RDKit Training: Learn Cheminformatics & Molecular Modeling for Drug Discovery

Master RDKit for cheminformatics and drug discovery! Learn molecular modeling, QSAR, fingerprinting, and AI-driven analysis with hands-on training.

View Fee

Click Here for Other cheminformatics Training

Customized Cheminformatics Modules

Module 1: Introduction to RDKit and Cheminformatics

1.1 Overview of RDKit

- What is RDKit? Importance in cheminformatics and drug discovery
- Key features and capabilities
- Comparison with other cheminformatics tools

1.2 Installation and Setup

- Installing RDKit on Windows, Linux, and macOS
- Setting up RDKit with Python
- Integrating RDKit with Jupyter Notebook

1.3 Understanding Chemical Data Formats

- Introduction to molecular representations
- SMILES, InChI, MOL, and SDF file formats
- Conversion between different chemical formats

1.4 First Steps with RDKit

- Loading and visualizing molecules
- Basic molecular operations (atom and bond manipulations)
- Generating 2D molecular structures

1.5 Hands-on Exercises

- Write Python scripts to load and visualize molecular structures
- Convert SMILES strings into different formats
- Basic molecular manipulation exercises

Module 2: Molecular Representations and Structure Handling in RDKit

2.1 Understanding Molecular Objects in RDKit

- Defining molecules as RDKit objects using `Chem.MolFromSmiles()` and `Chem.MolFromMolFile()`
- Accessing molecular properties: Atoms, Bonds, and Rings
- Printing molecular information using `MolToMolBlock()` and `MolToSmiles()`
- Identifying explicit and implicit hydrogens in molecular structures

2.2 Working with Different Molecular Representations

- Understanding molecular representations:
  - **SMILES:** Simplified Molecular Input Line Entry System
  - **InChI:** International Chemical Identifier
  - **MOL & SDF:** Structural Data File formats
- Reading and parsing chemical structures:
  - Converting between SMILES, InChI, and MOL using `Chem.MolToSmiles()`, `Chem.MolToInchi()`
  - Using `Chem.AddHs()` and `Chem.RemoveHs()` for hydrogen handling
  - Checking molecular validity using `rdkit.Chem.rdmolops.SanitizeMol()`

2.3 Loading and Manipulating Molecular Structures

- Reading molecular structures from:
  - SMILES strings: `mol = Chem.MolFromSmiles("CCO")`
  - MOL files: `mol = Chem.MolFromMolFile("molecule.mol")`
  - SDF files: Using `SDMolSupplier` to iterate through multi-molecule files
- Modifying molecular structures:
  - Accessing atoms and bonds: `mol.GetAtoms()` and `mol.GetBonds()`
  - Changing atomic properties (valency, hybridization)
  - Creating substructures and fragments using `rdkit.Chem.FragmentOnBonds()`

2.4 Generating and Optimizing 3D Molecular Structures

- Embedding molecules in 3D space:
  - Using `rdkit.Chem.AllChem.EmbedMolecule()` for 3D coordinates
  - Energy minimization using

  `rdkit.Chem.AllChem.UFFOptimizeMolecule()`
  - Generating multiple conformations with `rdkit.Chem.AllChem.EmbedMultipleConfs()`
- Analyzing 3D molecular structures:
  - Measuring bond lengths and angles
  - Comparing RMSD between conformers

2.5 Saving and Exporting Molecular Structures

- Exporting molecules to:
  - SMILES format: `smiles = Chem.MolToSmiles(mol)`
  - MOL file: `Chem.MolToMolFile(mol, "output.mol")`
  - SDF file: Writing multiple molecules using `SDWriter`
- Rendering molecules as images:
  - Using `rdkit.Chem.Draw.MolToImage()` for static visualization
  - Generating 2D depictions with `rdkit.Chem.Draw.MolDraw2D`

2.6 Hands-on Exercises

- Convert a list of SMILES strings to MOL format and save them
- Generate 3D structures for a set of molecules and optimize their energy
- Extract substructures from a given set of molecular structures
- Compare different representations of a molecule and analyze their differences
- Write a Python script to iterate through an SDF file and retrieve molecular properties

Module 3: Molecular Fingerprints, Descriptors, and Similarity Searching in RDKit

3.1 Introduction to Molecular Fingerprints

- Understanding molecular fingerprints and their significance
- Types of fingerprints in RDKit:
  - **Morgan (ECFP)** – Extended Connectivity Fingerprint
  - **MACCS** – Molecular ACCess System keys
  - **Atom-Pair & Topological Torsion** – Path-based fingerprints
  - **RDKit Fingerprint** – Default path-based fingerprint

3.2 Generating and Analyzing Molecular Fingerprints

- Creating fingerprints:
  - Generating Morgan fingerprints: `rdkit.Chem.AllChem.GetMorganFingerprintAsBitVect(mol, radius=2)`
  - Generating MACCS keys: `rdkit.Chem.rdMolDescriptors.GetMACCSKeysFingerprint(mol)`
  - Generating RDKit standard fingerprints:

    `rdkit.Chem.rdFingerprintGenerator.GetRDKitFPGenerator()`
- Understanding bit vectors and feature encoding
- Comparing different fingerprint methods for molecular similarity

3.3 Molecular Descriptor Calculation

- What are molecular descriptors? Importance in QSAR and cheminformatics
- Calculating physicochemical descriptors:
    - Molecular weight: `rdkit.Chem.Descriptors.MolWt(mol)`
    - LogP (lipophilicity): `rdkit.Chem.Crippen.MolLogP(mol)`
    - Topological polar surface area (TPSA): `rdkit.Chem.rdMolDescriptors.CalcTPSA(mol)`
    - Hydrogen bond donors and acceptors: `rdkit.Chem.rdMolDescriptors.CalcNumHBD(mol)` and `CalcNumHBA(mol)`
- Extracting multiple descriptors using `rdkit.Chem.Descriptors` module
- Storing descriptor data in Pandas DataFrames for analysis

3.4 Molecular Similarity Search

- Understanding similarity metrics:
    - Tanimoto similarity: `DataStructs.FingerprintSimilarity(fp1, fp2)`
    - Dice, Cosine, and Sokal similarity measures
- Performing similarity searches in large molecular databases
- Building a molecule similarity search function using RDKit
- Visualizing similar molecules using Matplotlib and RDKit drawing tools

3.5 Hands-on Exercises

- Generate and compare fingerprints for a set of molecules
- Calculate molecular descriptors for a dataset and analyze trends
- Perform a Tanimoto similarity search using a reference molecule
- Develop a Python script that ranks compounds by similarity to a given drug molecule

Module 4: Substructure Searching and Functional Group Identification in RDKit

4.1 Introduction to Substructure Searching

- Understanding molecular substructures and pattern matching
- Difference between exact structure matching and substructure matching
- Importance of SMARTS (SMiles ARbitrary Target Specification) notation

4.2 Performing Substructure Searches

- Using RDKit to identify substructures:
  - Checking if a molecule contains a substructure: `mol.HasSubstructMatch(submol)`
  - Finding multiple matches within a molecule: `mol.GetSubstructMatches(submol)`
  - Highlighting matched substructures in molecular visualization
- Creating SMARTS queries for complex molecular patterns
- Filtering large molecular datasets based on substructures

4.3 Identifying Functional Groups

- Understanding functional groups and their chemical properties
- Using RDKit to detect common functional groups:
  - Carboxyl (-COOH), Hydroxyl (-OH), Amino (-NH2), Ketones (C=O), etc.
  - Using `rdkit.Chem.Fragments` module for functional group identification
  - SMARTS patterns for detecting custom functional groups
- Counting occurrences of functional groups in a molecule
- Generating molecular fingerprints based on functional group presence

4.4 Filtering and Screening Molecules

- Creating molecular filters based on substructure presence
- Filtering databases based on:
  - Presence of reactive functional groups
  - Drug-likeness rules (Lipinski’s Rule of 5)
  - Structural complexity
- Automating substructure-based molecule selection using Pandas and RDKit

4.5 Hands-on Exercises

- Perform substructure searches in a molecular dataset
- Write a script to detect and count specific functional groups in molecules
- Filter a list of molecules based on drug-likeness criteria
- Visualize and highlight substructures in chemical compounds

Module 5: Chemical Reactions, Molecular Transformations, and Scaffold Analysis in RDKit

5.1 Introduction to Chemical Reactions in RDKit

- Understanding reaction representation in cheminformatics
- Defining chemical reactions using SMARTS patterns
- Basic reaction operations using `rdkit.Chem.rdChemReactions`

5.2 Defining and Applying Chemical Reactions

- Creating reaction templates using SMARTS:
    - Defining a reaction: `rdkit.Chem.rdChemReactions.ReactionFromSmarts()`
    - Identifying reactants, reagents, and products
    - Handling reaction specificity and stereochemistry
- Applying reactions to molecules:
    - Single-step transformations
    - Multi-step synthetic reaction planning

5.3 Scaffold Analysis and Molecular Core Extraction

- Understanding molecular scaffolds and their importance
- Extracting molecular cores using the Murcko Scaffold algorithm
- Analyzing core structures in a dataset of compounds

5.4 Functional Group Transformations

- Defining transformation rules using SMARTS
- Performing in-silico derivatization and retrosynthesis
- Automating functional group modifications for drug design

5.5 Hands-on Exercises

- Define and apply a SMARTS-based reaction to a dataset
- Extract Murcko scaffolds from a list of drug-like molecules
- Perform functional group modifications on a given molecule set
- Simulate multi-step reaction pathways

Module 6: Machine Learning with RDKit – QSAR Modeling and Predictive Analytics

6.1 Introduction to QSAR (Quantitative Structure-Activity Relationship)

- Understanding the concept of QSAR in cheminformatics
- Applications of QSAR modeling in drug discovery and toxicology
- Workflow for building a QSAR model

6.2 Extracting Molecular Features for Machine Learning

- Generating molecular descriptors using RDKit:
    - Physicochemical properties (LogP, MW, TPSA, etc.)
    - Structural features (hydrogen bond donors/acceptors, rotatable bonds)
    - Topological and 3D descriptors
- Generating molecular fingerprints as input features:

  - Morgan fingerprints (ECFP)
  - MACCS keys
  - RDKit fingerprint
- Converting molecular data into numerical format using Pandas and NumPy

6.3 Data Preprocessing for QSAR Modeling

- Handling missing and imbalanced data
- Feature selection techniques:
  - Correlation-based filtering
  - Principal Component Analysis (PCA)
- Scaling and normalization of molecular descriptors

6.4 Building Machine Learning Models for Molecular Property Prediction

- Introduction to machine learning algorithms for QSAR:
  - Linear Regression
  - Random Forest
  - Support Vector Machines (SVM)
  - Neural Networks
- Training a predictive model using Scikit-learn
- Cross-validation and performance evaluation (R^2, RMSE, MAE)

6.5 Molecular Property Prediction Using Pretrained Models

- Building predictive models for:
  - Drug-likeness
  - Toxicity prediction
  - Bioavailability screening
- Using machine learning for virtual screening of compound libraries

6.6 Hands-on Exercises

- Extract molecular descriptors and fingerprints from a dataset
- Preprocess data and perform feature selection
- Train a QSAR model to predict bioactivity
- Evaluate model performance and optimize hyperparameters
- Use the trained model to screen a virtual compound library

Module 7: Virtual Screening, Molecular Docking, and AI-Driven Drug Discovery with RDKit

7.1 Introduction to Virtual Screening

- What is virtual screening? Importance in drug discovery
- Ligand-based vs. structure-based virtual screening

- Workflow for in-silico screening using RDKit

7.2 Molecular Library Preparation for Screening

- Curating and preparing chemical libraries
- Filtering molecules using Lipinski’s Rule of 5
- Identifying and removing PAINS (Pan Assay Interference Compounds)
- Generating multiple conformations for screening

7.3 Similarity-Based Virtual Screening

- Using molecular fingerprints for virtual screening
- Measuring similarity using Tanimoto, Dice, and Cosine metrics
- Ranking molecules based on similarity to a reference compound

7.4 Structure-Based Molecular Docking

- Introduction to molecular docking and scoring functions
- Generating 3D molecular conformers for docking
- Docking ligands into target proteins using RDKit and external docking tools
- Analyzing binding interactions and docking scores

7.5 AI-Driven Drug Discovery with RDKit

- Using deep learning models for molecular property prediction
- Generating de novo drug-like molecules using generative models
- Integrating RDKit with TensorFlow and PyTorch for AI-based drug design

7.6 Hands-on Exercises

- Perform virtual screening on a library of drug-like molecules
- Use RDKit to filter and optimize hit compounds
- Dock selected molecules into a protein binding site
- Analyze docking results and refine ligand structures
- Train a deep learning model to generate novel drug candidates

Module 8: RDKit Integration with Data Science, Visualization, and Workflow Automation

8.1 RDKit and Data Science: Handling Large-Scale Molecular Datasets

- Reading and processing large datasets (SDF, CSV, and JSON)
- Using Pandas with RDKit for molecular data handling
- Optimizing performance for large-scale molecular processing

8.2 Advanced Molecular Visualization Techniques

- Generating high-quality molecular images using `rdkit.Chem.Draw`
- Customizing molecular drawings with atom and bond highlights
- Rendering 3D molecular structures using Py3Dmol and RDKit

8.3 Automating Workflow Pipelines with RDKit

- Building automated molecular filtering and processing workflows
- Batch processing and parallel computation techniques
- Using RDKit with Jupyter Notebooks for interactive analysis

8.4 RDKit Integration with External Tools and Libraries

- Connecting RDKit with Open Babel for format conversion
- Integrating RDKit with PyMOL for advanced visualization
- Using RDKit with deep learning libraries (TensorFlow, PyTorch)

8.5 Deploying RDKit-Based Applications

- Creating web-based cheminformatics applications using RDKit and Flask
- Building API services for molecular processing
- Deploying RDKit pipelines on cloud platforms (AWS, Google Cloud, Azure)

8.6 Hands-on Exercises

- Analyze and visualize a large molecular dataset using RDKit and Pandas
- Automate a workflow for filtering and saving drug-like molecules
- Integrate RDKit with Open Babel for batch molecular conversions
- Deploy a simple RDKit-based cheminformatics API

Module 9: Customizing RDKit for Advanced Applications and Research

9.1 Extending RDKit Functionality with Custom Scripts

- Modifying RDKit source code for specific research needs
- Creating custom molecular descriptors using Python
- Developing new chemical reaction rules with SMARTS patterns

9.2 Custom Fingerprint Design and Optimization

- Understanding the limitations of existing fingerprints
- Defining custom molecular fingerprint algorithms
- Optimizing fingerprints for improved QSAR performance

9.3 Machine Learning Model Deployment with RDKit

- Building and deploying trained QSAR models
- Creating web-based molecular screening tools
- Using cloud-based ML pipelines for large-scale cheminformatics

9.4 Implementing Custom Molecular Property Calculations

- Writing Python functions for unique molecular descriptors
- Automating molecular data processing for research applications
- Benchmarking custom descriptors against standard RDKit features

9.5 Integrating RDKit with High-Performance Computing (HPC)

- Running RDKit on GPU-accelerated environments
- Parallelizing cheminformatics tasks for large datasets
- Optimizing RDKit workflows for computational efficiency

9.6 Hands-on Exercises

- Develop and test a custom fingerprint algorithm
- Build a web tool for virtual screening using RDKit and Flask
- Deploy a cloud-based cheminformatics pipeline using RDKit
- Optimize an RDKit workflow for parallel execution on HPC clusters

Module 10: Real-World Case Studies and Advanced Project Work

10.1 Case Study: Drug Repurposing with RDKit

- Understanding the principles of drug repurposing
- Screening existing drugs for new targets using molecular similarity
- Analyzing structural and functional properties for repurposing potential

10.2 Case Study: Toxicity Prediction Using QSAR Models

- Building a predictive model for chemical toxicity
- Using molecular descriptors and machine learning for hazard assessment
- Validating and interpreting toxicity predictions

10.3 Case Study: Screening Natural Compounds for Drug Discovery

- Analyzing a dataset of plant-derived molecules
- Predicting bioavailability and ADMET properties
- Identifying lead compounds using docking and similarity screening

10.4 Case Study: Custom Molecular Design for a Target Protein

- Defining molecular requirements based on target structure
- Generating novel compounds using RDKit and AI-driven molecular design
- Docking simulations and optimization of lead compounds

10.5 Advanced Project Work

- Developing a cheminformatics pipeline for high-throughput screening
- Building an AI-based drug discovery model integrating RDKit and deep learning
- Creating a web-based molecular database with search and filtering features
- Automating molecular synthesis planning with RDKit and SMARTS-based reactions

10.6 Final Assessment and Certification

- Comprehensive project review and evaluation
- Submission of an independent research project using RDKit
- Certification of completion based on performance and practical assignments

RDKit Training Pricing (Given in Dollars keeping world wide queries into consideration)

Note: Only 30 mins to 60 mins class per day is entertained by our team, Tuesday to Friday are working days, Saturday, Sunday and Monday Class seekers will be charged 30% extra on total fee. All public holidays are as per Indian Calendar.

18% GST is charged extra to the selected slot fee.

1. Basic RDKit Workshop – $499

- **Duration:** 12 hours
- **Ideal for:** Students, entry-level cheminformatics professionals
- **Modules Covered:**
  - Module 1: Introduction to RDKit and Cheminformatics
  - Module 2: Molecular Representations and Structure Handling
  - Module 3: Molecular Fingerprints and Descriptor Calculation

2. Intermediate RDKit Training – $999

- **Duration:** 25 hours
- **Ideal for:** Industry professionals, bioinformatics researchers
- **Modules Covered:**
  - Module 1 to Module 3 (Basic Topics)
  - Module 4: Substructure Searching and Functional Group Identification
  - Module 5: Chemical Reactions and Molecular Transformations
  - Module 6: Machine Learning with RDKit – QSAR Modeling

3. Full RDKit Professional Course – $2,499

- **Duration:** 40+ hours
- **Ideal for:** Computational chemists, AI-driven drug discovery experts
- **Modules Covered:**
  - Module 1 to Module 6 (Basic & Intermediate Topics)
  - Module 7: Virtual Screening and Molecular Docking
  - Module 8: RDKit Integration with Data Science and Workflow Automation
  - Module 9: Customizing RDKit for Advanced Research

4. Custom Corporate Training – Starts at $5,000

- **Duration:** 50+ hours (Customized as per industry needs)
- **Ideal for:** Pharma R&D teams, biotech startups, cheminformatics research institutions
- **Modules Covered:**
  - All Modules (1-9) + Custom Modules Based on Enterprise Requirements
  - Module 10: Real-World Case Studies and Advanced Project Work