

Function Reference

Method: `clearValidation(question)`
Parameters: question object - object of the question
Return Value: Nil
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    wscScripting.clearValidation(question);
}
```

Method: `concatenateQuestion(questionContainer, questionText)`
Parameters: questionContainer - javascript script container
questionText - text for the template concatenation
Return Value: Nil
Example:

```
wscScripting.concatenateQuestion('{QuestionIdentity}', '{What is your
occupation {OCCUPATION} and when you shop at {STORE} how many units of
milk a week do you buy {HOWMANY} and how satisfied are you
{SATISFACTION}}');
```

Method: `derankChoice(question, choice)`
Parameters: question object - object of the question
choice object - object of a choice
Return Value: Nil
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var choice = wscScripting.getChoiceByValue(question, 1);
    if (choice) {
        wscScripting.derankChoice(question, choice);
    }
}
```

Method: `deselectChoice(question, choice)` second range
Parameters: question object - object of the question
choice object - object of a choice
Return Value: boolean - confirmation that the choice was deselected
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var choice = wscScripting.getChoiceByValue(question, 1);
    if (choice) {
        var isSelected = wscScripting.deselectChoice(question, choice);
    }
}
```

Method: **deselectChoiceByValue(question, number)** second range

Parameters: question object - object of the question
number - value of the question choice to check

Return Value: boolean - confirmation that the choice was deselected

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var isSelected = wscScripting.deselectChoiceByValue(question, 1);
}
```

Method: **deselectMatrixChoice(question, choice, row)**

Parameters: question object - object of the question
choice object - object of a choice
row object - object of a row

Return Value: boolean - confirmation that the choice was deselected

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var row = wscScripting.getRowByTagValue(question, 'BANKCODE',
    'AMER');
    if (row) {
        var choice = wscScripting.getChoiceByValue(question, 1);
        if (choice) {
            var isSelected = wscScripting.deselectMatrixChoice(question,
            choice, row);
        }
    }
}
```

Method: **disableQuestion(question)**

Parameters: question object - object of the question

Return Value: boolean - confirmation that the choice was disabled

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var isSelected = wscScripting.disableQuestion(question);
}
```

Method: **enableQuestion(question)**

Parameters: question object - object of the question

Return Value: boolean - confirmation that the choice was enabled

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var isSelected = wscScripting.enableQuestion(question);
}
```

Method: `findObjectByKey(array, key, value)`
Parameters: array of objects
string - key within objects in the array
value - value of key within object to search
Return Value: object - object within the array
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var oChoice = wscScripting.findObjectByKey(question.choices, 'text',
'Apple');
}
```

Method: `findOrdinalByKey(array, key, value)`
Parameters: array of objects
string - key within objects in the array
value - value of key within object to search
Return Value: int - ordinal within the array
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var nOrd = wscScripting.findOrdinalByKey(question.choices, 'text',
'Apple');
}
```

Method: `getABTesting()`
Parameters: Nil
Return Value: integer - Value (range 1..100) of for use by AB Testing
Example:

```
var ABTest = wscScripting.getABTesting();
if (ABTest <= 50) {
    // Split 50:50
}
```

Method: `getAgeFromDate()`
Parameters: object - get the value from a date question, pass in the returned object
Return Value: integer or NaN
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var getValue = wscScripting.getValue(question);
    var age = wscScripting.getAgeFromDate(getValue);
}
```

Method: **getBrowserData()**
Parameters: Nil
Return Value: object - Browser Object

```
o.browser = Browser Name  
o.version = Browser Version  
o.OS = Operating System
```

Example:

```
var browser = wscScripting.getBrowserData();  
if (browser.OS == 'Windows') {  
    // The respondent is on a Windows computer  
}
```

Method: **getChoiceByCode(question, value)** second range
Parameters: question object - object of the question
string - value of the tag

Return Value: object or undefined

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {  
    var choice = wscScripting.getChoiceByCode(question, 'TOYOTA');  
    if (choice) {  
        // I can do something with this choice  
    }  
}
```

Method: **getChoiceByIdentity(question, identity)** second range
Parameters: question object - object of the question
string - identity of the choice to search for

Return Value: object or undefined

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {  
    var choice = wscScripting.getChoiceByIdentity(question, sId);  
    if (choice) {  
        // I can do something with this choice  
    }  
}
```

Method: **getChoiceByTagValue(question, tagName, value)** second range
Parameters: question object - object of the question
string - name of the tag to search for
string - value of the tag being searched

Return Value: object or undefined

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {
```

```

    var choice = wscScripting.getChoiceByTagValue(question, 'position',
'Manager');
    if (choice) {
        // I can do something with this choice
    }
}

```

Method: **getChoiceByValue(question, value) second range**

Parameters: question object - object of the question
number - value of the question choice to retrieve

Return Value: object or undefined

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var choice = wscScripting.getChoiceByValue(question, 1);
    if (choice) {
        // I can do something with this choice
    }
}

```

Method: **getChoiceCommentByChoice(question, choice)**

Parameters: question object - object of the question
choice object - object of the choice

Return Value: string or undefined

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var choice = wscScripting.getChoiceByValue(question, 1);
    if (choice) {
        // I can do something with this choice
    }
}

```

Method: **getChoiceRandomizationItems(question, matrixnumber)**

Parameters: question object - object of the question

Return Value: Array = Array of randomization data

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var aItems = wscScripting.getChoiceRandomizationItems(question, 0);
}

```

Method: **getCurrentDateAsString()**

Parameters: Nil

Return Value: dateString = Current date as a string

Example:

```

var stringDate = wscScripting.getCurrentDateAsString();

```

Method: `getDateFromDateString(dateString)`
Parameters: string = Value of string type to be converted to a date in format usable by WSC
Return Value: date = Newly created date type
Example:

```
var stringDate = '2012.01.31.16.24';  
var newDate = wscScripting.getDateFromDateString(stringDate);
```

Method: `getDateFromStringFromDate(date)`
Parameters: Date = Value of Date type to be converted to a string in format usable by WSC
Return Value: string = Newly created string in format of YYYY.MM.DD.HH.mm
Example:

```
var newDate = new Date();  
var newString = wscScripting.getDateFromStringFromDate(newDate);  
  
// newString contains today's date  
// e.g. 2012.01.31.16.24
```

Method: `getDirection()`
Parameters: Nil
Return Value: string containing:-
ltr = Left to Right
rtl = Right to Left e.g. Arabic
Example:

```
var direction = wscScripting.getLanguageId();  
if (direction == 'rtl') {  
    // This is a survey using a RTL language  
}
```

Method: `getDisplayType()`
Parameters: Nil
Return Value: string containing:-
standard = Standard Display
tablet = Tablet Computer e.g. iPad
mobile = Mobile Phone / Cellular Phone e.g. iPhone
Example:

```
var display = wscScripting.getDisplayType();  
if (display == 'tablet') {  
    // This is a tablet based display  
}
```

Method: `getDistribution()`
Parameters: Nil
Return Value: object or undefined
Example:

```
var object = wscScripting.getDistribution();  
if (object) {
```

```

        // I can do something with this object
    }

```

Method: **getElementById(id)**

Parameters: string - Identity of an Html Element

Return Value: object or undefined

Example:

```

var object = wscScripting.getElementById('mycontrol');
if (object) {
    // I can do something with this element
}

```

Method: **getEventData(name)**

Parameters: string - Identity of an item of data temporarily stored for later use on the current page only

Return Value: value or undefined

Example:

```

var object = wscScripting.getEventData('myvalue');
if (object) {
    // I can do something with this value
    // Value contains the text 'Hello World!'
}

```

Method: **getFileBarcodeData(question)**

Parameters: question object - object of the question

Return Value: string = JSON package

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var barcode = wscScripting.getFileBarcodeData(question);
    JSON.parse(barcode)
}

```

Method: **getFirstSelectedChoice(question)**

second range

Parameters: question object - object of the question

Return Value: choice object or undefined

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var oChoice = wscScripting.getFirstSelectedChoice(question);
}

```

Method: **getFirstSelectedChoiceValue(question)**

second range

Parameters: question object - object of the question

Return Value: integer value or undefined

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');

```

```

if (question) {
    var nValue = wscScripting.getFirstSelectedChoiceValue(question);
}

```

Method: **getFirstSelectedMatrixChoice(question)** second range

Parameters: question object - object of the question

Return Value: choice object or undefined

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var oChoice = wscScripting.getFirstSelectedMatrixChoice(question);
}

```

Method: **getFirstSelectedMatrixChoiceValue(question)** second range

Parameters: question object - object of the question

Return Value: integer value or undefined

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var nValue =
wscScripting.getFirstSelectedMatrixChoiceValue(question);
}

```

Method: **getLanguageId()**

Parameters: Nil

Return Value: string - Two Character Language Code

Example:

```

var language = wscScripting.getLanguageId();
if (language == 'fr') {
    // This is a survey using French Language
}

```

Method: **getMatrixChoiceCommentByRow(question, choice, row)**

Parameters: question object - object of the question

choice object - object of a choice

row object - object of a row

Return Value: string - comments or undefined

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var row = wscScripting.getRowByTagValue(question, 'BANKCODE',
'AMER');
    if (row) {
        var choice = wscScripting.getChoiceByValue(question, 1);
        if (choice) {
            var comments =
wscScripting.getMatrixChoiceCommentByRow(question, choice, row);
        }
    }
}

```



```
}
```

Method: **getMatrixChoiceNumericTotal(question, choice)**

Parameters: question object - object of the question

choice object - object of a choice

row object - object of a row

Return Value: int - total or undefined

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var row = wscScripting.getRowByTagValue(question, 'BANKCODE',
    'AMER');
    if (row) {
        var choice = wscScripting.getChoiceByValue(question, 1);
        if (choice) {
            var number = wscScripting.getMatrixChoiceNumericTotal(question,
            choice);
        }
    }
}
```

Method: **getMatrixNumericTotal(question)**

Parameters: question object - object of the question

Return Value: int - total or undefined

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var total = wscScripting.getMatrixNumericTotal(question);
}
```

Method: **getMatrixRowNumericTotal(question, row)**

Parameters: question object - object of the question

row object - object of a row

Return Value: int - total or undefined

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var row = wscScripting.getRowByTagValue(question, 'ROW1', 'ALPHA');
    var total = wscScripting.getMatrixRowNumericTotal(question, row);
}
```

Method: **getMatrixValue(question, choice, row)**

Parameters: question object - object of the question

choice object - object of a choice

row object - object of a row

Return Value: int - total or undefined

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var row = wscScripting.getRowByTagValue(question, 'BANKCODE',
'AMER');
    if (row) {
        var choice = wscScripting.getChoiceByValue(question, 1);
        if (choice) {
            var sValue = wscScripting.getMatrixValue (question, choice,
row);
        }
    }
}

```

Method: **getMetaValue(question)**

Parameters: question object - object of the question

Return Value: string - value of the meta

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var sValue = wscScripting.getMetaValue (question);
}

```

Method: **getOriginalDataStatus()**

Parameters: Nil

Return Value: String - Unstarted, Draft, Completed

Example:

```

var status = wscScripting.getOriginalDataStatus();
if (status == 'Unstarted') {
    // I can do something because of this status
}

```

Method: **getPageRandomizationItems()**

Parameters: Nil

Return Value: array or undefined - Array of PageRandomisation Objects

Example:

```

var aItems = wscScripting.getPageRandomizationItems();

```

Method: **getQuestionByDataPipingCode('CODE')**

Parameters: string - Data Piping Code of a Question - Must be a WSC Data Piping Code.

If the question is not on the current page then you should use a Data Piping

Shortcut to include the question on the current page

Return Value: object or undefined

Example:

```

// Using a data piping code
var object = wscScripting.getQuestionByDataPipingCode('CODE');
if (object) {

```

```

        // I can do something with this question
    }

    // Using a data piping symbol if the Question is not on the same page
    // The data piping symbol with the code #data# is required to
    // tell the system to have the question available
    var object2 =
wscScripting.getQuestionByDataPipingCode('[@CODE#data#@]');
    if (object2) {
        // I can do something with this question
    }

```

Method: **getQuestionByIdentity(identity)**
Parameters: string - Identity of a Question - Must be a WSC internal identity
Return Value: object or undefined
Example:

```
var object = wscScripting.getQuestionByIdentity('61ce3764-1288-e111-8eae-0019b9c4ecf3');
if (object) {
    // I can do something with this question
}
```

Method: **getQuotaByCode(code)**
Parameters: string - Code of the Quota
Return Value: quota object - object of the quota
Example:

```
var oQuota = wscScripting.getQuotaByCode('GENDER');
```

Method: **getQuotaByIdentity(identity)**
Parameters: string - Identity of a Quota - Must be a WSC internal identity
Return Value: quota object - object of the quota
Example:

```
var object = wscScripting.getQuotaByIdentity('61ce3764-1288-e111-8eae-0019b9c4ecf3');
if (object) {
    // I can do something with this quota
}
```

Method: **getRecallCode()**
Parameters: Nil
Return Value: string - Unique Code which identifies the response
Example:

```
var recallCode = wscScripting.getRecallCode();
```

Method: **getRowByCode(question, value)**
Parameters: string - each question row must use the tag format: CODE:Somevalue for this method to work

Return Value: object - object of the row
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {  
  var row = wscScripting.getRowByCode(question, "ALPHA");  
}
```

Method: **getRowByTagValue(question, tagName, value)**
Parameters: question object - object of the question
string - name of the tag to search for
string - value of the tag being searched
Return Value: object or undefined
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {  
  var row = wscScripting.getRowByTagValue(question, 'BANKCODE',  
    'AMER');  
  if (row) {  
    // I can do something with this row  
  }  
}
```

Method: **getRowByValue(question, value)**
Parameters: int - rows must have the tag format of Value:1
Return Value: object - object of the row
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {  
  var row = wscScripting.getRowByValue(question, 1);  
}
```

Method: **getRowRandomizationItems(question)**
Parameters: question object - object of the question
Return Value: array or undefined - Array of RowRandomisation Objects
Example: Array = Array of randomization data

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {  
  var aItems = wscScripting.getRowRandomizationItems(question);  
}
```

Method: **getSelectedChoices(question)** second range
Parameters: question object - object of the question
Return Value: array or undefined
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {  
  var selectedChoices = wscScripting.getSelectedChoices(question);  
}
```

```

        if (selectedChoices) {
            // I can do something with this array
        }
    }
}

```

Method: `getSelectedMatrixChoices(question, row)` **second range**

Parameters: question object - object of the question
row object - object of a row

Return Value: array or undefined

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var row = wscScripting.getRowByTagValue(question, 'BANKCODE',
    'AMER');
    if (row) {
        var selectedChoices =
wscScripting.getSelectedMatrixChoices(question, row);
        if (selectedChoices) {
            // I can do something with this array
        }
    }
}

```

Method: `getSelectedRanks(question)`

Parameters: question object - object of the question

Return Value: array or undefined

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var ranks = wscScripting. getSelectedRanks(question);
}

```

Method: `getSubstringLeft(string, number)`

Parameters: string = Base string from which a new string will be extracted
number = Number of Characters from the Left side of the string to be extracted

Return Value: string = Newly extracted string

Example:

```

var newString = wscScripting.getSubstringLeft('Hello World!', 5);
// newString contains 'Hello'

```

Method: `getSubstringRight(string, number)`

Parameters: string = Base string from which a new string will be extracted
number = Number of Characters from the Right side of the string to be extracted

Return Value: string = Newly extracted string

Example:

```

var newString = wscScripting.getSubstringRight('Hello World!', 5);

```

```
// newString contains 'orld!'
```

Method: **getTagValueByName(tagList, tagName)**
Parameters: Array = Array of tags
string = string of the name of the tag
Return Value: object or undefined
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {  
    var row = question.rows[0];  
    if (row) {  
        var tag = wscScripting.getTagValue(row, 'BANKCODE');  
    }  
}
```

Method: **getTrimString(string)**
Parameters: string = Base string from which a new string will be created with spaces at either end of the string removed
Return Value: string = Newly created string
Example:

```
var newString = wscScripting.getTrimString(' Hello World! ');  
  
// newString contains 'Hello World!'
```

Method: **getUrlParameter(string)**
Parameters: string = parameter from URL
Return Value: string = URL parameter value
Example:

```
var language = wscScripting.getUrlParameter('lang');  
// return value "en"
```

Method: **getValidators(question)**
Parameters: question object - object of the question
Return Value: array
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {  
    var validators = wscScripting.getValidators(question);  
    if (validators) {  
        // I can do something with this array of validators  
    }  
}
```

Method: **getValue(question)**
Parameters: question object - object of the question
Return Value: value or undefined dependent on the question type

	value only suitable for SingleText, MultipleText, DemographicEmail, DemographicPhone, Number, Slider and DateTime Questions
Example:	<pre> Var question = wscScripting.getQuestionByDataPipingCode('CODE'); if (question) { var value = wscScripting.getValue(question); } </pre>
Method:	hideElement(element)
Parameters:	string = id of an Html control to hide
Return Value:	boolean - confirmation that the control was hidden
Example:	<pre> wscScripting.hideElement('mydiv'); </pre>
Method:	hideMatrixCell(question, choice, row) second range
Parameters:	question object - object of the question row object - object of a row choice object - object of a choice
Return Value:	Nil
Example:	
Method:	hideQuestion(question)
Parameters:	question object - object of the question
Return Value:	nil
Example:	<pre> var question = wscScripting.getQuestionByDataPipingCode('CODE'); wscScripting.hideQuestion(question); </pre>
Method:	htmlDecode(text)
Parameters:	text - text to be decoded
Return Value:	Decoded text string
Example:	<pre> var textDecoded = wscScripting.htmlDecode('&quot;Hello&quot;'); // Would contain "Hello" </pre>
Method:	htmlEncode(text)
Parameters:	text - text to be encoded
Return Value:	Encoded text string
Example:	<pre> var textEncoded = wscScripting.htmlEncode('"Hello"'); // Would contain &quot;Hello&quot;; </pre>
Method:	isAnyChoiceSelected(question, choices) second range
Parameters:	question object - object of the question choices object - array of choice objects to check
Return Value:	boolean - confirmation that the choice is selected

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question && question.choices) {
    var isSelected = wscScripting.isAnyChoiceSelected(question,
question.choices);
}
```

Method: **isAnyChoiceSelectedByValue(question, values)** **second range**

Parameters: question object - object of the question
array - array of number values to check

Return Value: boolean - confirmation that the choice is selected

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question && question.choices) {
    // Make an array of values
    var arrayChoices = new Array(1, 2, 3);
    var isSelected = wscScripting.isAnyChoiceSelectedByValue(question,
arrayChoices);
}
```

Method: **isAnyMatrixChoiceSelected(question, choices, row)** **second range**

Parameters: question object - object of the question
array - array of choice objects to check
row object - object of the row

Return Value: boolean - confirmation that the choice is selected

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question && question.choices) {
    var row = wscScripting.getRowByTagValue(question, 'BANKCODE',
'AMER');
    if (row) {
        // Make an array of just 1 choice
        var arrayChoices = new Array();
        arrayChoices.push(question.choices[0]);
        var isSelected = wscScripting.isAnyMatrixChoiceSelected(question,
arrayChoices, row);
    }
}
```

Method: **isAnyMatrixChoiceSelectedByValue(question, values, row)** **second range**

Parameters: question object - object of the question
array - array of number values to check
row object - object of the row

Return Value: boolean - confirmation that the choice is selected

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question && question.choices) {
    var row = wscScripting.getRowByTagValue(question, BANKCODE, 'AMER');
```



```

    if (row) {
        // Make an array of values
        var arrayChoices = new Array(1, 2, 3);
        var isSelected =
wscScripting.isAnyMatrixChoiceSelectedByValue(question,arrayChoices,
row);
    }
}

```

Method: **isChoiceSelected(question, choice)** second range

Parameters: question object - object of the question

choice object - object of a choice

Return Value: boolean - confirmation that the choice is selected

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var choice = wscScripting.getChoiceByValue(question, 1);
    if (choice) {
        var isSelected = wscScripting.isChoiceSelected(question,
choice);
    }
}

```

Method: **isChoiceSelectedByValue(question, value)** second range

Parameters: question object - object of the question

number - value of the question choice to check

Return Value: boolean - confirmation that the choice is selected

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var isSelected = wscScripting.isChoiceSelectedByValue(question, 1);
}

```

Method: **isMatrixChoiceSelected(question, choice, row)** second range

Parameters: question object - object of the question

choice object - object of a choice

row object - object of the row

Return Value: boolean - confirmation that the choice is selected

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var choice = wscScripting.getChoiceByValue(question, 1);
    if (choice) {
        var row = wscScripting.getRowByTagValue(question, 'BANKCODE',
'AMER');
        if (row) {
            var isSelected = wscScripting.isMatrixChoiceSelected(question,
choice, row);
        }
    }
}

```

```
}  
}
```

Method: `isMatrixChoiceSelectedByValue(question, value, row)` **second range**

Parameters:
question object - object of the question
number - value of the question choice to check
row object - object of the row

Return Value: boolean - confirmation that the choice is selected

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {  
    var row = wscScripting.getRowByTagValue(question, 'BANKCODE',  
    'AMER');  
    if (row) {  
        var isSelected =  
wscScripting.isMatrixChoiceSelectedByValue(question, 1, row);  
    }  
}
```

Method: `isQuotaFull(quota)`

Parameters: quota object - object of the quota

Return Value: boolean - confirmation that the quota is full

Example:

```
var quota = wscScripting.getQuotaByCode('QUOTACODE');  
if (quota) {  
    var isFull = wscScripting.isQuotaFull(quota);  
}
```

Method: `isStringHtmlEncoded(text)`

Parameters: text to be confirmed as encoded

Return Value: boolean - confirmation that the text is encoded

Example:

```
var isEncoded = wscScripting.isStringHtmlEncoded('&quot;Hello&quot;');  
if (isEncoded) {  
    // The text is encoded  
}
```

Method: `numberToStringWithCommas(number)`

Parameters: integer - number to be converted to string with commas

Return Value: string or undefined

Example:

```
var number = 100000000;  
var numberString = wscScripting.numberToStringWithCommas(number);
```

Method: `randomSelectChoice(question)` **second range**

Parameters: question object - object of the question

Return Value: object or undefined

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var choice = wscScripting.randomSelectChoice(question);
}
```

Method: **randomSelectMultipleChoice(question, maxCount)** **second range**

Parameters: question object - object of the question
int - number of choices to select

Return Value: object or undefined

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var choice = wscScripting.randomSelectMultipleChoice(question, 3);
}
```

Method: **randomizeArray(array)**

Parameters: array - array of single dimension values, like strings or numbers

Return Value: array of randomly sorted values

Example:

```
var array = wscScripting.randomizeArray(['One', 'Two', 'Three']);
```

Method: **rankChoice(question, choice, rank)**

Parameters: question object - object of the question
choice object - object of the choice
int - position to rank

Return Value: none

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var choice = wscScripting.getChoiceByValue(question, 1);
    if (choice) {
        wscScripting.rankChoice(question, choice, 1);
    }
}
```

Method: **resetMatrix(question, value)**

Parameters: question object - object of the question
value - value to place in each cell

Return Value: none

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var choice = wscScripting.getChoiceByValue(question, 1);
    if (choice) {
```

```

        wscScripting.resetMatrix(question, 'Hello');
    }
}

```

Method: **resetMatrixRow(question, row, value)**

Parameters: question object - object of the question
 row object - object of the row
 value - value to place in each cell

Return Value: none

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var row = wscScripting.getRowByTagValue(question, 'BANKCODE',
    'AMER');
    if (row && choice) {
        wscScripting.resetMatrixRow(question, row, 'Hello');
    }
}

```

Method: **resetQuestion(question)**

Parameters: question object - object of the question

Return Value: none

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    wscScripting.resetQuestion(question);
}

```

Method: **selectChoice(question, choice)**

second range

Parameters: question object - object of the question
 choice object - object of a choice

Return Value: boolean - confirmation that the choice was selected

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var choice = wscScripting.getChoiceByValue(question, 1);
    if (choice) {
        var isSelected = wscScripting.selectChoice(question, choice);
    }
}

```

Method: **selectChoiceByValue(question, number)**

second range

Parameters: question object - object of the question
 number - value of the question choice to check

Return Value: boolean - confirmation that the choice was selected

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');

```

```

if (question) {
    var isSelected = wscScripting.selectChoiceByValue(question, 1);
}

```

Method: **selectMatrixChoice(question, choice, row)** second range

Parameters: question object - object of the question
choice object - object of a choice
row object - object of a row

Return Value: boolean - confirmation that the choice was selected

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var row = wscScripting.getRowByTagValue(question, 'BANKCODE',
'AMER');
    var choice = wscScripting.getChoiceByValue(question, 1);
    if (row && choice) {
        var isSelected = wscScripting.selectMatrixChoice(question, choice,
row);
    }
}

```

Method: **selectMatrixChoiceByValue(question, value, row)** second range

Parameters: question object - object of the question
value - value of a choice
row object - object of a row

Return Value: boolean - confirmation that the choice was selected

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var row = wscScripting.getRowByTagValue(question, 'BANKCODE',
'AMER');
    if (row) {
        var isSelected = wscScripting.selectMatrixChoiceByValue(question,
1, row);
    }
}

```

Method: **setChoiceCommentByChoice(question, choice, comment)**

Parameters: question object - object of the question
choice object - object of a choice
string or int - string or number comment

Return Value: nil

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    var choice = question.choices[0];
    if (choice) {
        wscScripting.setChoiceCommentByChoice(question, choice, "Hello
World");
    }
}

```

```
}  
}
```

Method: `setChoiceRandomizationItems(question, choices, matrixNumber)`
Parameters: question object - object of the question
choice object array - array of choice objects
int - matrix number
Return Value: boolean
Example: `bSuccess = wscScripting.setChoiceRandomizationItems(question, aRandom,
0);`

Method: `setEventData(name, value)`
Parameters: string - Identity of an item of data temporarily stored for later use on the current page only
object - Value of an item of data temporarily stored for later use on the current page only
Return Value: boolean - confirmation that the value was correctly added
Example: `wscScripting.setEventData('myvalue', 'Hello World!');`

Method: `setMatrixChoiceCommentByRow(question, choice, row, comment)`
Parameters: question object - object of the question
choice object - object of the choice
row object - object of a row
string - value of a comment
Return Value: Nil
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {  
    var row = wscScripting.getRowByTagValue(question, 'BANKCODE',  
'AMER');  
    var choice = wscScripting.getChoiceByValue(question, 1);  
    if (row && choice) {  
        var isSelected =  
wscScripting.setMatrixChoiceCommentByRow(question, choice, row, 'A  
comment');  
    }  
}
```

Method: `setMatrixValue(question, choice, row, value)`
Parameters: question object - object of the question
choice object - object of the choice
row object - object of a row
int or string - value for the matrix
Return Value: Nil
Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');  
if (question) {
```

```

    var row = wscScripting.getRowByTagValue(question, 'BANKCODE',
    'AMER');
    var choice = wscScripting.getChoiceByValue(question, 1);
    if (row && choice) {
        var isSelected = wscScripting.setMatrixValue(question, choice,
row, 'A comment');
    }
}

```

Method: **setPageRandomizationItems(pageItems)**

Parameters: array of page randomization items

Return Value: bool Success

Example:

```

var aItems = wscScripting.getPageRandomizationItems();
// Make page 4 and page 6 swap the order
var oPage1 = aItems[3];
var oPage2 = aItems[5];
aItems[3] = oPage2;
aItems[5] = oPage1;
wscScripting.setPageRandomizationItems(aItems);

```

Method: **setQuestionRandomizationItems(pageId, pageItems)**

Parameters: page id - string id of the page
array of page randomization items

Return Value: Nil

Method: **setRowRandomizationItems(question, rowItems)**

Parameters: question - object of the question
array of row randomization items

Return Value: Nil

Method: **setStarRatingonClick(question, clickEvent)**

Parameters: question - object of the question
click event to be executed

Return Value: Nil

Example:

```

wscScripting.setStarRatingonClick(oQuestion, function() {
StarRating_Stars() });

```

Method: **setValidation(question, text)**

Parameters: question object - object of the question
string - text of the validation message

Return Value: Nil

Example:

```

var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {

```

```
wscScripting.setValidation(question, 'Something doesnt make sense!');
}
```

Method: **setValue(question, value)**

Parameters: question object - object of the question
value - type dependent on question type
value only suitable for SingleText, MultipleText, DemographicEmail, DemographicPhone, Number, Slider and DateTime Questions

Return Value: Nil

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    wscScripting.setValue(question, 'Hello World!');
}
```

Method: **showElement(string)**

Parameters: string = id of an Html control to show

Return Value: boolean - confirmation that the control was shown

Example:

```
wscScripting.showElement('mydiv');
```

Method: **showMatrixCell(question, choice, row)** **second range**

Parameters: question object - object of the question
choice object - object of a choice
row object - object of a row

Return Value: Nil

Method: **showQuestion(question)**

Parameters: question object - object of the question

Return Value: Nil

Example:

```
var question = wscScripting.getQuestionByDataPipingCode('CODE');
if (question) {
    wscScripting.showQuestion(question);
}
```

Method: **sortNumericArray(array, ascending)**

Parameters: array - array of single dimension numbers
bool - true for ascending

Return Value: array of sorted values

Example:

```
var array = wscScripting.randomizeArray([18, 2, 3], true);
// Will return [2, 3, 18]
```


Method: **sortQuotasByLeastFilled(array)**
Parameters: array – array of quota objects
Return Value: array of sorted quota objects
Example:

```
var quota1 = wscScripting.getQuotaByCode('quotacode');  
var quota2 = wscScripting.getQuotaByCode('quotacode');  
var quotas = [];  
quotas.push(quota1, quota2);  
var leastFilled = wscScripting.sortQuotasByLeastFilled(quotas);
```

Method: **sortQuotasByMostFilled(array)**
Parameters: array – array of quota objects
Return Value: array of sorted quota objects
Example:

```
var quota1 = wscScripting.getQuotaByCode('quotacode');  
var quota2 = wscScripting.getQuotaByCode('quotacode');  
var quotas = [];  
quotas.push(quota1, quota2);  
var mostFilled = wscScripting.sortQuotasByMostFilled(quotas);
```

Method: **sortStringArray(array, ascending)**
Parameters: array – array of string objects
bool – true for ascending
Return Value: array of sorted string objects
Example:

```
var aArray = ['BBB', 'DDD', 'CCC', 'AAA']);  
aArray = wscScripting.sortStringArray(aArray);  
// Will return ['AAA', 'BBB', 'CCC', 'DDD']
```

Method: **validateStringIsDecimal(string)**
Parameters: string – string version of decimal number
Return Value: bool – Is this a string version of a decimal number
Example:

```
var bIsDecimal = wscScripting.validateStringIsDecimal('12.34')  
// Will return true
```

Method: **validateStringIsInteger(string)**
Parameters: string – string version of integer number
Return Value: bool – Is this a string version of a integer number
Example:

```
var bIsInteger = wscScripting.validateStringIsInteger('12.34')  
// Will return false
```