# Spectronaut Linux custom parsing rule

To use FASTA files with custom header parsing rule in Spectronaut Linux, you need to import it using the `convertFASTA` command:

```
spectronaut convertFASTA -fasta <path_to_fasta_file> --parsingRule <path_to_parsing_rule_json> -o <output_folder>
```

This command will produce the files in Managed FASTA format .bgsfasta and put them in the `output_folder`. These files can be further used in search pipelines (similarly as for regular FASTA files with `-fasta` option).

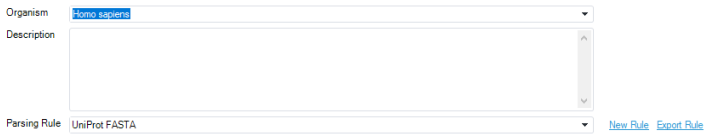
Parsing rule .json file can be created either in Windows Spectronaut Viewer or manually.

## Creating parsing rule .json file

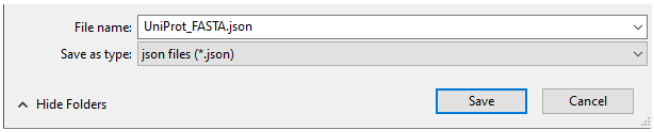
### With Spectronaut Viewer on Windows

The easiest way to get the parsing rule .json file is to use Windows Spectronaut Viewer GUI with a parsing rule creator and exporter.

Go to the **Database** perspective → **Protein Databases** and select any database imported from FASTA file. To create a new parsing rule, select **New rule**. To export the existing parsing rule, select the parsing rule of interest and press **Export Rule**.




Select the desired directory, and **Save as type: json files (*.json)**.




### Manually

The parsing rule .json file has a structure as follows:

```
{
    "name":"UniProt FASTA",
    "proteinIDToken":"Accession",
    "keywords":["Database","Accession","UniProtId","Protein Name","Protein Description","Organism","OrganismId","Gene","Protein Existence","Sequence Version"],
    "parsingRule":">$[Database]$|$[Accession]$[UniProtId]$|$[Protein Name]$ $[Protein Description]$ OS=$[Organism]$ OX=$[OrganismId]$ GN=$[Gene]$ PE=$[Protein Existen
  }
```

It is a JSON object with the following names:

1. “name” describing the parsing rule name,
2. “proteinIDToken” indicating which token from a list of “keywords” contains information about protein ID,
3. “keywords” being a JSON array of all tokens that will be matched in a parsing rule,
4. “parsingRule“ containing a parsing rule.

The parsing rule is a JSON string containing a series of alternating tokens and plain text fragments delimited with the symbol “$”. Plain text is matched exactly in the parsing rule, tokens are substituting a fragment of the parsing rule by a named value. Tokens used in a parsing rule should be escaped with brackets (“[“, “]”). If there is no plain-text fragment between two tokens, then they both match the whole respective fragment of a parsing rule.

For example, for the parsing rule above:

1. The rule expects that the header has a “>” character in the beginning.
2. Then all of the incoming text is kept as an Accession token until the “|” character.
3. Then all of the incoming text after the “|” sign is kept as Accession and UniProtId tokens until the next “|” character.
4. Then all of the incoming text is kept under the Protein Name token until the space “ “ character appears.
5. Etc.

### Report

Fragments of the FASTA header parsed by token `[Example token]` are later available in the report as a column "`PG.Example token`". If you want to have parsed values as default Spectronaut report columns, please use token names from the following list:

`"Database","Accession","UniProtId","Protein Name","Protein Description","Organism","OrganismId","Gene","Protein Existence","Sequence Version"`