

Code recipe sheet

Four recipes that cover module two. Type them once and keep them in your own snippets file.

1 Descriptors from a SMILES string

- `from rdkit import Chem` and `from rdkit.Chem import Descriptors`
- `m = Chem.MolFromSmiles(smiles)`, then check that `m` is not `None` before using it
- `Descriptors.MolWt`, `MolLogP`, `NumHDonors`, `NumHAcceptors`, `TPSA` and `NumRotatableBonds`
- Collect the rows into a pandas DataFrame and store the canonical SMILES with each row

2 Properties from the public chemistry service

- Build the address from the service root, then compound, name, the drug name, property, the property list and JSON
- `requests.get` with a timeout, check the status code, then read the JSON
- Cache every response to a local file and record the date of the fetch
- Retry two or three times with a short wait and log which compounds failed

3 A structure and its ligand

- Download the entry file by its four character identifier from the archive file service
- ATOM records hold protein atoms. HETATM records hold the ligand, waters, ions and additives
- Find the ligand three letter code in the header, then filter the HETATM rows carrying that code
- Print how many copies of the ligand are present and keep each copy separately

4 A reproducible table

- One row per compound, one column per property, one source per column
- Write the table to the output folder with the date in the file name
- Record the package versions used to produce it
- Never hand type a value into a fetched table

Based on the ERAS Africa Molecular Docking using Python course, 2026, and the study "Target Protein Pocket Volume and Antiretroviral Drug Physicochemical Properties" (Hailekiros, 2026) with its reproduction guide. Tool documentation from RDKit, AutoDock Vina, Meeko, RCSB PDB, PubChem and ProteinsPlus. Free to copy and adapt with attribution.