mpiea_mmr Documentation

Release 0.1.2

Andres FR

Mar 25, 2019

CONTENTS:

CHAPTER

ONE

MPIEA_MMR PACKAGE

1.1 Submodules

1.2 mpiea_mmr.blender_utils module

Utilities for interaction with Blender

class `mpiea_mmr.blender_utils.`**`ArgumentParserForBlender`**(*prog=None*, *usage=None*, *description=None*, *epilog=None*, *parents=[]*, *formatter_class=<class 'argparse.HelpFormatter'>*, *prefix_chars='-'*, *fromfile_prefix_chars=None*, *argument_default=None*, *conflict_handler='error'*, *add_help=True*, *allow_abbrev=True*)

Bases: `argparse.ArgumentParser`

This class is identical to its superclass, except for the parse_args method (see docstring). It resolves the ambiguity generated when calling Blender from the CLI with a python script, and both Blender and the script have arguments. E.g., the following call will make Blender crash because it will try to process the script's -a and -b flags:

```
blender --python my_script.py -a 1 -b 2
```

To bypass this issue this class uses the fact that Blender will ignore all arguments given after a double-dash ('–'). The approach is that all arguments before '–' go to Blender, arguments after go to the script. The following CLI calls work fine:

```
blender --python my_script.py -- -a 1 -b 2
blender --python my_script.py --
```

`get_argv_after_doubledash`(*argv*)

Parameters **`argv`** (`list of str`) – Expected to be sys.argv (or alike).

Returns The argv sublist after the first `'--'` element (if present, otherwise returns an empty list).

Return type list of str

Note: Works with any *ordered* collection of strings (e.g. list, tuple).

`parse_args``()`

This method is expected to behave identically as in the superclass, except that the sys.argv list will be pre-processed using get_argv_after_doubledash before. See the docstring of the class for usage examples and details.

Note: By default, *argparse.ArgumentParser* will call *sys.exit()* when encountering an error. Blender will react to that shutting down, making it look like a crash. Make sure the arguments are correct!

1.3 mpiea_mmr.foo_module module

Module containing a simple class with low memory and runtime requirements.

`class` `mpiea_mmr.foo_module.`**`Foo`**`(`*size=1000000*`)`

Bases: `object`

A simple class with low memory and runtime requirements.

`get_result``()`

Basic getter.

`loop``(`*times*`)`

Restart result and run computation a number of times.

1.4 Module contents

Init file for the add-on.

To install it, make sure Blender's Python is able to find it under addon_utils.paths(), and that the Blender version matches to make it installable.

Alternatively, run this init file as a script from Blender.

`mpiea_mmr.`**`register`**`()`

`mpiea_mmr.`**`unregister`**`()`

CHAPTER

TWO

INDICES AND TABLES

- genindex
- modindex
- search

PYTHON MODULE INDEX

INDEX