

---

NAME

RDKitSearchSMARTS.py - Perform a substructure search using SMARTS pattern

## SYNOPSIS

```
RDKitSearchSMARTS.py [--infileParams <Name,Value,...>] [--mode <retrieve or count>] [--negate <yes or no>] [--outfileParams <Name,Value,...>] [--overwrite] [--useChirality <yes or no>] [-w <dir>] [-o <outfile>] -p <SMARTS> -i <infile>
```

```
RDKitSearchSMARTS.py -h | --help | -e | --examples
```

## DESCRIPTION

Perform a substructure search in an input file using specified SMARTS pattern and write out the matched molecules to an output file or simply count the number of matches.

The supported input file formats are: SD (.sdf, .sd), SMILES (.smi., csv, .tsv, .txt)

The supported output file formats are: SD (.sdf, .sd), SMILES (.smi)

## OPTIONS

-e, --examples

Print examples.

-h, --help

Print this help message.

-i, --infile <infile>

Input file name.

--infileParams <Name,Value,...> [default: auto]

A comma delimited list of parameter name and value pairs for reading molecules from files. The supported parameter names for different file formats, along with their default values, are shown below:

```
SD, MOL: removeHydrogens,yes,sanitize,yes,strictParsing,yes
SMILES: smilesColumn,1,smilesNameColumn,2,smilesDelimiter,space,
        smilesTitleLine,auto,sanitize,yes
```

Possible values for smilesDelimiter: space, comma or tab.

-m, --mode <retrieve or count> [default: retrieve]

Specify whether to retrieve and write out matched molecules to an output file or simply count the number of matches.

-n, --negate <yes or no> [default: no]

Specify whether to find molecules not matching the specified SMARTS pattern.

-o, --outfile <outfile>

Output file name.

--outfileParams <Name,Value,...> [default: auto]

A comma delimited list of parameter name and value pairs for writing molecules to files. The supported parameter names for different file formats, along with their default values, are shown below:

```
SD: compute2DCoords,auto,kekulize,no
SMILES: kekulize,no,smilesDelimiter,space, smilesIsomeric,yes,
        smilesTitleLine,yes
```

Default value for compute2DCoords: yes for SMILES input file; no for all other file types.

--overwrite

Overwrite existing files.

-p, --pattern <SMARTS> [default: none]

SMARTS pattern for performing search.

-u, --useChirality <yes or no> [default: no]

Use stereochemistry information for SMARTS search.

-w, --workingdir <dir>

Location of working directory which defaults to the current directory.

## EXAMPLES

To retrieve molecules containing the substructure corresponding to a specified SMARTS pattern and write out a SMILES file, type:

```
% RDKitSearchSMARTS.py -p 'clcccccl' -i Sample.smi -o SampleOut.smi
```

To only count the number of molecules containing the substructure corresponding to a specified SMARTS pattern without writing out any file, type:

```
% RDKitSearchSMARTS.py -m count -p 'clcccccl' -i Sample.smi
```

To count the number of molecules in a SD file not containing the substructure corresponding to a specified SMARTS pattern and write out a SD file, type:

```
% RDKitSearchSMARTS.py -n yes -p 'clcccccl' -i Sample.sdf -o SampleOut.sdf
```

To retrieve molecules containing the substructure corresponding to a specified SMARTS pattern from a CSV SMILES file, SMILES strings in column 1, name in and write out a SD file, type:

```
% RDKitSearchSMARTS.py -p 'clcccccl' --infileParams  
  "smilesDelimiter,comma,smilesTitleLine,yes,smilesColumn,1,  
  smilesNameColumn,2" --outfileParams "compute2DCoords,yes"  
  -i SampleSMILES.csv -o SampleOut.sdf
```

## AUTHOR

Manish Sud(msud@san.rr.com)

## SEE ALSO

RDKitConvertFileFormat.py, RDKitFilterPAINS.py, RDKitSearchFunctionalGroups.py

## COPYRIGHT

Copyright (C) 2018 Manish Sud. All rights reserved.

The functionality available in this script is implemented using RDKit, an open source toolkit for cheminformatics developed by Greg Landrum.

This file is part of MayaChemTools.

MayaChemTools is free software; you can redistribute it and/or modify it under the terms of the GNU Lesser General Public License as published by the Free Software Foundation; either version 3 of the License, or (at your option) any later version.