Subject: Source Code Efficiency Minor Issue
Posted by aminhere on Wed, 08 Jun 2022 16:46:55 GMT
View Forum Message <> Reply to Message

Hello friends,

I recently discovered the U++ framework and I am having great success using it. I would like to sincerely thank the developers for all the hard work.

One recurring issue I have noticed while examining some code constructs like for example 'CodeEditor.cpp' is that all code does not make use of [else if] statements. For example, let us consider this piece of code from the previously mentioned cpp file:

```
if(IsSelection() && auto_enclose) {
 if(code == '(') {
  Enclose("(", ")");
  return true;
 }
 if(code == '{') {
  Enclose("{", "}");
  return true;
 }
 if(code == '[') {
  Enclose("[", "]");
  return true;
 }
}

// MODERATOR: use code tag four source code.
```

I am a novice programmer, and I am of the opinion that if we used [else if] statements for the 2nd and 3rd [if] cases, we would be able to save some processing power no matter how minute.
I understand that more experienced programmers like yourselves already know about this concept. That leads me to believe that this was done for some other reason. I would very much appreciate it if you would help me understand as to why this was done so that I am able to learn from you.

Thank you kindly,
Amin

Subject: Re: Source Code Efficiency Minor Issue
Posted by Klugier on Wed, 08 Jun 2022 19:09:08 GMT
View Forum Message <> Reply to Message

Hello Amin,

Welcome on our forunm

This is easy, because you have return statement within if block, so else if is redundant. The execution of if block will be break anyway. If there will be no return statement within if block, then you should use "else if" instead of "if" for performance reasons you have mentioned in your message.

Answer on stack overflow.

Klugier

Subject: Re: Source Code Efficiency Minor Issue
Posted by aminhere on Wed, 08 Jun 2022 21:33:36 GMT
View Forum Message <> Reply to Message

Hi,

Thank you for the prompt reply, I completely overlooked the returns!

Thank you,
Amin
