

Namespace ArgumentAPI

Classes

[APIControlUnit](#)

The loadbearing class, with which the user is able to communicate when defining options and their parameters, as well as when parsing the values obtained from these options during runtime.

[Enum](#)

Represents the Enum type that can be specified as an argument type. To facilitate working with it, an implicit cast is defined, so that the Enum class acts as a List<>.

[EnumParameter](#)

Encapsulates enum parameters for options.

[EnumParameterBuilder](#)

Provides method overrides for Enum type parameters.

[Float](#)

Represents the Float (float) type that can be specified as an argument type. It is encapsulated in a class because float is a value type, which does not work with the "dynamic" keyword. An implicit cast is defined so that the Float class acts as a float.

[FloatParameter](#)

Encapsulates float parameters for options.

[FloatParameterBuilder](#)

Provides method overrides for float type parameters.

[Int](#)

Represents the Int (int) type that can be specified as an argument type. It is encapsulated in a class because int is a value type, which does not work with the "dynamic" keyword. An implicit cast is defined so that the Int class acts as an int.

[IntParameter](#)

Encapsulates int parameters for options.

[IntParameterBuilder](#)

Provides method overrides for int type parameters.

[Option](#)

Houses the Option definitions.

[OptionBuilder](#)

Builder pattern class; used to provide a fluid option defining experience to the user, while not exposing them to the Option objects.

[Parameter](#)

Parent class for all types of parameters that can be assigned to options.

[ParameterBuilder](#)

Builder pattern class used to provide a fluid parameter defining experience to the user, while not exposing them to the Parameter objects.

[StringParameter](#)

Encapsulates string parameters for options.

[StringParameterBuilder](#)

Provides method overrides for string type parameters.