

Context

[Contexts](#) are an abstract class that manifests as either a [Character](#), [Skill](#), [Ability](#), or [Resource](#). While they come in all shapes and sizes, there are a few universal characteristics.

Properties

[Contexts](#) share the following properties

**optional,*

name*: **string**

The name of the [context](#).

Default: [key] | "

prototype*: **boolean**

A boolean representing whether or not the [context](#) is not instantiable

Default: False

prototypes*: **array <string>**

A list of **strings** representing the names of [contexts](#) of the same type whose properties should be merged with their own. The merger is done recursively within **objects**, with all keys present in the prototype but not in the original being set to the prototype value.. Arrays are not merged.

Default: []

properties*: **object <string, Evaluable>**

A mapping of **strings** to Evaluables. After prototype properties are filled in the [context](#) all values matching the **string** (including those found in arrays, and deep within embedded **objects**) are replaced with the Evaluable. Useful for DRY inheritance.

Character ([Context](#))

TODOcument

Skill ([Context](#))

TODOcument

Ability ([Context](#))

TODOcument

Resource ([Context](#))

TODOcument

Evaluable

[Evaluables](#), as the name may imply, are values that [Contexts](#) can evaluate into either a **string**, **number**, or [context](#). [Evaluables](#) like [Contexts](#), are abstract and manifest as either a **string**, **number**, Function, or **object** <**string**, [Evaluable](#)>. All [Evaluables](#) that are simply **strings** or **numbers** will evaluate into themselves. Any [Evaluables](#) that are Functions will evaluate into the product of their function. And any [Evaluables](#) that are **object** <**string**, [Evaluable](#)> will evaluate into the evaluation of the value while in the [context](#) of the key.

Function ([Evaluable](#))

object <**string**, **object**>

The **string** associated with any instantiable class is the class name without '_Func'

Context_Func (*Function*)

TODOcument

Addition_Func (*Function*)

TODOcument

Subtraction_Func (*Function*)

TODOcument

Multiplication_Func (*Function*)

TODOcument

Division_Func (*Function*)

TODOcument

Greater_Func (*Function*)

TODOcument

Less_Func (*Function*)

TODOcument

Greater_or_Equal_Func (*Function*)

TODOcument

Less_or_Equal_Func (*Function*)

TODOcument

Maximum_Func (*Function*)

TODOcument

Minimum_Func (*Function*)

TODOcument

Map_Func (*Function*)

TODOcument

Contains_Func (*Function*)

TODOcument

Or_Func (*Function*)

TODOcument

And_Func (*Function*)

TODOcument

Not_Func (*Function*)

TODOcument

Get_Func (*Function*)

TODOcument

Eval_Func (*Function*)

TODOcument

Die_Roll_Func (*Function*)

TODOcument