

# xl2roefact PyPi Library

- [xl2roefact PyPi Library](#)
  - [System modules](#)
    - [rdinv module logic](#)
  - [Install library](#)
    - [Install from PyPi](#)
  - [Working directories](#)
  - [Aspecte tehnice referitoare la formatul fisierului JSON aferent facturii](#)
  - [Detailed design technical documentation](#)

## System modules

`xl2roefact` main and basic modules are:

- `rdinv` read an Excel file and extract invoice data to a JSON file format
- `wrxml` write, convert the JSON invoice file to a XML file format, respecting schemes required by *RO EFact* standard
- `chkxml` check generated XML file
- `ldxml` load an invoice (ie, its XML associated file) to *ANAF SPV system*
- `chkisld` check if an invoice is already loaded in *ANAF SPV system*
- `config_settings` define system settings & parameters mainly used in invoice info / data detection and extract from invoice Excel format file
- `app_cli` contains the code for `xl2roefact` application command line (CLI) format

Below is presented the **skeleton logic** of those modules which and where is relevant ie meaning where is not enough obvious from code or code complexity exceed usual limits (*for example more than 100 lines of code per function*). For more technical details and specification regarding modules [see 810.05a-xl2roefact\\_DLD\\_specs.md file](#)

## rdinv module logic

Main function of `rdinv` module is `rdinv(...)` which has the following logic sections which are in **strict sequence in presented order**:

- *search of `invoice_items_area` subtable*. This area is expected to contain invoice lines and is "processed" first because it is more structured and easier to identify; after its identification the header area is considered upper of

it and footer area below it

- solve `invoice_items_area` in 2 step.... In this step the code-data-variables of items area will be initialized in order to hold information that will be found
- *localize and mark areas for...* section that follows natural the previous one by initializing code-data-variables for header and footer areas to hold their corresponding information
- solve `invoice_header_area` detailed initialize of header area code-data-variables
- *ReNaSt -RegNameStrategy* section that identify and extract the legal registered name of invoice customer
- *section to ( Excel data )-->( JSON ) format preparation and finishing* section which prepare Excel original data found to be saved as JSON as a more "electronic interchangeable" structure
- for more details about code logic description and presentation, please contact [RENware Software Systems](#)

## Install library

... explain there are 2 methods... \* install from PyPi \* install from sources (GitHub) #TODO...make\_it

### Install from PyPi

<--! #FIXME translate to EN --> Instalarea acesteia se face cu instrumentele standard Python. Recomandarea este pentru instalarea simpla cu: `pip install xl2roefact`, biblioteca fiind disponibila in repository-ul standard PyPy. Pentru instalarea din surse, biblioteca poate fi descarcata din [GitHub](#).

## Working directories

Below is a short description of most important directories that will (can !) be found on local development environment.

- `invoice_files/` default directory for Excel files which is intended to be processed
- `build/` this directory which will contain intermediary files resulted from building CLI application, library distribution parts, etc. Directory is subject of `.gitignore`
- `dist/` package files (wheels, dist), Windows executables, etc, generally all files subject of "public" distribution and download
- `test_*/` contains test invoice samples (from client, a RENware one, a 3rd party one) and some useful specs in dev & test process

## Aspecte tehnice referitoare la formatul fisierului JSON aferent facturii

Acest fisier este cel generat de catre aplicatie in urma executiei acesteia cu comanda `xl2json`. Structura de baza a acestui fisier este:

```
{
  "Invoice": {...},
  "meta_info": {...},
  "excel_original_data": {...}
}
```

Cheile de la primul nivel contin:

- **Invoice** - datele efective ale facturii
- **meta\_info** - informatii referitoare la procesarea facturii si mapa de conversie a cheii **Invoice** din formatul JSON in formatul XML cerut de sistemul *RO E-Fact*
- **excel\_original\_data** - informatiile originale din fisierul Excel, asa cum au fost ele identificate si gasite precum si locatia (adresele celulelor). Aceste informatii sunt utile in cazul in care exista neclaritati in urma procesului de conversie pentru "a intelege" de unde si cum arata informatiile originale din fisierul Excel

An [example of JSON generated file is available here](#)

## Detailed design technical documentation

---

Last update: February 2, 2024