Open for Innovation

KNIME

Dear Workshop Participant:

In order to facilitate a smooth KNIME Workshop experience, please follow the instructions below:

1) Go to www.knime.org and download the special distribution of KNIME provided for this workshop (link below). This special package provides the latest version of the KNIME Analytics Platform with all required extensions pre-installed. Additionally, it already contains the KNIME Workflows and data files that we will use in the workshop. The file is large (>1gb) so please do this well before the workshop begins.

   http://tech.knime.org/forum/knime-users/strasbourg-summer-school-2014

2) Unpack the archive file (.zip, .dmg, or .tar.gz) to a local directory on your computer.

3) Feel free to post any questions you may have prior to the workshop in the linked forum thread.

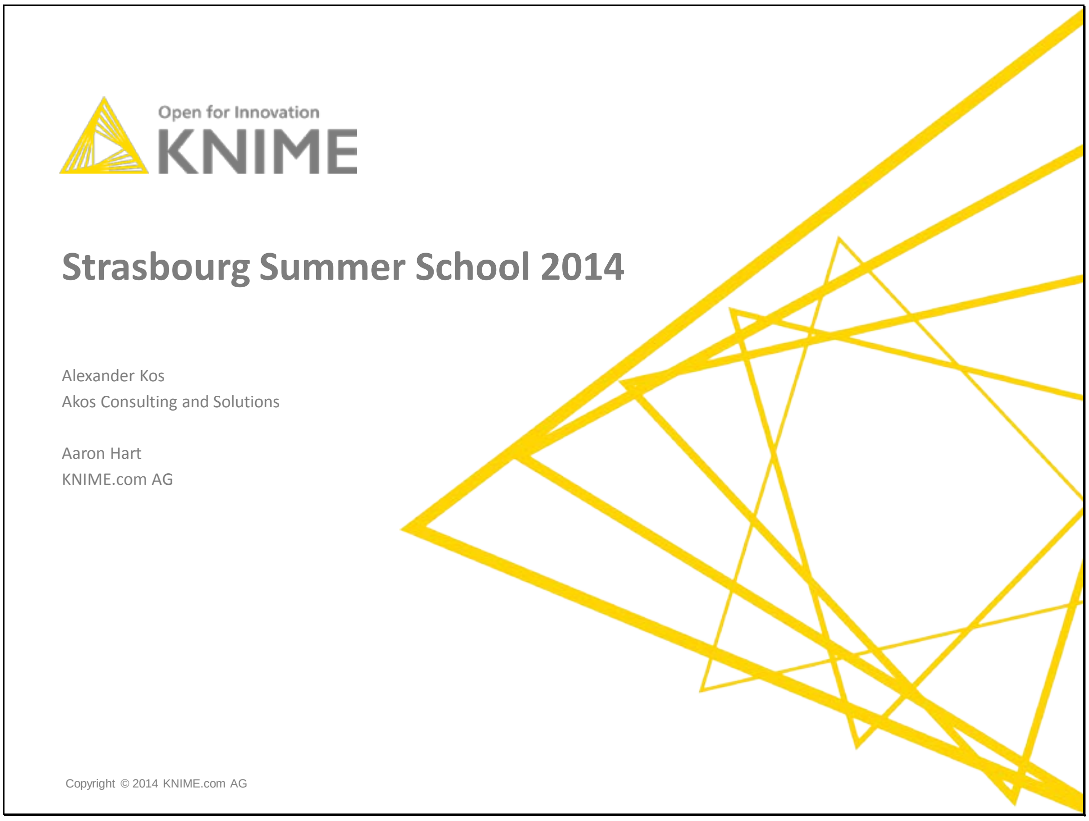
4) If you want to learn more about KNIME before the workshop, consider having a look at our youtube channel for guides to getting started and many other topics.

   http://www.youtube.com/knimetv

We look forward to seeing you all at the workshop!

Best regards,

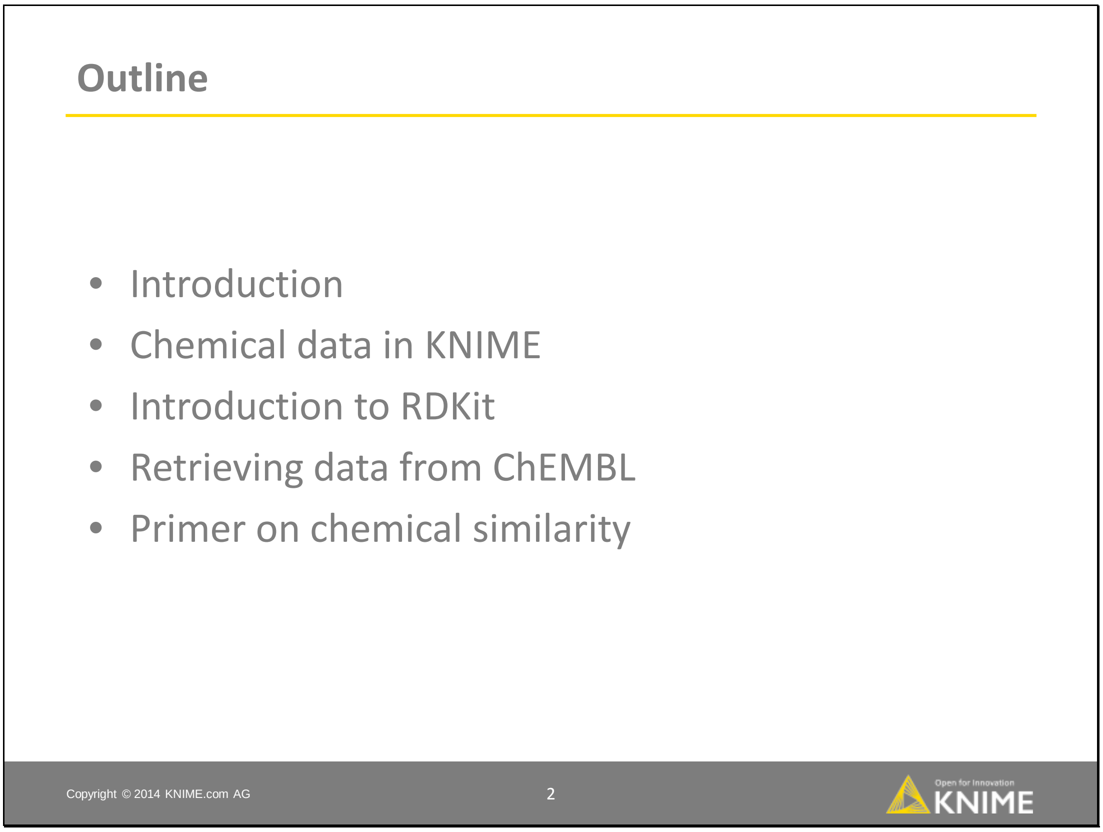
The KNIME Team



Strasbourg Summer School 2014

Alexander Kos
Akos Consulting and Solutions

Aaron Hart
KNIME.com AG

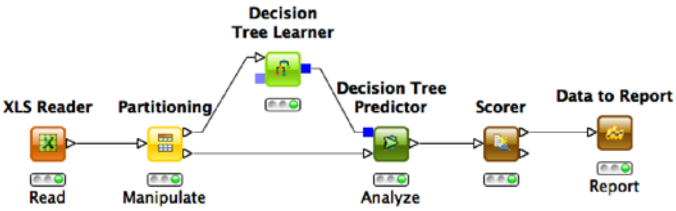


Outline

- Introduction
- Chemical data in KNIME
- Introduction to RDKit
- Retrieving data from ChEMBL
- Primer on chemical similarity

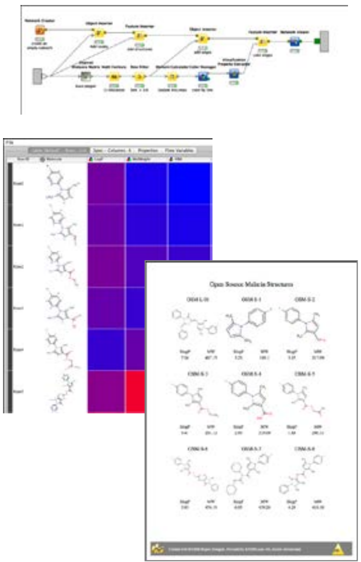
What is KNIME?

- Konstanz Information Miner
- Graphical programming tool
- Open Source, and frequently extended
- Broadly Supported by the cheminfo community

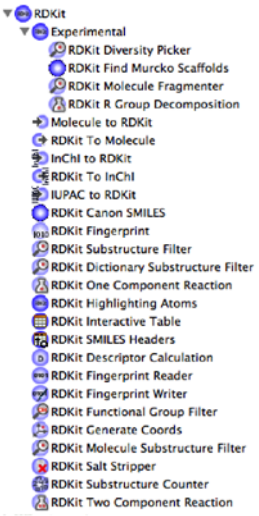


Why use KNIME?

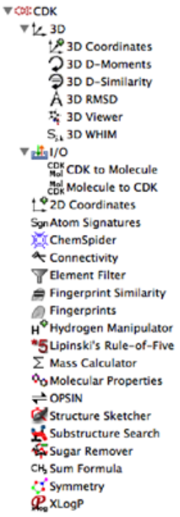
- It’s Free and Open Source
  - Fully functional, not crippleware!
  - An easy way to articulate complex processes - just annotate and share your workflow
- Integrate data from many potential sources (files, databases, web services)
- Mix and mash Commercial and Open Source tools



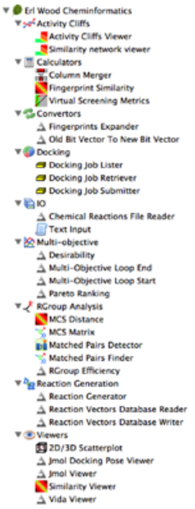
Selected Open Source extensions



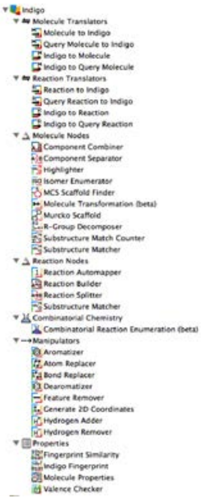
- RDKit
  - Experimental
    - RDKit Diversity Picker
    - RDKit Find Murcko Scaffolds
    - RDKit Molecule Fragmenter
    - RDKit R Group Decomposition
  - Molecule to RDKit
  - RDKit To Molecule
  - InChI to RDKit
  - RDKit To InChI
  - IUPAC to RDKit
  - RDKit Canon SMILES
  - RDKit Fingerprint
  - RDKit Substructure Filter
  - RDKit Dictionary Substructure Filter
  - RDKit One Component Reaction
  - RDKit Highlighting Atoms
  - RDKit Interactive Table
  - RDKit SMILES Headers
  - RDKit Descriptor Calculation
  - RDKit Fingerprint Reader
  - RDKit Fingerprint Writer
  - RDKit Functional Group Filter
  - RDKit Generate Coords
  - RDKit Molecule Substructure Filter
  - RDKit Salt Stripper
  - RDKit Substructure Counter
  - RDKit Two Component Reaction



- CDK
  - 3D
    - 3D Coordinates
    - 3D D-Moments
    - 3D D-Similarity
    - 3D RMSD
    - 3D Viewer
    - 3D WHIM
  - I/O
    - CDK to Molecule
    - Molecule to CDK
  - 2D Coordinates
  - Atom Signatures
  - ChemSpider
  - Connectivity
  - Element Filter
  - Fingerprint Similarity
  - Fingerprints
  - Hydrogen Manipulator
  - Lipinski's Rule-of-Five
  - Mass Calculator
  - Molecular Properties
  - OPSIN
  - Structure Sketcher
  - Substructure Search
  - Sugar Remover
  - Sum Formula
  - Symmetry
  - XLogP

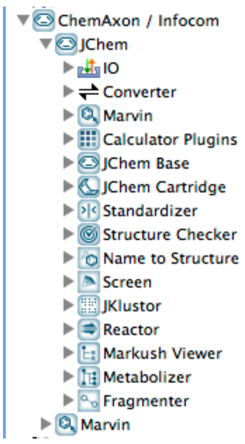


- Erl Wood Cheminformatics
  - Activity Cliffs
    - Activity Cliffs Viewer
    - Similarity network viewer
  - Calculators
    - Column Merger
    - Fingerprint Similarity
    - Virtual Screening Metrics
  - Convertors
    - Fingerprints Expander
    - Old Bit Vector To New Bit Vector
  - Docking
    - Docking Job Lister
    - Docking Job Retriever
    - Docking Job Submitter
  - IO
    - Chemical Reactions File Reader
    - Text Input
  - Multi-objective
    - Desirability
    - Multi-Objective Loop End
    - Multi-Objective Loop Start
    - Pareto Ranking
  - RGroup Analysis
    - MCS Distance
    - MCS Matrix
    - Matched Pairs Detector
    - Matched Pairs Finder
    - RGroup Efficiency
  - Reaction Generation
    - Reaction Generator
    - Reaction Vectors Database Reader
    - Reaction Vectors Database Writer
  - Viewers
    - 2D/3D Scatterplot
    - Jmol Docking Pose Viewer
    - Jmol Viewer
    - Similarity Viewer
    - Vida Viewer

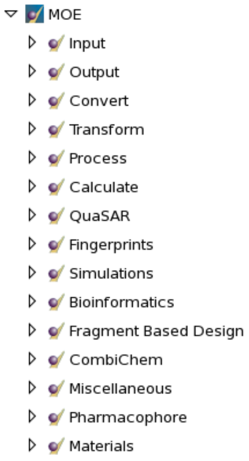


- Indigo
  - Molecule Translators
    - Molecule to Indigo
    - Query Molecule to Indigo
    - Indigo to Molecule
    - Indigo to Query Molecule
  - Reaction Translators
    - Reaction to Indigo
    - Query Reaction to Indigo
    - Indigo to Reaction
    - Indigo to Query Reaction
  - Molecule Nodes
    - Component Combiner
    - Component Separator
    - Highlighter
    - Isomer Enumerator
    - MCS Scaffold Finder
    - Molecule Transformation (beta)
    - Murcko Scaffold
    - R-Group Decomposer
    - Substructure Match Counter
    - Substructure Matcher
  - Reaction Nodes
    - Reaction Automapper
    - Reaction Builder
    - Reaction Splitter
    - Substructure Matcher
  - Combinatorial Chemistry
    - Combinatorial Reaction Enumeration (beta)
  - Manipulators
    - Aromatizer
    - Atom Replacer
    - Bond Replacer
    - Dearomatizer
    - Feature Remover
    - Generate 2D Coordinates
    - Hydrogen Adder
    - Hydrogen Remover
  - Properties
    - Fingerprint Similarity
    - Indigo Fingerprint
    - Molecule Properties
    - Valence Checker

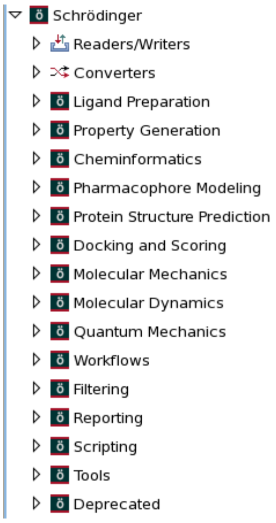
Selected commercial extensions



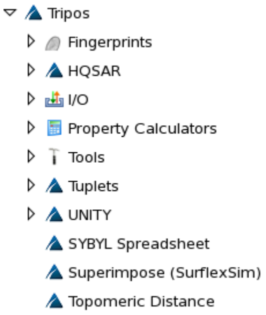
- ChemAxon / Infocom
  - JChem
    - IO
    - Converter
    - Marvin
    - Calculator Plugins
    - JChem Base
    - JChem Cartridge
    - Standardizer
    - Structure Checker
    - Name to Structure
    - Screen
    - JKlustor
    - Reactor
    - Markush Viewer
    - Metabolizer
    - Fragmenter
  - Marvin



- MOE
  - Input
  - Output
  - Convert
  - Transform
  - Process
  - Calculate
  - QuaSAR
  - Fingerprints
  - Simulations
  - Bioinformatics
  - Fragment Based Design
  - CombiChem
  - Miscellaneous
  - Pharmacophore
  - Materials



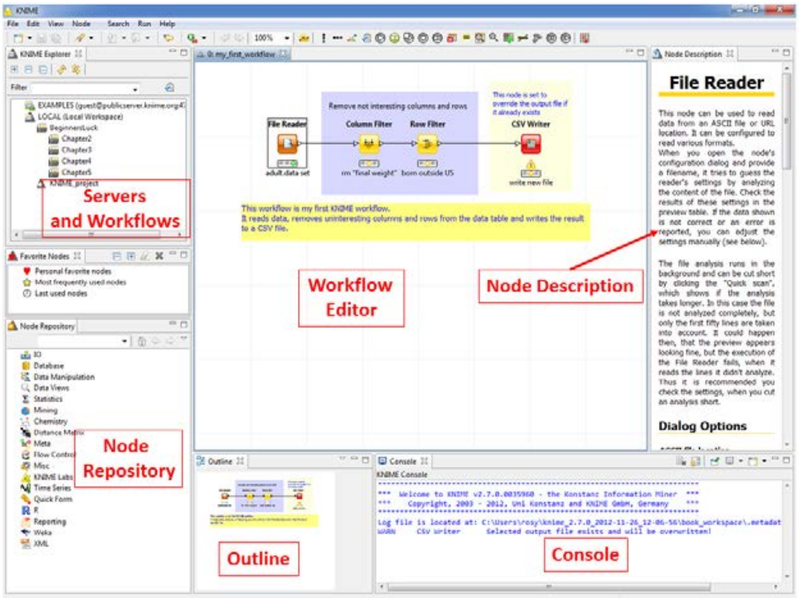
- Schrödinger
  - Readers/Writers
  - Converters
  - Ligand Preparation
  - Property Generation
  - Cheminformatics
  - Pharmacophore Modeling
  - Protein Structure Prediction
  - Docking and Scoring
  - Molecular Mechanics
  - Molecular Dynamics
  - Quantum Mechanics
  - Workflows
  - Filtering
  - Reporting
  - Scripting
  - Tools
  - Deprecated



- Tripos
  - Fingerprints
  - HQSAR
  - I/O
  - Property Calculators
  - Tools
  - Tuplets
  - UNITY
  - SYBYL Spreadsheet
  - Superimpose (SurflexSim)
  - Topomeric Distance

The KNIME Analytics Platform

The KNIME Workbench



This workflow is my first KNIME workflow.
It reads data, removes uninteresting columns and rows from the data table and writes the result to a CSV file.

Node Description

File Reader

This node can be used to read data from an ASCII file or URL location. It can be configured to read various formats.
When you open the node's configuration dialog and provide a filename, it tries to guess the reader's settings by analyzing the content of the file. Check the results of these settings in the preview table. If the data shown is not correct or an error is reported, you can adjust the settings manually (see below).

The file analysis runs in the background and can be cut short by clicking the "Quick scan", which shows if the analysis takes longer. In this case the file is not analyzed completely, but only the first fifty lines are taken into account. It could happen then, that the preview appears looking fine, but the execution of the File Reader fails, when it reads the lines it didn't analyze. Thus it is recommended you check the settings, when you cut an analysis short.

Dialog Options

Console

KNIME Console

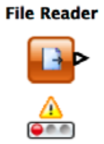
```
**************************************************************************
***  Welcome to KNIME v2.7.0.0035960 - the Konstanz Information Miner  ***
***    Copyright, 2003 - 2012, Uni Konstanz and KNIME GmbH, Germany    ***
**************************************************************************
Log file is located at: C:\Users\rosy\knime_2.7.0_2012-11-26_12-06-56\book_workspace\.metadat
WARN    CSV Writer      Selected output file exists and will be overwritten!
```

Nodes in KNIME

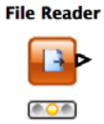
- May be provided by us, commercial partners, or the KNIME Open Source Community
  - Nodes ma be used to read, manipulate or write data.
  - KNIME's philosophy is to lean towards "1 node per task"
  - Mixing and matching nodes from many providers is seamless.

More on nodes…

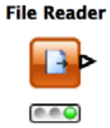
A node can have 3 states:



Idle:
The node is not yet configured and can not be executed with it's current settings.



Configured:
The node has been set up correctly, and may be executed at any time

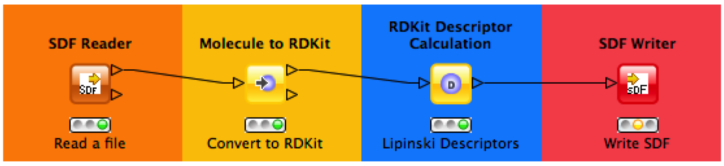


Executed:
The node has been successfully executed. Results may be viewed and used in downstream nodes.

And what is a workflow?

In KNIME, a workflow is just a few nodes strung together to complete a task…

Step 1: Read data file
Step 2: Manipulate types
Step 3: Analyze
Step 4: Export Results



Hotkeys

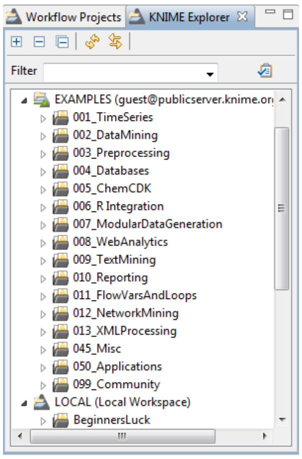
Task	Hotkey	Description
Node Configuration	F6	opens the configuration dialog of a node
Node Execution	F7	executes selected nodes
	Shift + F7	executes all configured nodes
	Shift + F10	executes configured nodes and opens all views
	F9	cancels selected running nodes
	Shift + F9	cancels all running nodes
Move Nodes and Annotations	Ctrl + Shift + Arrow	moves a selected node in the workflow editor
	Ctrl + Shift + PgUp/PgDown	Moves the selected up or down in z order
	F8	resets selected nodes
Workflow Operations	Ctrl + S	Saves the workflow
	Ctrl + Shift + S	Saves all open workflows
	Ctrl + Shift + W	Closes all open workflows
Meta-node	Shift + F12	Opens meta-node wizard

The Public Example Server

The KNIME Example Server provides access to many explanatory workflows.

In the KNIME Explorer panel:

- –right click the public server
- -select “Login”
- -No login credentials required



- EXAMPLES (guest@publicserver.knime.or
  - 001_TimeSeries
  - 002_DataMining
  - 003_Preprocessing
  - 004_Databases
  - 005_ChemCDK
  - 006_R Integration
  - 007_ModularDataGeneration
  - 008_WebAnalytics
  - 009_TextMining
  - 010_Reporting
  - 011_FlowVarsAndLoops
  - 012_NetworkMining
  - 013_XMLProcessing
  - 045_Misc
  - 050_Applications
  - 099_Community
- LOCAL (Local Workspace)
  - BeginnersLuck

Exercise 1

Launch KNIME.

Open a workflow by double clicking on it.

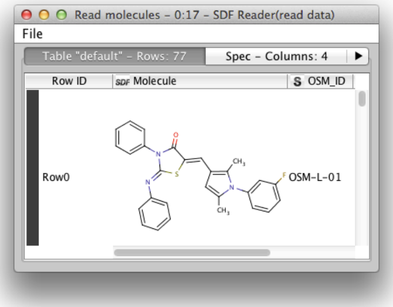
Right click on a node, and look view the resulting table (bottom option in the context menu)

Chemistry data in KNIME reading, writing and types

Overview of types in KNIME

- Basic KNIME types
  - string, integer, double

- KNIME core chemistry types:
  - smiles, sdf, mol, mol2
  - Structures in these formats can be rendered in KNIME tables



Read molecules – 0:17 – SDF Reader(read data)

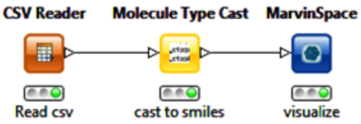
File

Table "default" – Rows: 77 | Spec – Columns: 4

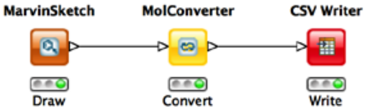
Row ID	SDF Molecule	S OSM_ID
Row0		OSM-L-01

Nodes for type manipulation

- Molecule Type Cast
  - Casts any string as a chemical type (ie. It tells KNIME “This is a smiles string”)
  - Useful when reading data form a csv file or database.



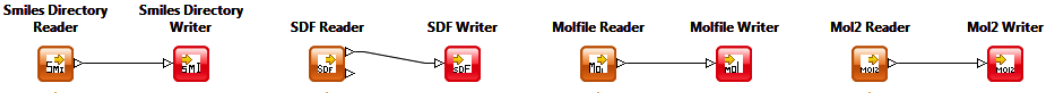
- Marvin MolConverter
  - Provided by Chemaxon/Infocom
  - Translates seamlessly between types (smiles ⇔ sdf ⇔ mrv)



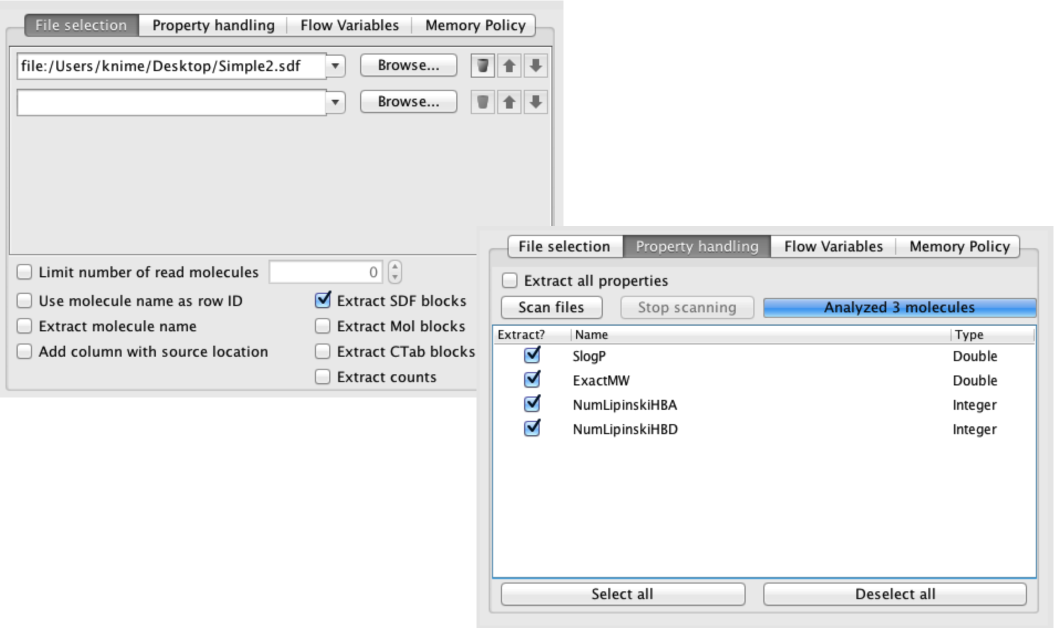
Nodes for reading and writing files

Reader and writers provided for:

- sdf, smiles, mol, mol2



A bit more about reading sd files



File selection | Property handling | Flow Variables | Memory Policy

file:/Users/knime/Desktop/Simple2.sdf ▾ Browse...

▾ Browse...

- ☐ Limit number of read molecules 0
- ☐ Use molecule name as row ID
- ☐ Extract molecule name
- ☐ Add column with source location
- ☑ Extract SDF blocks
- ☐ Extract Mol blocks
- ☐ Extract CTab blocks
- ☐ Extract counts

File selection | Property handling | Flow Variables | Memory Policy

☐ Extract all properties

Scan files | Stop scanning | Analyzed 3 molecules

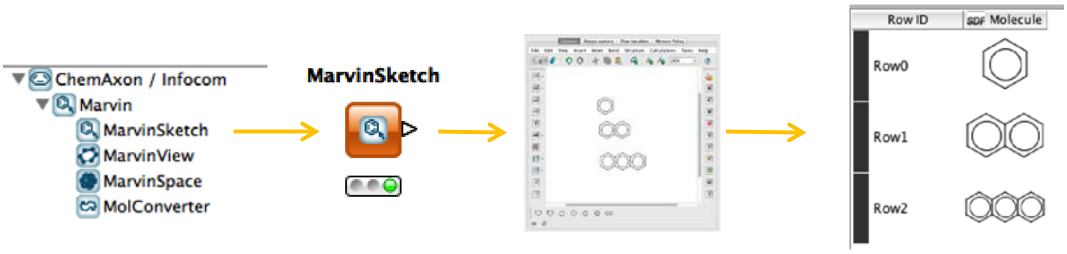
Extract?	Name	Type
☑	SlogP	Double
☑	ExactMW	Double
☑	NumLipinskiHBA	Integer
☑	NumLipinskiHBD	Integer

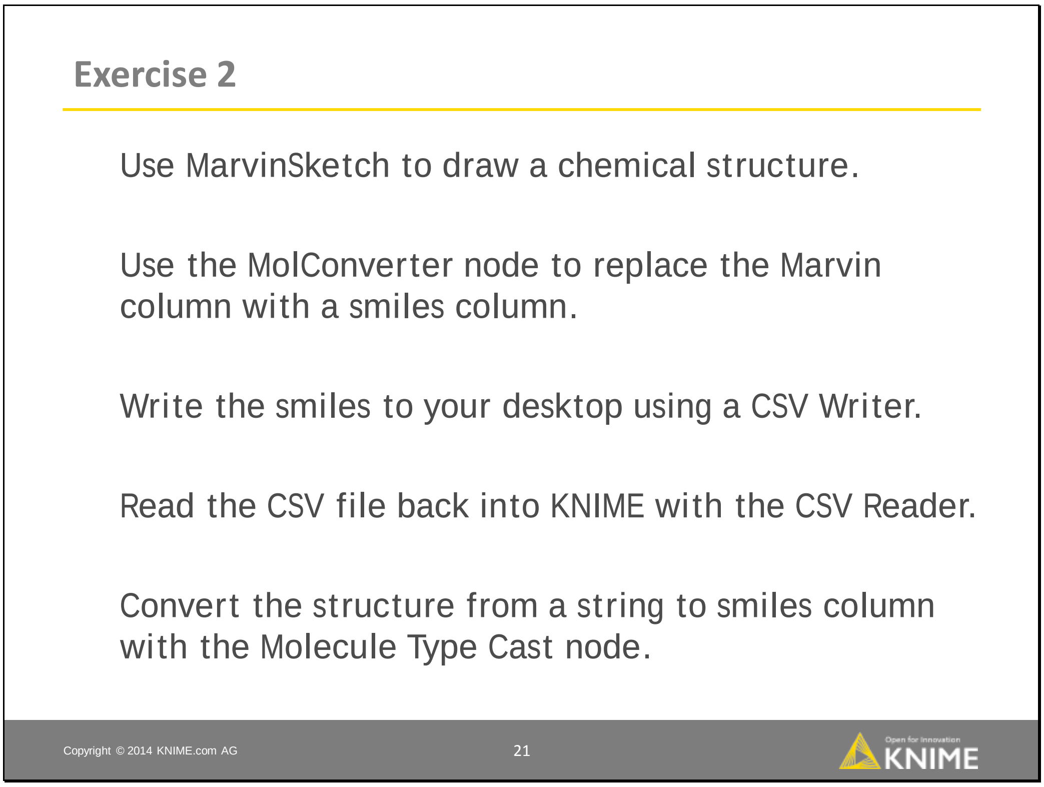
Select all | Deselect all

Sketching chemical structures – use Marvin

MarvinSketch

- Provided by Chemaxon/Infocom
- Sketch structures in the configuration dialog
- Execute node to inject structures into workflow





Exercise 2

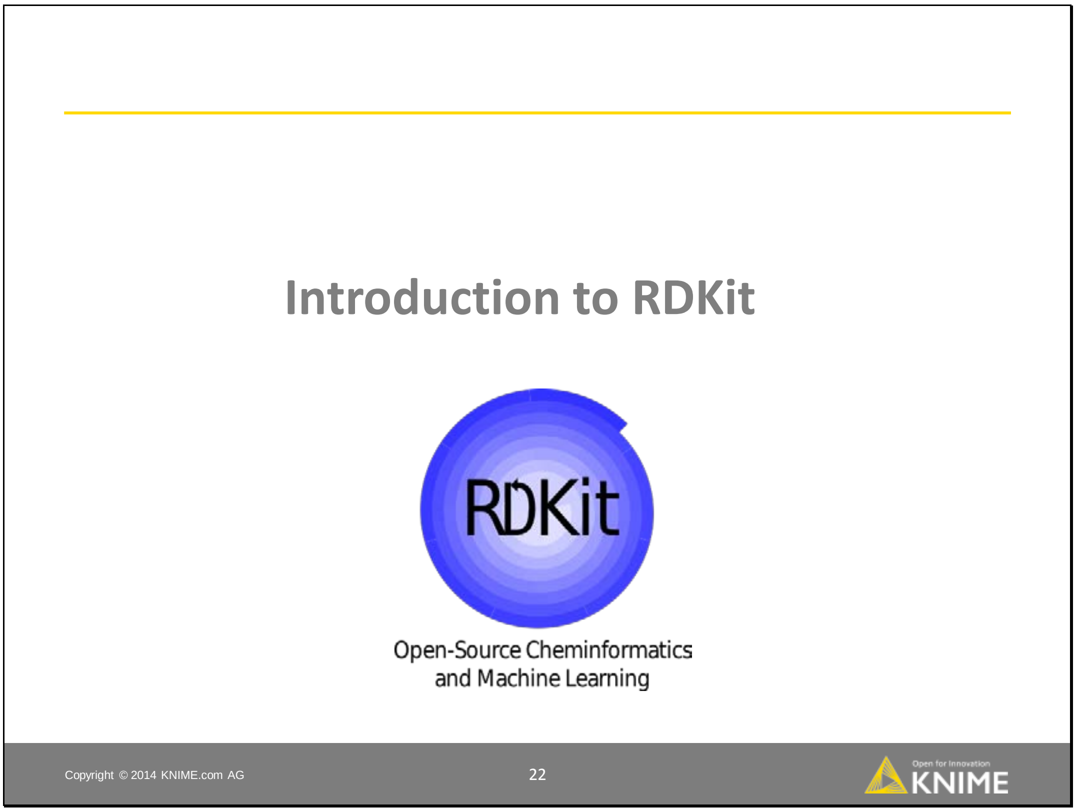
Use MarvinSketch to draw a chemical structure.

Use the MolConverter node to replace the Marvin column with a smiles column.

Write the smiles to your desktop using a CSV Writer.

Read the CSV file back into KNIME with the CSV Reader.

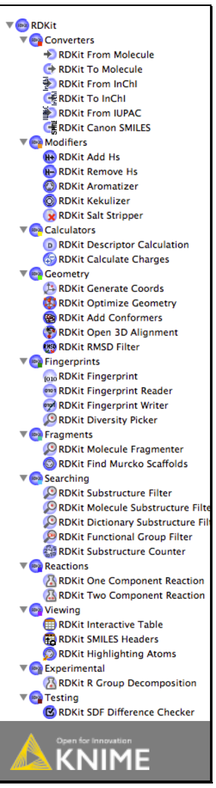
Convert the structure from a string to smiles column with the Molecule Type Cast node.



Introduction to RDKit

What is RDKit?

- Open source cheminfo library in c++
- Wrappers for KNIME maintained by the open source community
- Useful for:
  - Descriptor calculation
  - Cleaning structures
  - InChi conversion
  - Standardizing smiles
  - Fingerprints
  - Scaffolds/substructures
  - Reaction simulation
  - and more…

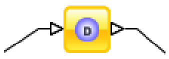


- RDKit
  - Converters
    - RDKit From Molecule
    - RDKit To Molecule
    - RDKit From InChI
    - RDKit To InChI
    - RDKit From IUPAC
    - RDKit Canon SMILES
  - Modifiers
    - RDKit Add Hs
    - RDKit Remove Hs
    - RDKit Aromatizer
    - RDKit Kekulizer
    - RDKit Salt Stripper
  - Calculators
    - RDKit Descriptor Calculation
    - RDKit Calculate Charges
  - Geometry
    - RDKit Generate Coords
    - RDKit Optimize Geometry
    - RDKit Add Conformers
    - RDKit Open 3D Alignment
    - RDKit RMSD Filter
  - Fingerprints
    - RDKit Fingerprint
    - RDKit Fingerprint Reader
    - RDKit Fingerprint Writer
    - RDKit Diversity Picker
  - Fragments
    - RDKit Molecule Fragmenter
    - RDKit Find Murcko Scaffolds
  - Searching
    - RDKit Substructure Filter
    - RDKit Molecule Substructure Filte
    - RDKit Dictionary Substructure Fil
    - RDKit Functional Group Filter
    - RDKit Substructure Counter
  - Reactions
    - RDKit One Component Reaction
    - RDKit Two Component Reaction
  - Viewing
    - RDKit Interactive Table
    - RDKit SMILES Headers
    - RDKit Highlighting Atoms
  - Experimental
    - RDKit R Group Decomposition
  - Testing
    - RDKit SDF Difference Checker

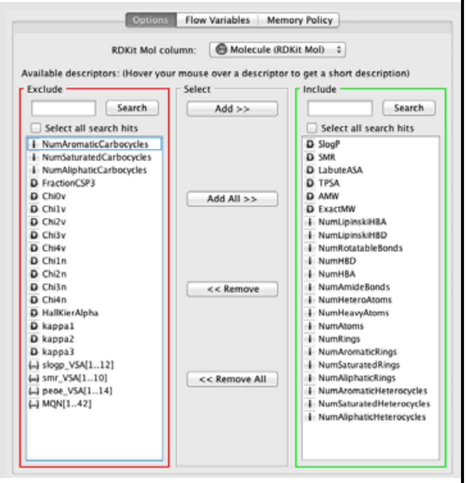
Popular RDKit nodes: Descriptor Calculator

- Input smiles/sdf
- Can predict/calculate many descriptors

(e.g. logP, MW, HBA, HBD)



RDKit Descriptor Calculation



RDKit Mol column: Molecule (RDKit Mol)

Available descriptors: (Hover your mouse over a descriptor to get a short description)

Exclude

□ Select all search hits

- NumAromaticCarbocycles
- NumSaturatedCarbocycles
- NumAliphaticCarbocycles
- FractionCSP3
- Chi0v
- Chi1v
- Chi2v
- Chi3v
- Chi4v
- Chi1n
- Chi2n
- Chi3n
- Chi4n
- HallKierAlpha
- kappa1
- kappa2
- kappa3
- slogp_VSA[1..12]
- smr_VSA[1..10]
- peoe_VSA[1..14]
- MQN[1..42]

Select

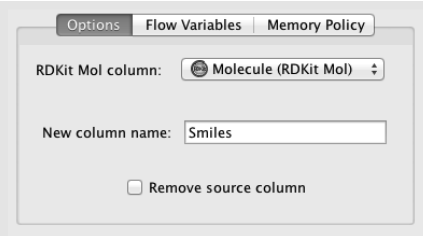
Include

□ Select all search hits

- SlogP
- SMR
- LabuteASA
- TPSA
- AMW
- ExactMW
- NumLipinskiHBA
- NumLipinskiHBD
- NumRotatableBonds
- NumHBD
- NumHBA
- NumAmideBonds
- NumHeteroAtoms
- NumHeavyAtoms
- NumAtoms
- NumRings
- NumAromaticRings
- NumSaturatedRings
- NumAliphaticRings
- NumAromaticHeterocycles
- NumSaturatedHeterocycles
- NumAliphaticHeterocycles

Popular RDKit nodes: Cannon Smiles

- Input smiles/sdf
- Calculate smiles such that one string is produced per molecule. Useful for resolving duplicate structures in data from several sources



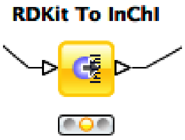
RDKit Mol column: Molecule (RDKit Mol)

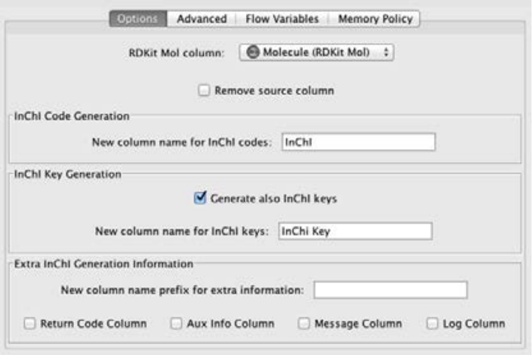
New column name: Smiles

□ Remove source column

Popular RDKit nodes: InChi Keys

- Input smiles/sdf
- Generate InChi keys and codes. Useful when searching for information about your structure, without revealing it.





RDKit Mol column: Molecule (RDKit Mol)

□ Remove source column

InChI Code Generation

New column name for InChI codes: InChI

InChI Key Generation

☑ Generate also InChI keys

New column name for InChI keys: InChi Key

Extra InChI Generation Information

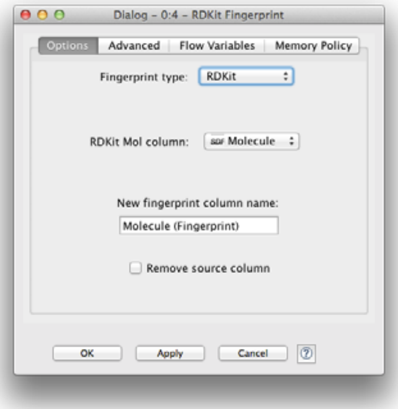
New column name prefix for extra information:

□ Return Code Column □ Aux Info Column □ Message Column □ Log Column

Popular RDKit nodes: RDKit Fingerprint

- Generate chemical hashed fingerprints using a variety of methods. May be later used for building activity models, diversity picking, or clustering.

RDKit Fingerprint



Fingerprint type: RDKit

RDKit Mol column: Molecule

New fingerprint column name: Molecule (Fingerprint)

☐ Remove source column

Exercise 3

Read the OSM Structures by dragging the SD File from the explorer to your workflow.

Calculate the Molecular Weight of the structures using RDKit Descriptors

Standardize the smiles using Cannon Smiles.

Generate InChi Keys and Codes for the OSM Structures

Write the structures to your desktop using the SDF Writer. Include the mw as a property in the output file.

Accessing ChEMBL

What is ChEMBL?

A public database of bioactive druglike compounds

~1.3 mio compounds

~ 9k targets

~12 mio bioactivitities

Provided by the European Bioinformatics Institute

Accessible online at www.ebi.ac.uk/chembl

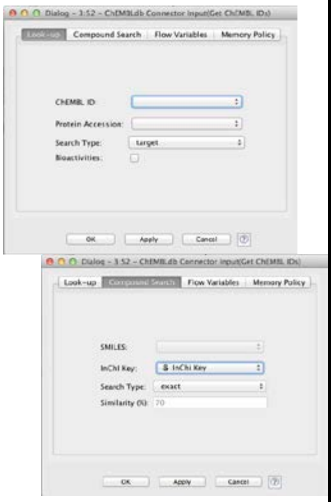
or via EBI provided KNIME nodes…

New Node: ChEMBLdb Connector

Access data in ChEMBL via a web service call
(internet access required)

Lookup by ChEMBLID or InChi Key
Retrieve structure and bioactivity data

Compound search using smiles
exact, similarity, or substructure

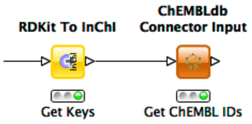


Dialog - 3:52 - ChEMBLdb Connector Input(Get ChEMBL IDs)

CHEMBL ID:
Protein Accession:
Search Type: target
Bioactivities: □

Dialog - 3:52 - ChEMBLdb Connector Input(Get ChEMBL IDs)

SMILES:
InChi Key: InChi Key
Search Type: exact
Similarity (%): 70



Exercise 4

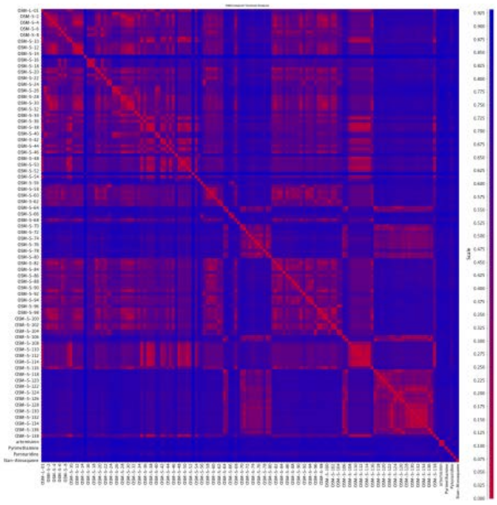
Read the OSM Structures by dragging the SD File from the explorer to your workflow.

Generate InChi Keys for the OSM Structures. Use these to execute an exact search in ChEMBL

Use GroupBy on chemblid to find unique entries.

Search for bioactivities for these compounds and filter to keep activities against target CHEMBL364 (Plasmodium falciparum) that are IC50 values and reported in “nM”. Hint: use 3 Row Filter nodes.

Chemical Similarity



Chemical Similarity Overview

Two methods commonly used for evaluating similarity:

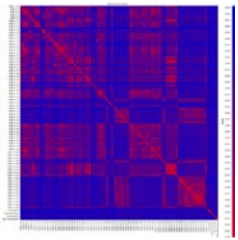
1. Chemical hashed fingerprints

   The more similar the fingerprint, the more similar the molecule

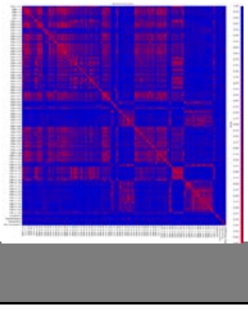
2. Maximum common substructure

   The larger the MCS, the more similar the molecules

CHF



MCS

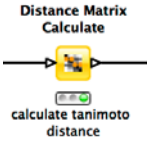


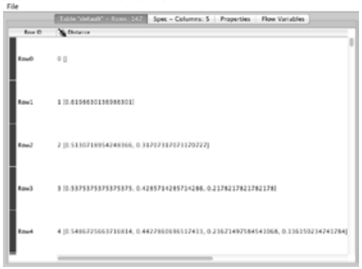
New node: Distance Matrix Calculate

Creates a special column with pairwise similarities

(n^2-n)/2 distances = Heavy computation for large libraries

Several methods
- Tanimoto for fingerprint comparison
  - (number of shared bits/number of bits)
- Euclidean for normalized real numbers





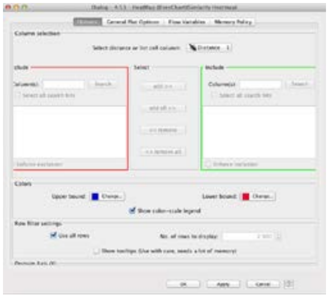
Row ID	Distance
Row0	0 []
Row1	1 [0.6198630136986301]
Row2	2 [0.5130718954248366, 0.3170731707317072]
Row3	3 [0.5375375375375375, 0.4285714285714286, 0.2178217821782178]
Row4	4 [0.5486725663716814, 0.4427860696517413, 0.23671497584541068, 0.136150234741784]

New node: JFreeChart Heatmap

Provides a nice quick view of compound similarity.

Works directly with distance matrices

Uses RowIDs for axis labels



Select distance or list cell column: Distance

Upper bound: 🟦 Change...

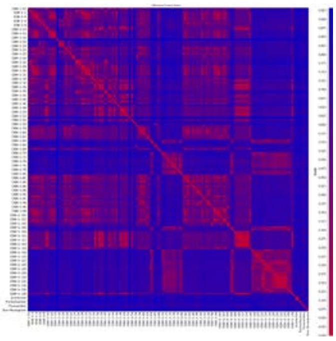
Lower bound: 🟥 Change...

☑ Show color-scale legend

☑ Use all rows

No. of rows to display:

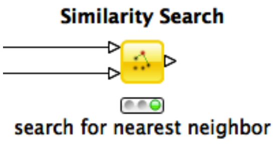
□ Show tooltips (Use with care, needs a lot of memory)

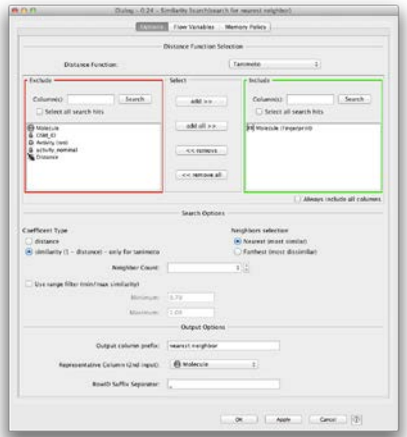


New node: Similarity Search

Query molecules in top port, corpus to search in bottom port

Find n nearest neighbors, possible within a similarity threshold (e.g. 0.7-1.0)




Dialog - 0:24 - Similarity Search(search for nearest neighbor)

Options | Flow Variables | Memory Policy

Distance Function Selection

Distance Function: Tanimoto

Exclude
Column(s): Search
□ Select all search hits
Molecule
OSM_ID
Activity (raw)
activity_nominal
Distance

Select
add >>
add all >>
<< remove
<< remove all

Include
Column(s): Search
□ Select all search hits
Molecule (fingerprint)

□ Always include all columns

Search Options

Coefficient Type
○ distance
● similarity (1 – distance) - only for tanimoto

Neighbors selection
● Nearest (most similar)
○ Farthest (most dissimilar)

Neighbor Count: 1

□ Use range filter (min/max similarity)
Minimum: 0.70
Maximum: 1.00

Output Options

Output column prefix: nearest neighbor
Representative Column (2nd input): Molecule
RowID Suffix Separator: _

Exercise 5

Read the sd files, import the OSM_ID for each structure.

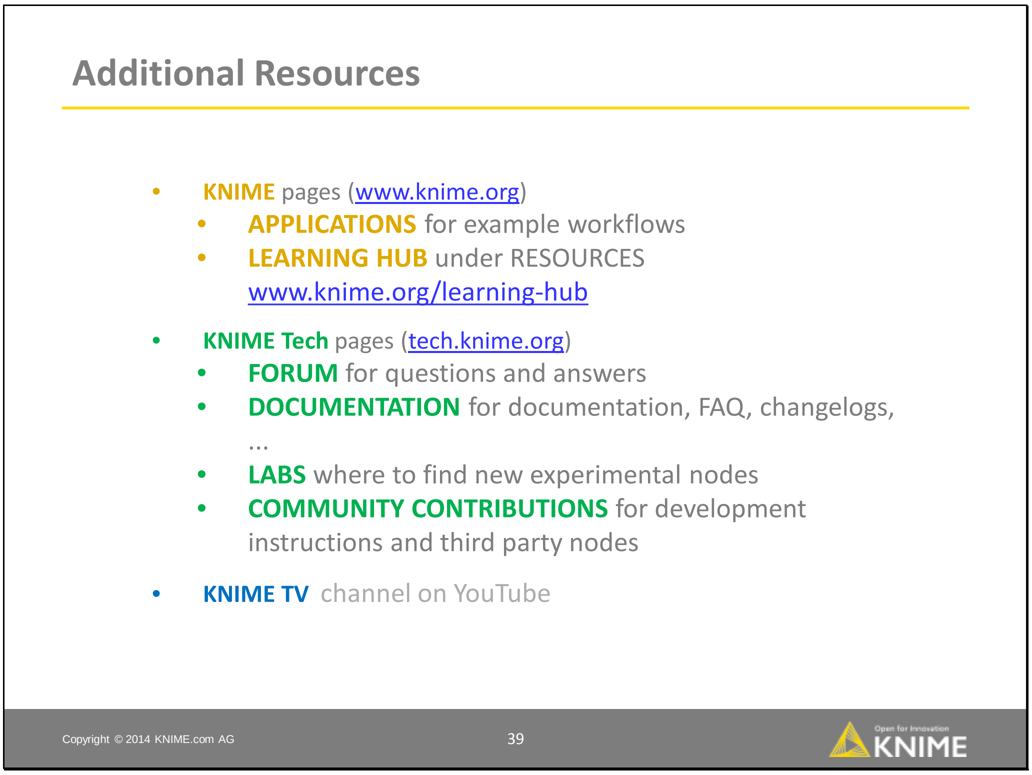
Use a RowID node to label each row by its OSM_ID

Use RDKit to generate fingerprints for your structures

Create a Distance Matrix using Tanimoto similarity for the structures.

Create a similarity heatmap of our library. Hint: Use JfreeChart Heatmap

Use a Row Splitter to take the first structure and search for the most similar entry out of the remaining structures.



Additional Resources

- **KNIME** pages (www.knime.org)
  - **APPLICATIONS** for example workflows
  - **LEARNING HUB** under RESOURCES www.knime.org/learning-hub
- **KNIME Tech** pages (tech.knime.org)
  - **FORUM** for questions and answers
  - **DOCUMENTATION** for documentation, FAQ, changelogs, …
  - **LABS** where to find new experimental nodes
  - **COMMUNITY CONTRIBUTIONS** for development instructions and third party nodes
- **KNIME TV** channel on YouTube