Different Editor plugins review

Margarita GRINVALD

July 26, 2013

In InformaWeb we try to implement a new type of question, based on highlighting text.
The highlight will work on top of a code editor with Syntax Highlight feature.

Typical usage scenarios are:

1. Teacher providing correct solutions to the questions by highlighting pieces of text
2. Students highlighting keywords
3. Students highlighting blocks representing different structures to identify

User Experience Requirements

1. Highlighting adhers to existing standards (will behave like in well known text-editors such as Word, with different color options.)
2. Availabilty of arbitrary *n* visually distinct colors for different types of elements to identify in the piece of code.
3. Dehighlighting as intuitive as possible, either by allowing the user to remove only entire blocks of highlighted text or by allowing the inverse of the highlighting process.

Implementation Details

Events fired

On user's cursor activity an event is fired and a check needs to be done, whether it was just a *mouseClick* or some piece of text was selected.

CodeMirror The event that fires is *"cursorActivity"*, and with the help of *editor.somethingSelected()* method the check is easily and correctly done by the plugin.

Ace Editor The events available are two: *changeCursor* and *changeSelection*.
With the first the method *editSession.selection.isEmpty()* always returns *false*, meaning that a simple click will trigger the creation of a new selection marker from the previous cursor position to the current one.
The second event is more "secure", meaning that simple clicks won't create new selections in the text, but the event is fired three times instead of one when a simple click occurs. This causes a problem, since the first time it's fired the method *editSession.selection.isEmpty()* returns *false*, causing therefore the creation of a new selection mark in the text.
To avoid the problem I was suggested to call the check method *isEmpty()* synchronously. This helps to solve the issue, and properly use *changeSelection* to properly respond to user activity and if needed add a selection marker.