

Background

You have a text in your report that consist of some variable part like the current page number embedded in some phrase. You also want to localize that phrase. There are two basic approaches:

Split up the phrase into pieces, supply each piece via parameter and concatenate them in your report. Very simple and straight-forward. But unfortunately no so good for translation if the structure of the phrase changes between languages. Lets say you have a list of clothes containing name and color.

In English your list would look like this:

a red shirt
a green beret
a blue shoe

Now in French that structure changes to:

une chemise rouge
un béret vert
un soulier bleu

And you want your report to honor that difference. Sure you could bend the report to some unified structure that'll sound more or less natural in various languages. Or you could let your translators work around it. But why not do it right?

What you need to do is to supply this structure in a way that lets the report fill in the blanks appropriately. For English that would be:

[Article] [Color] [Name]

while in French that would be:

[Article] [Name] [Color]

The Problem

Unfortunately this is not so obvious from the documentation of FastReports. If you try to assign these expressions to a report parameter from your code it will just show up as a literal string.

The Solution

Instead of assigning this expression as a value:

```
fastReport.SetParameterValue("Phrase", "[Article] [Color] [Name]");
```

you need to make one more turn to arrive at your goal:

```
fastReport.GetParameter("Phrase").Expression = "[Article] [Color] [Name]";
```

Granted, this is a somewhat exotic example and while these problems might not appear too often, they do pop up from time to time. This way you can resolve them in a clean manner without