Materiály - Lekce č. 01

Úvod do automatizovaného testování

## Instrukce

Všechny automatické testy budeme ukládat na

* [github.com/Radek4k/jak\_automatizovat\_testy](https://github.com/Radek4k/jak_automatizovat_testy)

Na stejné adrese najdete i příklady, které budou postupně přibývat,

## Verzovací systém GIT

Je nutný pro spolupráci a verzování zdrojových kódů automatizace.

### Návody

* Celý tutorial [itnetwork.cz/programovani/git](https://www.itnetwork.cz/programovani/git)
* Jak začít s GITem včetně instalace [rogerdudler.github.io/git-guide/](http://rogerdudler.github.io/git-guide/)
* Přehledná tabulka s příkazy [basic-git-commands](https://confluence.atlassian.com/bitbucketserver/basic-git-commands-776639767.html)

### Základní příkazy

#vypíše rozdíl repozitář vs moje lokální uložiště

git status

#přidání nového souboru do lokálního repozitáře

git add soubor.txt

#přidání všechny změny do lokálního repozitáře, vhodné pokud děláte více úprav naráz

git add -A

#zamknutí a příprava na přidání do repozitář, soubor je stáje jen u mě na lokálu

git commit -m "komentář ke commitu"

#odeslání všeho do společného repozitáře - hodí se pokud chcete uložit svoje změny pro ostatní

git push

#příjem změn ze společného repozitáře - hodí se např. pokud si chcete stáhnout změny které provedli vaši kolegové

git pull

#vypíše historii všech commitů

git log

#naklonuje úložiště do nového adresáře

git clone <https://github.com/Radek4k/jak_automatizovat_testy.git>

#odstranění souboru z repozitáře

git rm

### Další příkazy

#vytvoří prázdné úložiště Git nebo znovu inicializujte již existující

git init

#status celeho workspace

git status -s

#jak vrati zmenu zpatky, presun nejnovejsi verze z HEAD(centralni repozitar) do pracovniho repozitare

git checkout -- <filename>

#přepnutí na branch se jménem *master*

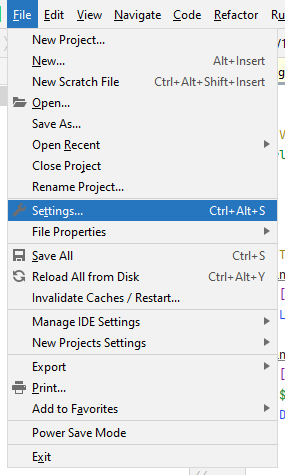
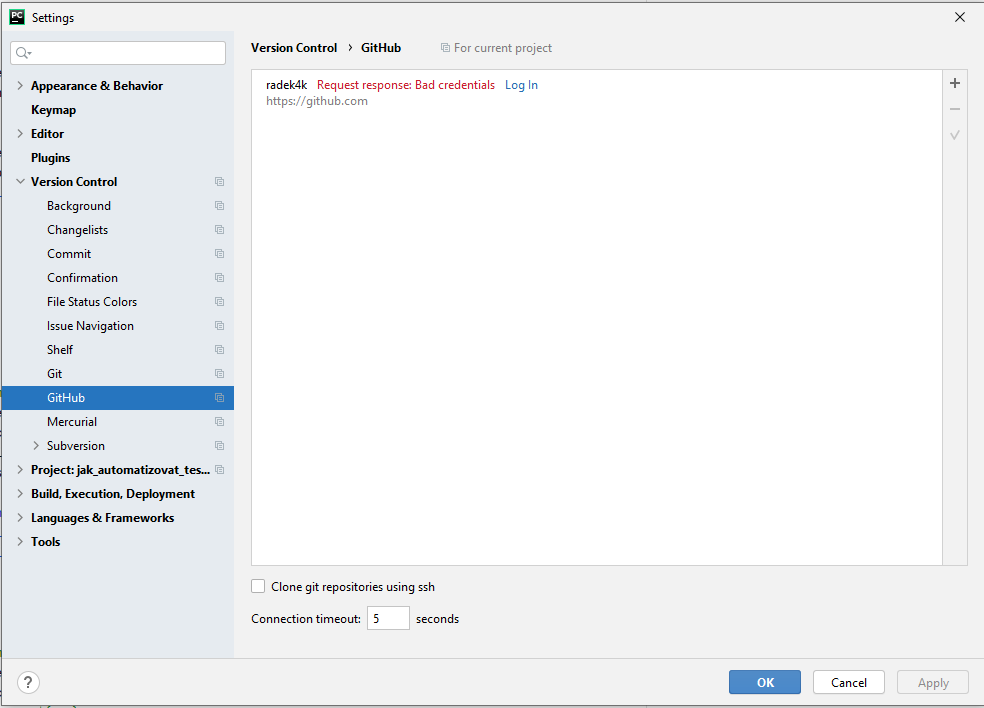
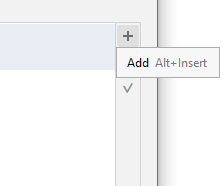
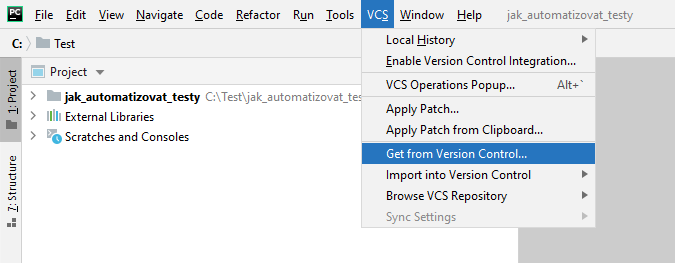
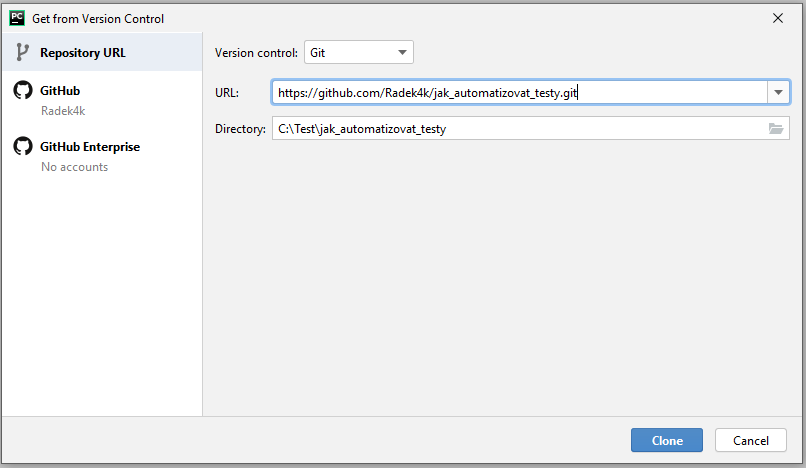
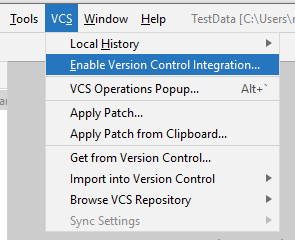
git checkout master

#jak smazat Local GitHub Repository

stačí jen smazat celý adresář a nebo pokud chcete si soubory nechat v počítači tak jen smazat adresář “.git”, který se nachází v kořenovém adresáři vašeho repozitáře

rm -rf <repo\_folder>/.git

## Pycharm + GitHub [zdroj1](https://www.jetbrains.com/help/pycharm/github.html), [zdroj2](https://www.jetbrains.com/help/pycharm/enabling-version-control.html#associate_directory_with_VCS)

* postup jak rozchodit GIT pro PyCharm a tedy se “vyhnout” složitým příkazům gitu
* díky tomu můžete stahovat i ukládat soubory na GitHub pomocí tlačítek.
* **Postup:**
* stáhněte si [PyCharm](https://www.jetbrains.com/pycharm/) (jedná se o IDE - vývojové prostředí)
* spusťte PyCharm
* vyberte v menu **File** -> **Settings** nebo stiskněte Ctrl+Alt+S
* 
* vyberte **Version Control** a potom **GitHub**
* stiskněte **+** (nachází se napravo)
* 
* Zvolit přihlášení přes **GitHub** (prohlížeč) a nebo přes **access token**. [Jak nastavit access token na GitHubu](https://docs.github.com/en/github/authenticating-to-github/keeping-your-account-and-data-secure/creating-a-personal-access-token)
* Po úspěšném propojení s GitHubem zvolete v menu **VCS** a **Get from Version Control**
* 
* Vyplňte formulář
  + GIT
  + <https://github.com/Radek4k/jak_automatizovat_testy.git>
* 
* A stiskněte **Clone**
* Tím se vám do vám zvoleného adresáře naklonuje celý repozitář
* Poslední věc co je třeba udělat je zapnout ovládání s Gitem
* v Menu vyberte **VCS** a **Enable Control Integration**
* 
* klikněte na **OK**
* pokud jste vše udělali správně ukáží se vám na pravé straně menu tyto tlačítka:
* 
* 
  + slouží ke stažení aktuální verze repozitáře ze serveru do vašeho počítače
* 
  + dělá commit, tedy uloží do vašeho lokálního repozitáře změny které jste provedli
* 
  + odešle vaše commitlé změny na server

## Visual Studio + GitHub

Připojit si uložiště (repozitář) přimo do Visual Studia je možné podle [tohoto návodu](https://docs.microsoft.com/cs-cz/visualstudio/get-started/tutorial-open-project-from-repo-visual-studio-2019?view=vs-2019&tabs=vs168later) .

## GitHub Desktop

* Jede jen na Windows
* Stáhnou je možné na [desktop.github.com](https://desktop.github.com/)
* [Návod](https://www.techrepublic.com/article/how-to-install-github-desktop/) jak nainstalovat

## Jak zapnout Syntax highlight v Pycharmu pro Robot framework

Dělá to plugin<https://plugins.jetbrains.com/plugin/7415-robot-framework-support>

Plugin je pro syntax highlight (obarvení zdrojáků).

Jak instalovat pluginy do Pycharmu najdete [zde](https://www.jetbrains.com/help/pycharm/managing-plugins.html).

## Automatizace testů

1. volba domény
2. výběr nástroje
3. každý test se skládá z
   1. akce (musí vyvolat akci, činnost něco změnit, nastavit, provést)
   2. kontroly (očekávaný výsledek AT)

## Jak dělat kontroly (assertion)

**RobotFramework (**[**návod ke všem assertům**](http://robotframework.org/robotframework/latest/libraries/BuiltIn.html)**)**

All keywords that have "should" in their name are asserts - [Should Be True](http://robotframework.org/robotframework/latest/libraries/BuiltIn.html#Should%20Be%20True), [Should Contain](http://robotframework.org/robotframework/latest/libraries/BuiltIn.html#Should%20Contain) (that's for membership - works on lists, dicts, strings - any container), [List Should Contain Sub List](http://robotframework.org/robotframework/latest/libraries/Collections.html#List%20Should%20Contain%20Sub%20List), and some even do type casting on the arguments - [Should Be Equal As Numbers](http://robotframework.org/robotframework/latest/libraries/BuiltIn.html#Should%20Be%20Equal%20As%20Integers).

## Pojmy

* repozitář
  + obecně znamená úložiště a obzvláště se toto slovo používá pro: softwarový repozitář (verzování) — datové úložiště systému správy verzí
  + místo kde jsou uložené zdrojové kódy se všemi komentáři, změnami a celou historií
* [**pip**](https://phoenixnap.com/kb/install-pip-windows)
  + je systém správy balíků používaný k instalaci a správě softwarových balíčků napsaných v Pythonu
  + je vhodný pro zjednodušení instalací frameworků, nástrojů a knihoven
  + dodatečná instalace
    - <https://www.liquidweb.com/kb/install-pip-windows/>
    - python -m pip install --upgrade pip
  + standartně je součástí Python instalace
* [**npm**](https://phoenixnap.com/kb/install-node-js-npm-on-windows) 
  + podobné jako pip ale pro JavaScript

## Nástroje a frameworky vhodné pro automatizaci

* [Robot Framework](https://robotframework.org/)
  + [instalace](https://robotframework.org/robotframework/latest/RobotFrameworkUserGuide.html#installation-instructions)
    - předpoklad: máte nainstalovaný **python 3** s [pip](https://phoenixnap.com/kb/install-pip-windows)
    - pip install robotframework
  + [jak začít](https://github.com/robotframework/QuickStartGuide/blob/master/QuickStart.rst)
  + [kompletní příručka](https://robotframework.org/robotframework/latest/RobotFrameworkUserGuide.html)
* [Cypress.io](https://www.cypress.io/)
  + [instalace](https://docs.cypress.io/guides/getting-started/installing-cypress)
    - předpoklad: máte nainstalovaný [npm](https://phoenixnap.com/kb/install-node-js-npm-on-windows)
    - npm install cypress --save-dev
  + [jak začít](https://docs.cypress.io/guides/getting-started/writing-your-first-test)
  + [kompletní příručka](https://docs.cypress.io/guides/overview/why-cypress)
* [Selenium](https://www.selenium.dev/) (zvolte Selenium WebDriver)
  + [instalace](https://www.selenium.dev/downloads/)
    - instalace obecně
      * vyberte si verzi pro jazyk ve kterém chcete programovat (ruby, python, javascript, C#, java)
      * je nutné stáhnout i příslušný [driver](https://www.selenium.dev/documentation/en/webdriver/driver_requirements/), pro konkrétní prohlížeč
    - instalace pro python s pip
      * pip install selenium
  + [jak začít](https://www.selenium.dev/documentation/en/getting_started_with_webdriver/)
  + návody
    - [kompletní příručka](https://www.selenium.dev/documentation/en/)
    - [užitečné příkazy](https://www.softwaretestinghelp.com/selenium-webdriver-commands-selenium-tutorial-17/)
    - [návod pro selenium s pythonem](https://selenium-python-test.readthedocs.io/)
* [Webdriver.io](https://webdriver.io/)
* [Appium](https://appium.io/)
* [Puppeteer](https://pptr.dev/)
* Přehled dalších nástrojů <https://kitner.cz/testovani_softwaru/testovaci-nastroje/>

## Jak rozjet CV1

* Je třeba mít nainstalovanou knihovnu **requests,** vyřešíte tímto příkazem
  + pip install robotframework-requests

## DÚ

* **Cvičení č.0: naučit se pracovat GITem**
  + buďto přes příkazovou řádku, např GitBash
  + nebo přes IDE např PyCharm, návod je výše
* **Cvičení č.1: Zvolte si doménu (weby, webové aplikace, mobilní aplikace, sítě, databáze...)**
  + rozhodněte se co chcete automatizovat
  + naplánujte si pravidelné studium dané domény min 1x týdně, min 2 hodiny
  + pošlete mi do 26.6. email (na radek@kitner.cz) co jste si vybrali
* **Cvičení č.2: Zvolte si nástroj (RobotFW, Cypress.io, Selenium...)**
  + zjistěte výhody vašeho zvoleného nástroje a zapište je
  + v tomto nástroji pak budete dělat většinu svých DÚ/projektů
  + pošlete mi do 26.6. email co jste si vybrali a seznam výhod
* **Cvičení č.3: vytvořit 2 pozitivní automatické testy**
  + testy vytvořte pro vaši doménu
  + a ve vašem zvolením nástroji
  + výsledky uložit do GITu do vašeho adresáře (git commit + git push)
  + do 26.5.
* **Cvičení č.4: vytvořit 2 negativní automatické testy**
  + testy vytvořte pro vaši doménu
  + a ve vašem zvolením nástroji
  + výsledky uložit do GITu do vašeho adresáře (git commit + git push)
  + do 26.5.

Testovací Data

***#přihlašovací údaje***

***#Login novak***

***#Heslo tajnenovak***

***#Login admin***

***#Heslo tajneadmin***