Subject: Re: httpRequests in secondary non-gui-main Thread
Posted by JeyCi on Tue, 03 Nov 2020 14:07:14 GMT

mirek wrote on Tue, 03 November 2020 13:57I would guess that 'showDone' is something that you call from the thread to show progress in GUI. Do you lock things propertly? E.g. with GuiLock __?

no it isn't... I just didn't care about the name of the method... but as you can notice this method is being got from another class (asyncHttp)... so I'm starting the WorkerThread as I have shown above... & I separated the loading itself to the separate class that takes vector done in WorkerThread() as Constructor's parameter & several THISFNs as parameters of the method showDone (but its real purpose is exactly loading stuff & returning results to functions of gui_class, in which I of course use GuiLock & Call([=]{this->finished.SetArray(i, va)})...
well but it was prior design, then I of course refactored for not returning results to gui... just from the beginning I wanted to see what is returning - but in the refactored code everything goes to SaveFile(n,res)... I had finally decreased the quantity of synchronization points to zero now... just have now taken for the test the shorter code - to reduce it even more for testing purposes (but it just appered to be the code still with synchronization)...
but the problem in all my step_by_step dev.process is the same - the line I've taken in MemoryIgnore... perhaps you're right - perhaps I shouldn't do loading stuff in the separate class?.. not knowing how to make this class thread-safe?.. perhaps it is not thread-save in my situation described?
mirek wrote on Tue, 03 November 2020 13:57You should have invoked that menu command I have suggested, that would put the whole backtrace snapshot on the clipboard that you could post here (e.g. as attachement).
 :o ah, I see now... ok when I will shorten the whole code even more - perhaps... in order not to provide unneeded info... (if given description doesn't clarify the surrounding of the problem)