# Fun with While function in FileMaker

We can use While function (or FM.Loop function in our plugin) to do loops directly in FileMaker. With Let statement we can combine several calls to our plugin including While() calls into one expression. You can put this into a custom function or in a Set Variable call to calculate it where you need.

**1. Simple loop**

You can write a complete encapsulated While call which defines a list of numbers with values counting up:

```
While([ list = ""; i = 1]; i ≤ 10; [ list = list
& i & ¶; i = i + 1 ] ; list )
```

This returns a list of 1 to 10.

**2. Fill JSON array**

Next we write a Let statement for combining several MBS calls and a while call to fill a JOSN array. Here we use our object reference numbers to avoid passing around the half finished JSON and speed up the performance of the calculation.

```
Let ( [
  // Create array object
  j = MBS( "JSON.CreateArrayRef" );
  // make a loop to add numbers
  r = While ( i = 0 ; i < 10 ; [
     r = MBS( "JSON.AddNumberToArray"; j; i );
     i = i + 1 ] ; i );
  // format the result
  text = MBS( "JSON.Format"; j );
  // free json
  r = MBS( "JSON.Release"; j )
];
// return result as text
 text )
```

This returns the following JSON:

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9]

**3. JSON for record values**

Next we use a while loop to walk over the list of fields with our plugin and fill in all values for a record into a JSON object:

```
Let ( [
  // Create object as reference
  j = MBS( "JSON.CreateObjectRef" );
  // List fields in current layout
  fields = FieldNames ( get(FileName) ; get(LayoutName ));
  // how many?
  c = ValueCount ( fields );
  // make a loop to add field values
  r = While ( i = 1 ; i ≤ c ; [
     name = GetValue ( fields ; i );
     r = MBS( "JSON.AddItemToObject"; j;
name; MBS( "JSON.CreateValue"; GetField ( name ) ));
     i = i + 1 ] ; i );
  // format the result
  text = MBS( "JSON.Format"; j );
  // free json
  r = MBS( "JSON.Release"; j )
];
// return result as text
 text )
```

Result may be:

```
{
	"First":	"John",
	"Last":	"Miller",
	"Group":	"",
	"Company":	"Some Ltd.",
	"ID":	12
}
```

**4. Same with XML:**

```
Let ( [
  // XML template
  xml = MBS( "XML.Parse"; "<record></record>");
  // List fields in current layout
  fields = FieldNames ( get(FileName) ; get(LayoutName ));
  // how many?
  c = ValueCount ( fields );
  // make a loop to add field values
  r = While ( i = 1 ; i ≤ c ; [
    name = GetValue ( fields ; i );
    r = MBS( "XML.SetPathValue";
xml; "record." & name; 0; GetField ( name ) );
    i = i + 1 ] ; i );
  // format the result
  text = MBS( "XML.Format"; xml );
  // free json
  r = MBS( "XML.Release"; xml )
];
// return result as text
 text )
```

```xml
<?xml version="1.0" encoding="UTF-8"?>
<record>
  <First>John</First>
  <Last>Miller</Last>
  <Group/>
  <Company>Some Ltd.;</Company>
  <ID>12</ID>
</record>
```

We hope you enjoy those snippets.