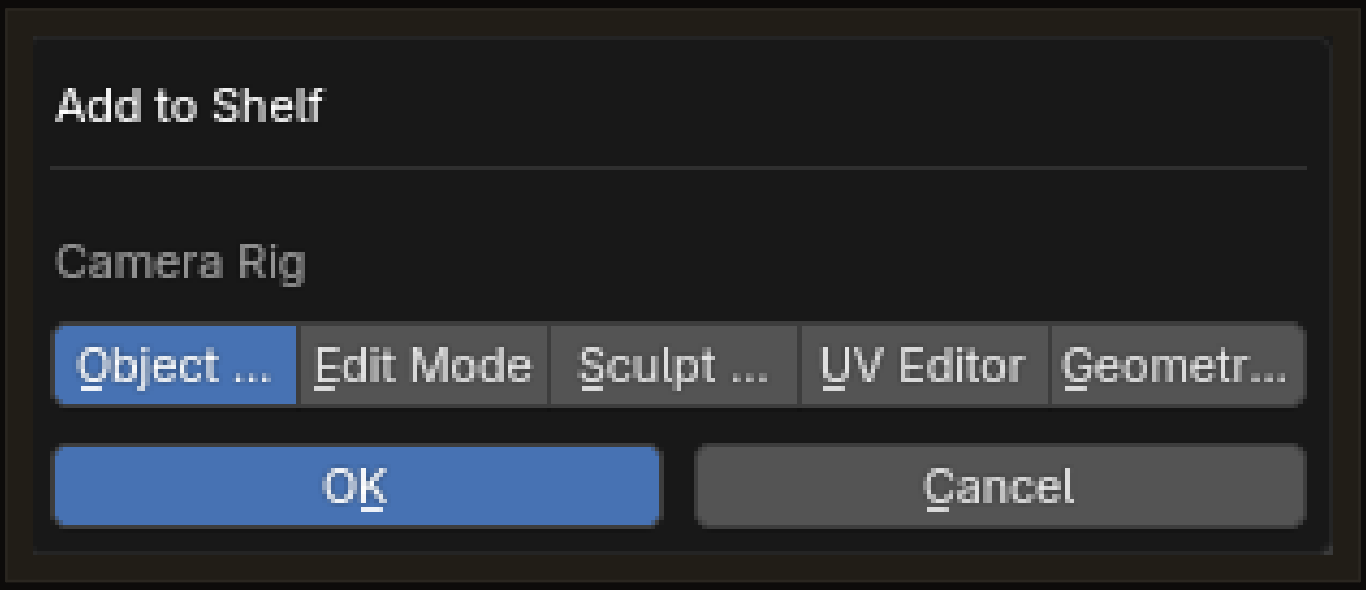
2. ПКМ на любой кнопке интерфейса, включая сторонние аддоны → «Add to Shelf».

3. Найдите функцию через поиск операторов (F3), затем так же — ПКМ → «Add to Shelf».

4. ПКМ на кнопке шелфа → «Add Script»: пустая кнопка и Text Editor, куда вставляется скрипт.

5. ПКМ на кнопке шелфа → «Import Script (.py)»: кнопка из готового Python-файла (подробнее в разделе «Скрипт на кнопку»).

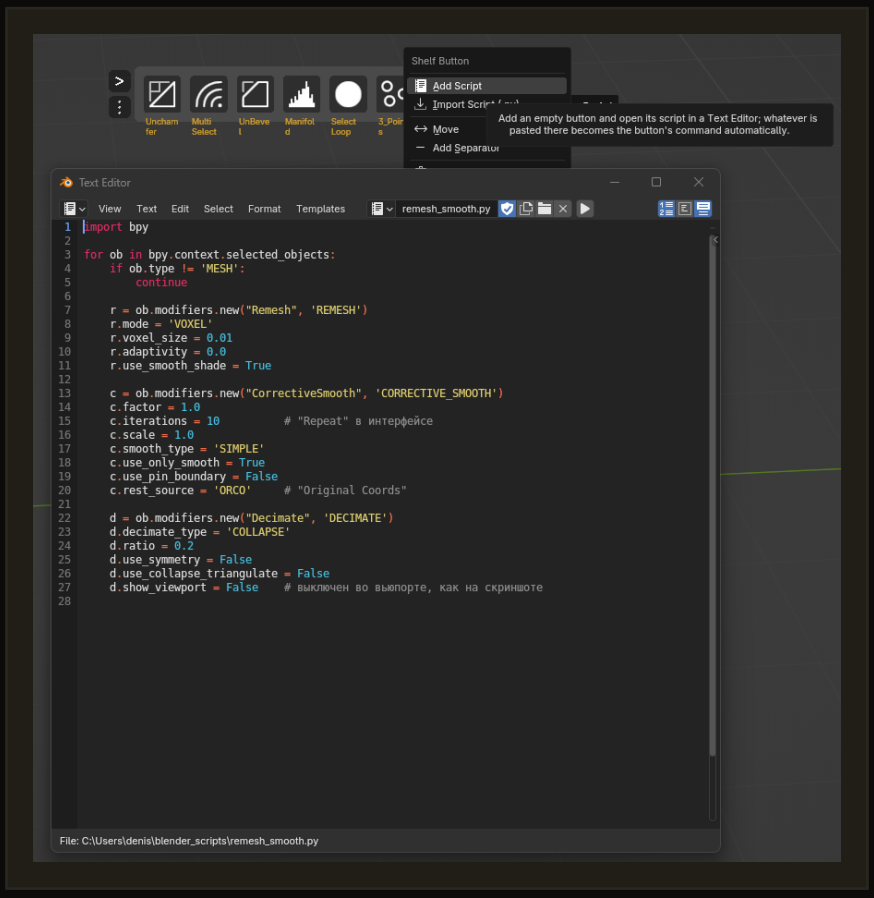
Выбор шелфа при добавлении



Диалог «Add to Shelf»: выбор шелфа, в который попадёт кнопка

Если включено несколько шелфов, при «Add to Shelf» открывается диалог выбора. Он сразу предлагает шелф, подходящий текущему редактору и режиму.

Скрипт на кнопку



«Add Script»: пункт меню и Text Editor, куда вставляется код

Кнопка может запускать не одну команду, а целый Python-скрипт. Правый клик по кнопке шелфа открывает меню с двумя новыми пунктами.

Скрипт можно написать самому или попросить ИИ: опишите, что должна делать кнопка, и вставьте готовый код.

«Add Script» создаёт пустую кнопку со стандартной иконкой и открывает её скрипт в Text Editor. Вставьте туда код — через долю секунды он сам запишется в команду кнопки, ничего нажимать не нужно.

«Import Script (.py)» создаёт кнопку из готового Python-файла, название берётся из имени файла.

Иконка подбирается автоматически: по тегу «# icon: ИМЯ» в первых строках скрипта, по первой команде bpy.ops или по названию. Если подходящей нет, остаётся стандартная — её можно заменить вручную.

Скрипт копируется в кнопку. Если потом изменить файл на диске, кнопка не обновится — импортируйте его заново.

Иконки

Для популярных инструментов иконка подбирается автоматически: в комплекте есть собственные иконки UV-операций (Unwrap, Pack, Stitch, Gridify, Straighten, Export Layout и другие) и основных кистей скульпта (Draw, Clay Strips, Grab, Smooth, Inflate и другие). Если подходящей иконки нет, выберите свою кнопкой с папкой рядом с полем Icon.

Копирование функций между шелфами

Кнопка «Copy To...» в карточке функции копирует её в другой включённый шелф — например, из Object в Edit. Правка команды в одном списке не затрагивает копию в другом.

Что писать в Command

Вы можете добавить собственный скрипт или макрос (последовательность команд), написав его самостоятельно или попросив ИИ (Claude, ChatGPT и т. п.). Для длинных скриптов есть «Edit as Script» и «Apply Script»: код правится в текстовом редакторе Blender. Для быстрой правки параметров одиночного вызова оператора удобнее «Edit Parameters» — Python редактировать вручную не придётся.