
Announcement

[Guillaume Rongier](#) · Oct 8, 2021

[Open Exchange](#)

fhir-scraper

A simple python script to copy/scrap/crawl a FHIR repository to another one.

The script is written in Python 3.

Install it from pip

```
python -m venv .venv
source .venv/bin/activate
pip install git+https://github.com/grongierisc/fhir-scraper
cp .venv/bin/fhir-scraper.py .
```

Edit fhir-scraper.py for your needs.

Run it :

```
python fhir-scraper.py
```

How to run it from git

First clone this repository.

```
git clone https://github.com/grongierisc/fhir-scraper
```

You can install it quickly into a Virtual Environment. First, you need to move to the source folder and create an Virtual Environment (all commands are for Unix-based OS and may need tweaking on Windows):

```
python -m venv .venv
source .venv/bin/activate
```

On Windows without Windows Subsystem for Linux, you will need to change the last command to `.venv/bin/activate.bat`.

These commands will create a new directory, visit it, create the virtual environment, and activate it. Then you can install the dependencies.

```
pip install .
```

Before running it, you can refer to the configuration.

Now you can run the app.

```
python .venv/bin/fhir-scraper.py
```

How to configure it

Edit the app.py file to configure the variable in main.

```
### Source Repository
urlClient = "http://localhost:52773/fhir/r4/"

### Source authorization heard
source_authorization=None

### Destination Repository
urlServer = "http://localhost:32783/fhir/r4/"

### Destination authorization heard
target_authorization=None

### List of resource to fetch
### If None, the script will fetch all available resources
p_resources=None

### Limit the number of resource to fetch
p_limit=100
```

How it works

By default the algorithm will read the metadata of the source.

For each resource found, it will retrieve it, transform it and insert it in the target with the same ID.

The transformation is done by the transform function, it is based on the keys/values of the resource. In this example, the transformation ignores the endpoint keys and transforms the id values.

In case of error, the script moves to the next resource.

Future Development

If this script is successful, I will try to optimize its development for example by adding parameters with the python click module and maybe publish it on Pypi.

[#FHIR #InterSystems IRIS for Health](#)

[Check the related application on InterSystems Open Exchange](#)