RewriteLooping

A RewriteRule loops when the target of the rule matches the initial rewrite pattern. For example, if you were to rewrite all URLs starting with "foo" to "foo.php", then "foo.php" would also get rewritten to "foo.php", and so on, forever.

For example, consider the following rule:

```
RewriteRule ^/example http://example.com/example.html [R]
```

A simple redirect. The resulting URL matches the RewriteRule (because it starts with */example* and redirects. And again. And again. You get the idea.

This is solved by prefacing your RewriteRule with a RewriteCond to catch that condition, thus:

```
RewriteCond %{REQUEST_URI} !^/example\.html$
RewriteRule ^/example http://example.com/example.html [R]
```

That says the same thing as the previous recipe, except that it also says to only run the rule if the incoming request doesn't look like *example.html*. Problem solved. No more looping.

There is also a method involving regular expressions using a common method called "negative lookahead":

```
RedirectMatch ^/(?!example).* http://example.com/example.html
RewriteRule ^/(?!example).* http://example.com/example.html [R]
```