Local environment

Guide to add the following applications without admin rights. One of the objectives of this guide is to put all the applications and its configurations into the same common folder to facilitate to do manual backups or move between computers.

Contents

Manual configuration

Go to Easy go if you want a quick and semiautomatic configuration (recommended).

Download

Download these tools and applications.

Tools

Tool	Version	Download	License
Git	2.31.1	Source	Source
Java	jdk-11.0.11+9	Source	Source
Apache Maven	3.8.1	Source	Source
Node.js	14.17.0	Source	Source

Applications

Application	Version	Download	License
Visual Studio Code	1.56.1	Source	Source
IntelliJ Community	2021.1.1	Source	Source
Git Extensions	3.5	Source	Source
Postman	8.4.0	Source	Source
Fiddler Classic	`5.0.20204.45441`. Latest	Source	Source
Notepad++	7.9.5	Source	Source
Greenshot	1.2.10.6	Source	Source
ConEmu	21.04.22	Source	Source

Setup

First steps:

- Create a new folder at `C:\`, e.g. `C:\Tools`, where to install all the applications and tools. This root folder can be different, and from now on known as: `%TOOLS%`. This value will be defined as an environment variable (environment variables), but be careful if you change the value because some applications do not support environment variables in paths, and you have to adapt it.
- Inside this new folder, create two empty folders: `%TOOLS%\_config` and `%TOOLS%\_data`.
- Create the next base structure:

Base structure

```
📦Tools
 ┣ 🗀_config
 ┣ 🗀_data
 ┣ 🗀ApacheMaven
 ┃ ┗ 🗀Current
 ┣ 🗀ConEmu
 ┃ ┗ 🗀Current
 ┣ 🗀Fiddler
 ┃ ┗ 🗀Current
 ┣ 🗀Git
 ┃ ┗ 🗀Current
 ┣ 🗀GitExtensions
 ┃ ┗ 🗀Current
 ┣ 🗀Greenshot
 ┃ ┗ 🗀Current
```

```
📦Tools
 ┣ 🗀Java
 ┃ ┗ 🗀OpenJDK
 ┃ ┃ ┗ 🗀Current
 ┣ 🗀JetBrains
 ┃ ┗ 🗀IntelliJ
 ┃ ┃ ┗ 🗀Current
 ┣ 🗀Node.js
 ┃ ┗ 🗀Current
 ┣ 🗀Notepad++
 ┃ ┗ 🗀Current
 ┣ 🗀Postman
 ┃ ┗ 🗀Current
 ┗ 🗀VSCode
 ┃ ┗ 🗀Current
```

Install

To finish the installation process, we should move the applications to the target folder. As a summary, the `.zip` and `.7z` files should be unzipped (without the root folder) in the target application `Current` folder. The `.exe` files have a specific configuration. A short description of the process for each:

- Git (`PortableGit-2.31.1-64-bit.7z.exe`): unzip the `.exe` in `%TOOLS%\Git\Current` and run `post-install.bat`, or just run the `.exe` and move the extracted content to `%TOOLS%\Git\Current`.
- Java (`OpenJDK11U-jdk_x64_windows_hotspot_11.0.11_9.zip`): unzip in `%TOOLS%\Java\OpenJDK\Current`.
- Apache Maven (`apache-maven-3.8.1-bin.zip`): unzip in `%TOOLS%\ApacheMaven\Current`.
- Node.js (`node-v14.17.0-win-x64.zip`): unzip in `%TOOLS%\Node.js\Current`.
- Visual Studio Code (`VSCode-win32-x64-1.56.1.zip`): unzip in `%TOOLS%\VSCode\Current` and create `data` folder within Visual Studio Code folder. Portable mode.
- IntelliJ Community (`ideaIC-2021.1.1.win.zip`): unzip in `%TOOLS%\JetBrains\IntelliJ\Current`.
  - Open `C:\Tools\JetBrains\IntelliJ\Current\bin\idea.properties` file, uncomment and change next properties:

```
idea.config.path=${idea.home.path}/data/config
idea.system.path=${idea.home.path}/data/system
idea.plugins.path=${idea.home.path}/data/plugins
idea.log.path=${idea.home.path}/data/log
```

- Git Extensions (`GitExtensions-Portable-3.5.0.11713-0a5ef9ca6.zip`): unzip in `%TOOLS%\GitExtensions\Current` .
- Postman (`Postman-win64-8.4.0-Setup.exe`): run the `.exe` and when the installation process finishes, move the content in `%LOCALAPPDATA%\Postman\app-8.4.0` to `%TOOLS%\Postman\Current` .
- Fiddler Classic (`FiddlerSetup.exe`): unzip the `.exe` , open unzipped folder `$PLUGINSDIR` , and unzip again the `.exe` in `%TOOLS%\Fiddler\Current` . Details.
- Notepad++ (`npp.7.9.5.portable.x64.zip`): unzip in `%TOOLS%\Notepad++\Current` .
- Greenshot (`Greenshot-NO-INSTALLER-1.2.10.6-RELEASE.zip`): unzip in `%TOOLS%\Greenshot\Current` .
- ConEmu (`ConEmuPack.210422.7z`): unzip in `%TOOLS%\ConEmu\Current` . Also included in Git Extensions. Portable instructions, 7-Zip package and FAQ.

Result structure (cut to only show two folders or files)

```
📦Tools
┣ 🗀_config
┣ 🗀_data
┣ 🗀ApacheMaven
┃ ┗ 🗀Current
┃ ┃ ┣ 🗀bin
┃ ┃ ┗ 🗀boot
┣ 🗀ConEmu
┃ ┗ 🗀Current
┃ ┃ ┣ 🗀ConEmu
┃ ┃ ┗ 🗀plugins
┣ 🗀Fiddler
┃ ┗ 🗀Current
┃ ┃ ┣ 🗀$PLUGINSDIR
┃ ┃ ┗ 🗀ImportExport
┣ 🗀Git
┃ ┗ 🗀Current
┃ ┃ ┣ 🗀bin
┃ ┃ ┗ 🗀cmd
┣ 🗀GitExtensions
┃ ┗ 🗀Current
┃ ┃ ┣ 🗀ConEmu
┃ ┃ ┗ 🗀cs
┣ 🗀Greenshot
┃ ┗ 🗀Current
┃ ┃ ┣ 🗀Languages
┃ ┃ ┗ 🗀Plugins
```

```
📦Tools
┣ 🗀Java
┃ ┗ 🗀OpenJDK
┃ ┃ ┗ 🗀Current
┃ ┃ ┃ ┣ 🗀bin
┃ ┃ ┃ ┗ 🗀conf
┣ 🗀JetBrains
┃ ┗ 🗀IntelliJ
┃ ┃ ┗ 🗀Current
┃ ┃ ┃ ┣ 🗀bin
┃ ┃ ┃ ┗ 🗀jbr
┣ 🗀Node.js
┃ ┗ 🗀Current
┃ ┃ ┣ 🗀node_modules
┃ ┃ ┗ 📜CHANGELOG.md
┣ 🗀Notepad++
┃ ┗ 🗀Current
┃ ┃ ┣ 🗀autoCompletion
┃ ┃ ┗ 🗀functionList
┣ 🗀Postman
┃ ┗ 🗀Current
┃ ┃ ┣ 🗀locales
┃ ┃ ┗ 🗀resources
┗ 🗀VSCode
┃ ┗ 🗀Current
┃ ┃ ┣ 🗀bin
┃ ┃ ┗ 🗀data
```

ℹ️ Is important to verify the structure to do not have problems with extra folder when unzipping. ℹ️

Configuration

Git

- Create a new file or open it if it exists: `%USERPROFILE%\.gitconfig`.
- Recommended configuration. Update with your own information for, at least, `user.name` and `user.email`.

```
[credential]
  helper = manager
[http]
  sslVerify = false
[user]
  name = Example
  email = example@example.com
[core]
  editor = \"C:/Tools/GitExtensions/Current/GitExtensions.exe\" fileeditor
[diff]
  guitool = vscode
[difftool "vscode"]
  path = C:/Tools/VSCode/Current/Code.exe
  cmd = \"C:/Tools/VSCode/Current/Code.exe\" --wait --diff \"$LOCAL\" \"$REMOTE\"
[merge]
  tool = vscode
  guitool = vscode
[mergetool "vscode"]
  path = C:/Tools/VSCode/Current/Code.exe
  cmd = \"C:/Tools/VSCode/Current/Code.exe\" --wait \"$MERGED\"
[pull]
  rebase = false
[fetch]
  prune = false
[rebase]
  autoStash = false
```

⚠️ If you changed the `%TOOLS%` folder to a different one, update the paths too. ⚠️

Apache Maven

At the beginning of the file `%TOOLS%\ApacheMaven\Current\conf\settings.xml` add the `localRepository`:

```
<settings>
  <localRepository>${env.TOOLS}\_data\Maven\repository</localRepository>
  <!-- Existing configuration... -->
<settings>
```

Node.js

NPM

- Create a new file: `%TOOLS%\_config\.npmrc`, or move from `%USERPROFILE%\.npmrc` if it exists.
- Add next properties:

  ```
  prefix=${TOOLS}\_data\npm
  cache=${TOOLS}\_data\npm\cache
  tmp=${TOOLS}\_data\npm\tmp
  ```

Install recommended packages:

- Install Angular: `npm i -g @angular/cli`.
- Install Yarn: `npm i -g yarn`.

Yarn

- Create a new file or open it if it exists: `%USERPROFILE%\.yarnrc`.
- Add next properties:

  ```
  prefix "C:\\Tools\\_data\\Yarn"
  global-folder "C:\\Tools\\_data\\Yarn\\global"
  cache-folder "C:\\Tools\\_data\\Yarn\\cache"
  "--link-folder" "C:\\Tools\\_data\\Yarn\\link"
  ```

⚠ If you changed the `%TOOLS%` folder to a different one, update the paths too. ⚠

IntelliJ Community

Open IntelliJ and apply basic configuration opening `Customize` > `All settings...`:

- Go to `File | Settings | Build, Execution, Deployment | Build Tools | Maven | Importing`: at *JDK for importer* select `Use JAVA_HOME`.
- Go to `File | Settings | Build, Execution, Deployment | Build Tools | Maven` and update:
  - *Maven home path*: `C:\Tools\ApacheMaven\Current`.
  - *User settings file*: `C:\Tools\ApacheMaven\Current\conf\settings.xml`.
  - *Local repository*: `C:\Tools\_data\Maven\repository`.
  - ⚠ If you changed the `%TOOLS%` folder to a different one, update the paths. ⚠

ConEmu

When opening for the first time, select the ConEmu folder to save settings: `%TOOLS%\ConEmu\Current\ConEmu.xml`.

Environment variables

Configuration needed for applications to: properly work, can be used in command line, be discoverable by others applications.

- Open environment variables: `Win` + `R` > `cmd` > `rundll32.exe sysdm.cpl,EditEnvironmentVariables`.
- Configure environment variables
  - Add new user variables (or update):

        - Variable name: `TOOLS`, variable value: `C:\Tools`.
        - Variable name: `JAVA_HOME`, variable value: `%TOOLS%\Java\OpenJDK\Current`.
        - Variable name: `NPM_CONFIG_USERCONFIG`, variable value: `%TOOLS%\_config\.npmrc`.
    - `PATH` variable:
        - Double-click on `PATH` (or create if it does not exist).
        - In the new window click on `Edit text...`.
        - Add at the end: `;%TOOLS%\ApacheMaven\Current\bin;%TOOLS%\Git\Current\cmd;%TOOLS%\Java\OpenJDK\Current\bin;%TOOLS%\Node.js\Current;%TOOLS%\VSCode\Current\bin;%TOOLS%\_data\npm;`.

Easy go

In the `.\Tools` folder (documentation root), all the applications are nearly fully configured with this setup and only needs some minor actions.

⚠️ Before running the script(s) read carefully next comments: ⚠️

- Run this at your own risk.
- If some environment variables are already present in `PATH`, these will be duplicated and other variables with the same name will be overwritten.
- Verify the configuration at the end. Use log files to restore previous values if something is wrong.

Actions:

- Copy the content in `.\Tools` to `C:\Tools`.
- Run the script to create all the needed environment variables: `call .\Scripts\create_env_vars.cmd`.
- Apply the steps defined in: Git and Yarn.
- In case that you have some existing configuration files, remove them, or move the content and then remove: `%USERPROFILE%\.m2\settings.xml`, `%USERPROFILE%\.npmrc`.

WIP

- Explain why `Current` structure and empty folder with version.
- Clean up the system: old configuration and programs.
- Script to automatically download the *latest* version of all the programs.
- Backup of the `C:\Tools` folder.
  - Create .zip with all: applications, `_data` (optional) and `_config`.
  - Copy Chrome bookmarks.

Tools and applications

- Python: latest and 2.7.
- nvm.
- Yarn 2+.
- MiKTeX.
- Microsoft Teams.
- Fiddler Everywhere.
- SoapUI.
- Cmder.
- Eclipse.