Programming Conventions

This manual contains the **Naming Conventions** when creating your solutions, projects, datastore classes, collections, datastore class methods, datasources, and attributes.

The **Reserved Keywords** section informs you of all the words that are reserved and not used in your variable names, functions, datastore classes, and attributes.

Naming Conventions

Since Wakanda solutions, projects, datastore classes, collections, datastore class methods, datasources, and attributes are handled through JavaScript, they must comply with JavaScript's naming rules: names and IDs must begin with a letter (A-Z, a-z) and can contain letters (A-Z, a-z), digits (0-9), and underscores. Accented and non-Roman characters are accepted; however, the following characters are not: !@#$%^&*()+=-[]';,./{}|\":<>?~.

Widgets must follow the standard HTML specifications: widget IDs must begin with a letter (A-Z, a-z) and can contain letters (A-Z, a-z), digits (0-9), and underscores. Characters, such as !@#$%^&*()+=-[]';,./{}|\":<>?~, as well as accented and non-Roman characters are not allowed for the ID of a widget.

Spaces in any of the names or IDs are also not allowed. If you include one, the Datastore Model Designer and GUI Designer will automatically replace it with either an underscore or a hyphen.

Following JavaScript's conventions, variables and functions begin with a lowercase letter while constructors and class names begin with an uppercase letter and the rest in camel case (i.e., all the spaces in the words are removed and each word is capitalized). In Wakanda, we have defined a few of our own naming conventions along these same lines:

- **Datastore Class**: The first letter must be capitalized to follow the logic of the datasource whose first letter must be lowercase. For the "LineItem" datastore class, the datasource is named "lineItem," by default. Generally, the class name is singular.
- **Collection**: Collection names follow the same rule as datastore class names. If you have a datastore class named "LineItem," Wakanda generates "LineItemCollection" as its collection name. You can also rename it to "LineItems" (the plural form of the datastore class name), if you prefer.
- **Attribute**: It is suggested that the first letter of an attribute be lowercase so that you can easily distinguish datastore classes/collections from attributes. For example, "firstName." Although allowed, avoid naming attributes the same as datastore classes
- **Datastore Class Method**: It is suggested that the name of a datastore class method be written in camel case and can begin with either a lowercase or an uppercase letter. For example, "checkEmployee" or "CheckEmployee."
- **Widget**: It is suggested that the first letter of the widget ID be lowercase and that all other words are in camel case. For example, "myGrid."
- **Datasource**: The first letter of a datasource name must be lowercase to follow the logic of the datastore class whose first letter must be in uppercase. For the "LineItem" datastore class, the datasource is named "lineItem," by default.

Reserved Keywords

The following keywords, as defined by ECMAScript 3, 5, and 5.1, are considered “reserved” and should not be used as variable names, functions, datastore classes, and attributes:

abstract
boolean
byte
char
break
class
case
catch
const
continue
debugger
default
delete
do
double
else
enum
export
extends
final
finally
float
for
function
goto
if
in
instanceof
int
interface
implements
import
long
native
new
package
private
protected
public
return
short
static
super
synchronized
switch
this
throw
throws
transient
try
typeof
var
void
volatile
while
with

Wakanda Reserved Keywords

Here are other reserved keywords in Wakanda that we suggest you not use as variable names, functions, datastore classes, and attributes:

administrator
application
console
dataService
directory
ds
httpServer
name
os

process
rpcService
sessionStorage
settings
solution
storage
webAppService
wildchar