

Plugin Attributes

Use this [wiki plugin](#), introduced in [Tiki6](#), to assign generic attributes to the current object. Attributes are provided as parameters. For tiki.geo.lat,

```
{attributes tiki_geo_lat=...}
```

Removing the plugin or values in it will not remove the attributes. See [Object Attributes and Relations](#) for the available attributes.

Usage

The plugin Attributes is not enabled by default, you can enable it from the Control Panel, Editing and Plugins, Plugins.

```
{attributes tiki_geo_lat="" tiki_geo_lon="" tiki_geo_google_zoom="" tiki_relation_target=""
tiki_relation_source=""}
```

Parameters

Assign generic attributes to the current object

Introduced in Tiki 6.


[Go to the source code](#)

Preferences required: wikiplugin_attributes

Parameters

no parameters

Examples of using it to set fixed values and use them in a tracker

This method was used in a Tiki12 shop and I retrieved the code 

In this use case we want to have fixed/editable values that can be used in different features of Tiki. There is several advantages;

- Changing one value will impact many other objects (items, calculation, etc).
- Editing can be very limited (permissions to edit this page only)
- Editing can be super relaxed (permissions to edit this page only)
- It may help to keep fixed values through time (VAT may change from time to time however you need to keep the value for a specific date)

Here some example of plugin attributes in a wiki page and the usage in another place (mainly tracker field)

Display a value using attribute in many wiki pages (plugin List)

On the wiki page name PluginAttributes

```
{attributes nice_guy_name="Bernard"}
```

On any other wiki page

```
{LIST()} {OUTPUT(report="y")} {CALC()}(attribute (object (str wiki page) (str PluginAttributes)) (property nice.guy.name)){CALC} is the first name of a nice guy name. {OUTPUT} {LIST}
```

Will produce:

Bernard is the first name of a nice guy name.

Note: This should work in many other places like Blogs, Articles, Forums, Modules, etc. where there is a wiki area and it accept the wiki syntax.

Using a constant from the pluginAttributes in a tracker field calculation

On the wiki page

```
{attributes shop_general_handling_fees="0.2"}
```

Calculation in a mathematical field type in a tracker

```
(mul Price (attribute (object (str wiki page) (str Constants)) (property shop.general.handling.fees)))
```

Using a constant from the pluginAttributes to return different value in a php file

On the wiki page

```
{attributes shop_XXA_minimum="499" shop_XXB_minimum="599"}
```

usage and condition in a php file

```
// check min amount $shop_constants = TikiLib::lib('attribute')->get_attributes('wiki page', 'Constants'); if ($input->supplierOrderName->int() == TikiLib::symbols()->Supplier['XXA']) { $minimum = $shop_constants[ÃfÆ'Ã,ÃçÃfÃçÃçâ, -ÃiÃ,Ã-ÃfÃçÃçâ, -Ã¾Ã,Ãçshop.XXA.minimum']; } else if ($input->supplierOrderName->int() == TikiLib::symbols()->Supplier['XXB']) { $minimum = $shop_constants[ÃfÆ'Ã,ÃçÃfÃçÃçâ, -ÃiÃ,Ã-ÃfÃçÃçâ, -Ã¾Ã,Ãçshop.XXB.minimum']; } else { throw new Denied(tr('Unsupported Shop')); }
```

Using a constant from the pluginAttributes in a Smarty template

On the wiki page

```
{attributes shop_freeshipping_limit="699"}
```

```
<p class="flat_cost">{tr}Shipping{/tr} {if ($cart_total > $shop_constants.shop_freeshipping_limit)}<span class="free_shipping">ÃfÆ'Ã+â€™ Ãfâ€šÃ,ÃçÃfÆ'Ã,ÃçÃfÃçÃçâ, -ÃiÃ,Ã-Ãfâ€|Ã,ÃiÃfÆ'Ãçâ, -ÃiÃfâ€š Ã,Ãº0.00</span>{else}ÃfÆ'Ã+â€™ Ãfâ€šÃ,ÃçÃfÆ'Ã,ÃçÃfÃçÃçâ, -ÃiÃ,Ã-Ãfâ€|Ã,ÃiÃfÆ'Ãçâ, -ÃiÃfâ€š Ã,Ãª{flat_cost|escape}{/if} {* $prefs.payment_currency|escape *}</p>
```

Related pages

- [Object Attributes and Relations](#)
- [Mathematical Calculation Tracker Field](#)
- [PluginRelations](#)
- [PluginAddRelation](#)
- For all plugins* [Wiki Plugin](#) - has links to all plugins and describes how the syntax works for a plugin
- [Mods Type Wikiplugins](#)

Aliases

- [PluginAttribute](#)