## NAME

RDKitDrawMolecules.py - Draw molecules and generate an image file

## SYNOPSIS

RDKitDrawMolecules.py [--alignmentSMARTS <SMARTS>] [--atomLabelFontSize <number>] [ --bondLineWidth <number>] [--compute2DCoords <yes | no>] [--fontBold <yes or no>] [ --highlightSMARTS <SMARTS>] [--infileParams <Name,Value,...>] [--kekulize <yes or no>] [ --molImageSize <width,height>] [--numOfMolsPerRow <number>] [--overwrite] [--showMolName <yes or no>] [-w <dir>] -i <infile> -o <outfile>

RDKitDrawMolecules.py -h | --help | -e | --examples

## DESCRIPTION

Draw molecules in a grid and write them out as an image file. The SVG image file appears to be the best among all the available image file options, as rendered in a browser. The Python modules aggdraw/cairo are required to generate high quality PNG images.

The options '--atomLabelFontSize' and '--bondLineWidth' don't appear to work during generation of a SVG image.

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

The output image file can be saved in any format supported by the Python Image Library (PIL). The image format is automatically detected from the output file extension.

Some of the most common output image file formats are: GIF (.gif), JPEG (.jpg), PNG (.png), SVG (.svg), TIFF (.tif)

## OPTIONS

--alignmentSMARTS <SMARTS> [default: none]

SMARTS pattern for aligning molecules to a common template.

--atomLabelFontSize <number> [default: 12]

Font size for drawing atom labels. This option is ignored during generation of a SVG output file.

-b, --bondLineWidth <number> [default: 1.2]

Line width for drawing bonds. This option is ignored during generation of a SVG output file.

-c, --compute2DCoords <yes | no> [default: auto]

Compute 2D coordinates of molecules before drawing. Default: yes for SMILES file; no for all other file types.

-e, --examples

Print examples.

-f --fontBold <yes or no> [default: yes]

Make all text fonts bold during generation of a SVG output file. This option is ignored for all other output files.

-h, --help

Print this help message.

--highlightSMARTS <SMARTS> [default: none]

SMARTS pattern for highlighting atoms and bonds in molecules. All matched substructures are highlighted.

-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.

-k, --kekulize <yes or no> [default: yes]

Perform kekulization on molecules. This option is ignored during generation of a SVG output file.

-m, --molImageSize <width,height> [default: 400,300]

Image size of a molecule in pixels.

-n, --numOfMolsPerRow <number> [default: 2]

Number of molecules to draw in a row.

-o, --outfile <outfile>

Output file name.

--overwrite

Overwrite existing files.

-s, --showMolName <yes or no> [default: yes]

Show molecule names under the images.

-w, --workingdir <dir>

Location of working directory which defaults to the current directory.

## EXAMPLES

To automatically compute 2D coordinates for molecules in a SMILES file and generate a SVG image file containing 2 molecules per row in a grid with cell size of 400 x 300 pixels, type:

```
% RDKitDrawMolecules.py -i Sample.smi -o SampleOut.svg
```

To automatically compute 2D coordinates for molecules in a SMILES file and generate a SVG image file containing 4 molecules per row in a grid with cell size of 200 x 200 pixels and without any keulization along with highlighting a specific set of atoms and bonds indicated by a SMARTS pattern, type:

```
% RDKitDrawMolecules.py -n 4 -m "200,200" -k no --fontBold no
  --highlightSMARTS  'c1ccccc1' -i Sample.smi -o SampleOut.svg
```

To generate a PNG image file for molecules in a SD file using existing 2D coordinates, type

```
% RDKitDrawMolecules.py --compute2DCoords no -i Sample.sdf
  -o SampleOut.png
```

To automatically compute 2D coordinates for molecules in a CSV SMILES file with column headers, SMILES strings in column 1, and name in column 2 and generate a PDF image file, type:

```
% RDKitDrawMolecules.py --infileParams "smilesDelimiter,comma,
  smilesTitleLine,yes,smilesColumn,1,smilesNameColumn,2"
  -i SampleSMILES.csv -o SampleOut.pdf
```

## AUTHOR

Manish Sud(msud@san.rr.com)

## SEE ALSO

RDKitConvertFileFormat.py, RDKitRemoveDuplicateMolecules.py, RDKitSearchFunctionalGroups.py, RDKitSearchSMARTS.py

## COPYRIGHT