If we want to trigger process only on any push

{

1. create a webhook in github and pass it a live url to sent any push information
2. in that url catch git json data and fetch information about pushed files
3. run pull on temp repo so that all the changes are updated in that repo
   1. pull whole repo by running GITPULL.bat from code folder to actual repo folder
4. copy pushed files (retrieved from json) to history folder through c# application
   1. Copy new/updated/deleted files from json into specific folder (history)
5. Now push these new files copied into history folder to repo by running GITPUSH.bat from code
   1. We will only push specific folder so DO NOT FORGET TO CHANGE FOLDER PATH in this bat file
   2. Make sure that GIT USER AND PASSWORD is saved in cache before running push otherwise command will fail

}

If we want to trigger process on given time interval

{

1. Process will auto start

3. At given time interval (after every 10 minutes)

}

Notes

* Use grok for creating live url of local IIS site (on command prompt **ngrok http 81**)
* (Download single file) git checkout origin/master -- DailyTaskManager/DailyTaskManager/Program.cs
* CACHE (To save repo user name/password for push run [git config --global credential.helper wincred] in git bash <https://help.github.com/articles/caching-your-github-password-in-git/#>)
* Permanent (https://git-scm.com/docs/git-credential-store)
  + $ git config credential.helper store
  + $ git push http://example.com/repo.git
  + Username: <type your username>
  + Password: <type your password>
  + [several days later]
  + $ git push http://example.com/repo.git
  + [your credentials are used automatically]
  + Run following command on repo to save email address and name

git config --global user.email "you@example.com

sgit config --global user.name "Your Name"

git config user.email “kashif\_atiq@hotmail.com”

sgit config user.name " kashifatiq"

output =

D:\Projects\GitTest>git pull origin master

Already up-to-date. (OK)

ERROR =

From https://github.com/kashifatiq/DailyTaskManager

\* branch master -> FETCH\_HEAD