❓ Why `REY_FetchV4` ?

1. What is `REY_FetchV4` ?

- it's like a ***'Package Manager'*** but built with CMake

2. When will YOU need `REY_FetchV4` ?

- if your project requires some other library made by some other person on github.
- (assuming that it's a project in C/C++ or similar ones) with which `cmake` works

3. Why is this important?

- Managing External Library is a big hassle.... i can't describe in words >_<

4. Why yet another package manager?

- Well there are many many out there. But non with `cmake` ....
- even tho `cmake` is very popular and used literally almost by every popular tools/libraries you are gonna use

5. Why should YOU use `REY_FetchV4` ?

- can help a lot 😁 if you are someone who forgets `git clone/recursive/submodule` stuffs
- many ways to Fetch/Grab
  i. `Scout [/find/look-for]`
  ii. `[Git] Submodule`
  iii. `ZipLinks`
  iv. `[Git] Clone/Fetch` ☺
  - see `.\REY_FetchV4\REY_FetchV4_X_DOCS.cmake`
- Super Easy to include a new package support
  - e.g. Adding support for `REY_FetchV4_rapidyaml.cmake` ---> Took me about a minute
  - /see guide below

6. How does `REY_FetchV4` work?

- Rest of the document is dedicated to answer this question ☺

🎯 V4 [Short]

1. `REY_FetchV4.cmake` :- THE SOFTWARE
2. `REY_FetchV4_X. <lib-name>.cmake` :- "X" = variables 😊
3. `REY_FetchV4_MOD.<lib-name>.cmake` :- Modifying Variables ☺
4. `REY_FetchV4. <lib-name>.cmake` :- INTERFACE = Includes and Patches/Joins/Connets Together 😊
5. `REY_FetchV4_X_RESET.cmake` :- RESET ---> EMPTY-STRINGs

```
# 4 Ways/Flavours [more to be added]
# ----------------------------------------
    1. find_library() + find_path()
    2. Git_SubModule()
    3. Zip_Links()
    4. Git_Clone() [faster than CMAKE_FETCH]

Pseudocode inside REY_FetchV4.cmake
# ----------------------------------------
```

📜 V4 [Long]

1. `REY_FetchV4.cmake` :- The Goddamn Main file, where all the work happens
2. `REY_FetchV4_X.<lib-name>.cmake` :- "X" = variables 😊
3. `REY_FetchV4_MOD.<lib-name>.cmake` :- imagine B uses A. C uses B. C also directly uses A. C wants to control how B fetches A.
   - C should have a copy of this file & SET `REY_FetchV4_MODS_PATH`
4. `REY_FetchV4.<lib-name>.cmake` :- includes the above two thingies
   - Here in These files ---> We can do Library Specific Extra Stuffs
5. `REY_FetchV4_X_Reset.cmake` :- RESETs the Variables back to EMPTY-STRING

👀 Official Alternative:- `FetchContent_MakeAvailable()`

example:- https://rapidyaml.readthedocs.io/latest/sphinx_using.html#quickstart-build-samples

🛠 1. How to add new Packages

1. ***make & rename a copy of*** `REY_FetchV4.rapidyaml/vulkan/fmt.cmake`
   - [all of these 3 files are identical, only change is library name]
   i. change the library name inside that new copied file to `<new package that you wanna have support for>`
2. ***make & rename a copy of*** `REY_FetchV4_X.rapidyaml/vulkan/fmt.cmake`
   - choose which of the 4-Ways of fetching you want! & change those variables ☺
3. *yes, it's soooo easy, just like scoop.sh* 😁
4. if you want some extra `feature/functionality/new way to grab or fetch` inside `REY_FetchV4.cmake`
   - create me an issue on Github ☺

🍣 [old docs - before release] Why `REY_FetchV4`

```
0. REY_FetchV4 --> Mix of it all below
    +5:- Use Github to keep track of LIBS & Include Versions
         https://github.com/REYNEP/libs-win32
         https://github.com/REYNEP/libs-include

1. How Blender Does it --> 4/10
    -1:- Libs & Include has to be Manually Updated.

2. CMAKE FETCH --> 7/10
    -1:- Download Progress is shown by taking up so many lines, like hundreds.
    -2:- Can't Fetch SPECIFIC Folder/SubDirectory from a repo from GITHUB

3. Python Script to Download --> 5/10
4. GIT SubModule --> 4/10
5. Maintaining a .zip in GDrive --> 2/10
6. Just supplying all the external library binaries along with the project souurce --> 1/10
```

```
# Problem:- Authors that are gonna use "REY_LoggerNUtils" will need to modify variables inside
".CMakeFiles/REY_FetchV3_fmt.cmake"
#           But How can they? That file is inside REYNEP's "REY_LoggerNUtils" Repo.

# Solution-1 [HECTIC]
#     the variables from inside of ".CMakeFiles/REY_FetchV3_fmt.cmake"
#     Authors can just set those themselves before add_subdirectory(REY_LoggerNUtils)
#         HECTIC ISSUE:- after add_subdirector(REY_LoggerNUtils) is done ---> Authors also need
 to RESET Those Variables to EMPTY

# Solution-2 [HECTIC]
#     have a variable inside REY_LoggerNUtils/CMakeLists.txt to control PRIORITY_BASED_LOCATION
 for ".CMakeFiles/REY_FetchV3_fmt.cmake"
#     like authors that are gonna use REY_LoggerNUtils, having the modified copy of
".CMakeFiles/REY_FetchV3_fmt.cmake"
#          and then they are directing REY_LoggerNUtils/CMakeLists.txt to include their
MODIFIED COPY
#
#           HECTIC ISSUE:- But this would mean that REY_Fetch can't ever be like a PACKAGE
Manager itself....

# Solution-3 [WORKS & Feels like a Package Manager]
```

```
# The Idea is:- REY_LoggerNUtils/CMakeLists.txt includes ".CMakeFiles/REY_FetchV3_fmt.cmake"

# Now, obviously any library that uses REY_LoggerNUtils, can't modify
REY_LoggerNUtils/CMakeLists.txt
#     +obviously any library that uses REY_LoggerNUtils, can't modify ".CMakeFiles/
REY_FetchV3_fmt.cmake"

# 1. Having multiple copies of ".CMakeFiles/REY_FetchV3_fmt.cmake" felt dumb
# 2.
```

My Approach to External Libraries:- `CMAKE_FETCH` + Blender's Approach

```
1. CMAKE_FETCH most stuffs ----> like fmt
2. .libs_REYMOD = Libraries CUT/Stripped/Modified by REYNEP ----> like vulkan
    Supplying these like Blender supplies their External Libraries

3. Even the stuff inside .libs_REYMOD would be fetches by CMAKE 😁
```