BLENDER+PYTHON TERMINOLOGY GUIDE

V 1.0

bpy

Blender's official Python library that allows users to interact with the current **Blender session**

bpy.data

Submodule to directly access data saved in the .blend file (such as cameras, meshes, and brushes)

bpy.context

Submodule to access the current "input" of the user, such as selection, currently active scene or camera

bpy.ops

Submodule to access all of Blender's operators

bpy.types

Submodule to access all the Python classes Blender registers and stores (including classes defining operators and panels)

bpy.props

Submodule defining common datatypes (strings of text, numbers, enumerators) utilized by Blender, often used to define settings for operators or data in the .blend file

Decorator

A function that takes another function as an argument and returns a new function with enhanced functionality.

@staticmethod

A built-in decorator used to define a method that doesn't operate on an instance of the class (i.e., it doesn't use self). Static methods are called on the class itself, not on an instance of the class.

@classmethod

A built-in decorator used to define a method that operates on the class itself (i.e., it uses cls). Class methods can access and modify class state that applies across all instances of the class

@property

A built-in decorator used to define a method as a property, which allows you to access it like an attribute. This is useful for encapsulating the implementation of a method while still providing a simple interface.

Operator

A class registered in Blender that operates on the current Blender session, such as deleting an object or saving user preferences

Invoke

Run the function, assuming the existence of a user interface and

interactivity.

Execute

Run the function, not assuming the existence of a user interface (ie Blender may be running from the terminal) or user interactivity

Extension

A resource that users can subscribe to. A universal standard that currently can be adopted add-ons and themes.

Repository

A collection of extensions, stored either on a webpage or a hard drive, that are accessible to search, download, and manage from within

Blender.

Manifest

A text file containing metadata describing the extension for Blender and repositories to understand

Info Area

A UI area defined in Blender that prints most user actions

as Python code

System Console

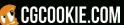
Console

Interactive

A console managed by Blender for logging info, warnings, or errors while using the app

A UI area defined in Blender that allows users to run Python code line-by-line on the current Blender session

UTILITY Scripting your first Blender add-on



BLENDER+PYTHON TERMINOLOGY GUIDE

V 1.0

Blender's official Python bpy

library that allows users to interact with the current

Blender session

bpy.data Submodule to directly access data

saved in the .blend file (such as cameras, meshes, and brushes)

bpy.context Submodule to access the current

"input" of the user, such as selection. currently active scene or camera

Submodule to access all of Blender's bpy.ops

operators

Submodule to access all the Python bpy.types

classes Blender registers and stores (including classes defining operators

and panels)

bpy.props Submodule defining common

datatypes (strings of text, numbers, enumerators) utilized by Blender, often used to define settings for operators or data in the .blend file

Decorator

A function that takes another function as an argument and returns a new function with enhanced functionality.

@staticmethod

A built-in decorator used to define a method that doesn't operate on an instance of the class (i.e., it doesn't use self). Static methods are called on the class itself, not on an instance of the class.

@classmethod

A built-in decorator used to define a method that operates on the class itself (i.e., it uses cls). Class methods can access and modify class state that applies across all instances of the class.

@property

A built-in decorator used to define a method as a property, which allows you to access it like an attribute. This is useful for encapsulating the implementation of a method while still providing a simple interface.

Operator

A class registered in Blender that operates on the current Blender session, such as deleting an object or saving user preferences

Invoke

Run the function, assuming the existence of a user interface and

interactivity.

Execute

Run the function, not assuming the existence of a user interface (ie Blender may be running from the terminal) or user interactivity

Extension

A resource that users can subscribe to. A universal standard that currently can be adopted add-ons and themes.

Repository

A collection of extensions, stored either on a webpage or a hard drive, that are accessible to search. download, and manage from within Blender.

Manifest

A text file containing metadata describing the extension for Blender and repositories to understand

Info Area

A UI area defined in Blender that prints most user actions

as Python code

System Console

A console managed by Blender for logging info, warnings, or errors while

using the app

Interactive Console

A UI area defined in Blender that allows users to run Python code line-by-line on the current Blender session



