

FITUR DAN TUTORIAL HITUNGAN KIMIA

Panduan Lengkap Penggunaan Fitur Komputasi ChemcomPy /3py
untuk Struktur XYZ, Python Multi-Metode, Solvent, Add-Ons, QSAR,
Frekuensi, dan Analisis Output

Fokus Buku

Buku ini menjelaskan fitur Komputasi secara prosedural: mulai dari membuka fitur, memahami trigger /3py, menyiapkan lingkungan WSL Ubuntu, membuat file XYZ dan Python, memilih metode komputasi, menjalankan script, membaca output, memakai add-ons, solvent, QSAR, momen dipol, dan melakukan validasi hasil.

<https://kasmui.cloud/aibuku/index.html#komputasi>

Identitas Buku

Komponen	Keterangan
Judul	Fitur dan Tutorial Hitungan Kimia
Ruang Lingkup	Tutorial penggunaan fitur Komputasi ChemcomPy /3py dari kode HTML yang diberikan.
Sasaran Pembaca	Dosen, mahasiswa, guru kimia, peneliti pemula, dan praktisi kimia komputasi yang ingin membuat paket perhitungan molekul secara terarah.
Prasyarat	Dasar struktur molekul, koordinat XYZ, terminal/PowerShell, Python, dan pemahaman awal metode kimia komputasi.
Sifat Dokumen	Panduan prosedural, bukan validasi ilmiah final untuk semua metode. Setiap angka hasil komputasi harus diverifikasi dari output program yang benar-benar berjalan.

Kata Pengantar

Fitur Komputasi dalam kode program ini dirancang sebagai pusat kerja kimia komputasi yang menggabungkan pembuatan struktur molekul, validasi file, pembuatan file Python, pemilihan metode, serta penyiapan output hasil perhitungan dalam satu alur terpadu. Kehadiran trigger /3py memudahkan pengguna karena tidak perlu lagi menjalankan beberapa prompt secara terpisah untuk membuat file XYZ, memvalidasi koordinat, dan menghasilkan script Python.

Buku ini disusun agar pengguna dapat memahami fungsi setiap bagian secara runtut. Pembahasan dimulai dari orientasi fitur, persiapan lingkungan WSL Ubuntu, penggunaan prompt terpadu /3py, pemilihan metode komputasi, solvent, add-ons, QSAR, hingga interpretasi output dan troubleshooting. Setiap bab dilengkapi langkah kerja, contoh perintah, daftar output, dan checklist praktik.

Prinsip penting yang digunakan dalam buku ini adalah kejujuran komputasi. Jika backend atau pustaka tertentu belum tersedia, script tidak boleh membuat hasil fiktif. Nilai seperti energi solvasi, intensitas Raman, chemical shift NMR, heat of formation, NBO, polarizability, atau hydration energy harus dihitung hanya jika engine yang digunakan benar-benar mendukungnya; jika tidak, output wajib mencatat status tidak tersedia atau tidak dihitung.

Daftar Isi

- **BAB 1. Pengenalan Fitur Komputasi ChemcomPy /3py**
 - 1.1 Hakikat dan tujuan fitur Komputasi
 - 1.2 Peta fitur utama
 - 1.3 Alur kerja dari prompt sampai output
 - 1.4 File yang dihasilkan dan kegunaannya
 - 1.5 Prinsip validasi dan kejujuran hasil
- **BAB 2. Persiapan Lingkungan Hitungan Kimia di Windows dan WSL Ubuntu**
 - 2.1 Arsitektur Windows -> WSL 2 -> Ubuntu -> Conda -> Python
 - 2.2 Instalasi WSL Ubuntu dan pengaturan folder kerja
 - 2.3 Instalasi Miniforge/Conda dan environment kimia
 - 2.4 Library utama: RDKit, Open Babel, PySCF, Psi4, xTB, JupyterLab
 - 2.5 Pengujian instalasi dan troubleshooting awal
- **BAB 3. Tutorial Menggunakan Trigger /3py dan Menyiapkan Paket Molekul**
 - 3.1 Format dasar trigger /3py
 - 3.2 Membuat struktur XYZ tervalidasi
 - 3.3 Membuat file Python multi-metode
 - 3.4 Pemilihan metode: MM, semiempirik, HF, DFT, hybrid DFT
 - 3.5 Basis set, charge, spin, job, dispersi, frekuensi, dan dipol
- **BAB 4. Tutorial Fitur Lanjutan: Solvent, Add-Ons, QSAR, Frekuensi, dan Dipol**
 - 4.1 PCM Solvent dan model solvasi
 - 4.2 Add-ons IR, Thermodynamics, UV-Vis, NMR, Raman, Delta Hf, NBO
 - 4.3 QSAR Properties dan descriptor molekul
 - 4.4 Frekuensi vibrasi dan validasi minimum lokal
 - 4.5 Momen dipol dan analisis sifat molekul
- **BAB 5. Membaca Output, Praktik Kasus, Validasi Hasil, dan Troubleshooting**
 - 5.1 Menjalankan script dan menyimpan log
 - 5.2 Membaca hasil energi, geometri, orbital, dan spektrum
 - 5.3 Studi praktik bertahap
 - 5.4 Checklist validasi hasil
 - 5.5 Troubleshooting dan pengembangan lanjutan

BAB 1

PENGENALAN FITUR KOMPUTASI

ChemcomPy /3py

1.1 Hakikat dan Tujuan Fitur Komputasi

Fitur Komputasi adalah blok kerja khusus dalam halaman aplikasi yang diberi identitas Komputasi dan diarahkan untuk membantu pengguna membuat paket awal hitungan kimia komputasi. Inti fitur ini adalah prompt terpadu ChemcomPy /3py yang menyatukan tiga pekerjaan yang sebelumnya terpisah: pembuatan struktur molekul, validasi file XYZ, dan pembuatan file Python multi-metode.

Dengan pendekatan tersebut, pengguna cukup menulis satu trigger untuk memperoleh dua file utama, yaitu file koordinat molekul berformat .xyz dan file script Python berformat .py. File XYZ dipakai sebagai struktur awal, sedangkan file Python dipakai untuk menjalankan optimasi, single point, frekuensi, momen dipol, QSAR, solvent, dan add-ons sesuai pilihan.

Gagasan Utama

Fitur Komputasi bukan sekadar tombol untuk menyalin prompt, tetapi sebuah workflow: menentukan molekul, membuat struktur awal, memvalidasi koordinat, memilih metode, menjalankan perhitungan, menyimpan output, lalu membaca hasil secara ilmiah.

1.2 Peta Fitur Utama

Berdasarkan kode program, blok Komputasi menampilkan beberapa kartu akses dan panel tutorial. Setiap kartu memiliki fungsi tertentu dan perlu dipahami sebagai bagian dari satu rangkaian kerja.

Fitur	Fungsi Praktis
Tutorial WSL Ubuntu	Menyiapkan sistem Linux di Windows, environment Conda, Python, dan library kimia komputasi.
Add-Ons	Mengaktifkan keluaran tambahan seperti IR, Thermodynamics, UV-Vis, NMR, Raman, Delta Hf, dan NBO.
PCM Solvent	Memilih pelarut dan model solvasi seperti none, PCM, CPCM, DDCOSMO, SMD, COSMO, ALPB, atau GBSA.
QSAR Properties	Menghasilkan descriptor molekul seperti muatan parsial, LogP, refractivity, surface area, volume, polarizability, hydration energy, dan massa.
Prompt Terpadu /3py	Membuat paket lengkap file XYZ dan Python multi-metode dengan format trigger tunggal.

1.3 Alur Kerja dari Prompt sampai Output

Alur kerja fitur Komputasi dapat dibaca sebagai tahapan produksi paket hitungan kimia. Pengguna tidak langsung masuk ke interpretasi data, tetapi harus memastikan struktur, metode, lingkungan, dan output sudah benar.

1. Buka halaman aplikasi dan pilih menu Komputasi.
2. Baca bagian Tutorial WSL Ubuntu jika komputer belum siap menjalankan Python kimia komputasi.
3. Buka Prompt Terpadu /3py, salin prompt, kemudian gunakan trigger sesuai nama molekul.
4. Tentukan nama molekul dan nama file XYZ dalam tanda kutip, misalnya /3py "etanol etanol.xyz".
5. Tambahkan argumen opsional: family, engine, method, basis, job, charge, spin, solvent, addons, qsar, dipole, atau frekuensi.
6. Minta AI menghasilkan file .xyz dan .py sesuai nama molekul.
7. Simpan file ke folder kerja WSL atau folder proyek komputasi.
8. Jalankan file Python melalui terminal untuk memperoleh output log dan file hasil.
9. Baca hasil energi, geometri akhir, frekuensi, orbital, solvent, QSAR, dan status dukungan backend.
10. Validasi hasil sebelum dipakai untuk laporan, praktikum, atau riset.

1.4 File yang Dihasilkan dan Kegunaannya

File Output	Kegunaan
nama-molekul.xyz	Struktur awal tervalidasi dalam format XYZ; dipakai untuk input komputasi dan visualisasi.
nama-molekul.py	Script Python utama untuk menjalankan metode yang dipilih.
running_output.log	Catatan jalannya program, waktu awal, waktu akhir, status sukses/gagal, dan pesan error.
final_structure.xyz	Koordinat geometri akhir setelah optimasi jika job optimize dijalankan.
energy_summary.txt/csv/json	Ringkasan energi, metode, basis set, charge, spin, dan status konvergensi.
frequency_summary.txt/csv/json	Daftar frekuensi vibrasi, mode imajiner jika ada, dan validasi minimum lokal.
qsar_properties.txt/csv/json	Descriptor QSAR untuk tabulasi data dan pemodelan.
solvent_summary.txt/csv/json	Ringkasan perhitungan pelarut, model solvasi, energi gas, energi solvasi jika tersedia.
addons_summary.txt/csv/json	Status keluaran IR, Thermodynamics, UV-Vis, NMR, Raman, Delta Hf, dan NBO.

1.5 Prinsip Validasi dan Kejujuran Hasil

Dalam kimia komputasi, hasil yang rapi belum tentu benar. Karena itu fitur Komputasi harus dipakai dengan prinsip validasi. Validasi paling awal adalah memastikan jumlah atom, simbol atom, koordinat, charge, spin, metode, basis set, dan backend sesuai tujuan. Validasi berikutnya adalah

memeriksa konvergensi optimasi, tidak adanya frekuensi imajiner untuk minimum lokal, serta konsistensi energi dan geometri.

- **Jangan membuat angka fiktif.** Jika backend tidak mendukung solvent, NMR, Raman, NBO, Delta Hf, atau polarizability, output harus menulis not_available, failed, atau tidak dihitung.
- **Pisahkan hasil cepat dan hasil riset.** Descriptor RDKit seperti LogP dan refractivity dapat cepat dihitung, sedangkan energi kuantum memerlukan backend quantum chemistry.
- **Catat parameter lengkap.** Setiap hasil harus menyimpan metode, basis set, charge, spin, job, solvent, model solvasi, dan add-ons.
- **Verifikasi struktur.** Struktur awal dari AI atau RDKit harus diperiksa di Avogadro, Jmol, atau viewer lain sebelum dipakai untuk riset serius.
- **Gunakan pembanding.** Untuk laporan riset, bandingkan minimal dua metode/basis set atau bandingkan fase gas dengan solvent jika relevan.

BAB 2

PERSIAPAN LINGKUNGAN HITUNGAN KIMIA DI WINDOWS DAN WSL UBUNTU

2.1 Arsitektur Windows -> WSL 2 -> Ubuntu -> Conda -> Python

Fitur Komputasi dalam kode memasukkan tutorial WSL Ubuntu karena banyak pustaka kimia komputasi lebih stabil dijalankan pada lingkungan Linux. Windows tetap dapat dipakai sebagai sistem utama, sedangkan WSL 2 menyediakan Ubuntu sebagai lingkungan Linux yang berjalan berdampingan dengan Windows.

Lapisan	Peran
Windows	Sistem operasi utama untuk browser, editor, file explorer, dan PowerShell.
WSL 2	Lapisan kompatibilitas Linux yang menjalankan kernel Linux di Windows.
Ubuntu	Distribusi Linux tempat Python, Conda, dan paket komputasi diinstal.
Miniforge/Conda	Manajer environment untuk mengatur versi Python dan dependensi ilmiah.
Python Kimia	Lingkungan kerja untuk RDKit, Open Babel, PySCF, Psi4, xTB, JupyterLab, dan ChemcomPy.

2.2 Instalasi WSL Ubuntu dan Pengaturan Folder Kerja

Instalasi WSL sebaiknya dilakukan dari PowerShell Administrator. Setelah Ubuntu terpasang, pengguna membuat folder proyek khusus agar file XYZ, Python, log, dan output tidak bercampur dengan file sistem.

```
wsl --install
wsl --list --verbose
wsl -d Ubuntu
```

11. Buka PowerShell sebagai Administrator.
12. Jalankan perintah `wsl --install` untuk memasang WSL dan Ubuntu default.
13. Restart Windows jika diminta.
14. Buka Ubuntu dari Start Menu atau terminal.
15. Buat username dan password Linux.
16. Cek status WSL dengan `wsl --list --verbose`.
17. Masuk ke Ubuntu dan buat folder kerja proyek.

```
mkdir -p ~/kimia-komputasi
cd ~/kimia-komputasi
mkdir -p struktur script output log data
```

Folder yang disarankan adalah struktur untuk file XYZ, script untuk file Python, output untuk hasil perhitungan, log untuk catatan running, dan data untuk tabel descriptor atau hasil rekap.

2.3 Instalasi Miniforge/Conda dan Environment Kimia

Conda membantu menghindari benturan dependensi. Gunakan environment khusus, misalnya bernama kimia, sehingga paket kimia komputasi tidak mengganggu Python sistem.

```
sudo apt update && sudo apt upgrade -y
sudo apt install -y wget curl git build-essential
# Setelah Miniforge terpasang:
conda create -n kimia python=3.12 -y
conda activate kimia
python --version
```

Setiap kali menjalankan script ChemcomPy, aktifkan environment terlebih dahulu. Jika prompt terminal menunjukkan (kimia), berarti environment sudah aktif.

2.4 Library Utama: RDKit, Open Babel, PySCF, Psi4, xTB, JupyterLab

Kode fitur Komputasi menyebut beberapa komponen kunci: RDKit untuk cheminformatics dan descriptor cepat, Open Babel untuk konversi format molekul, PySCF untuk hitungan kuantum berbasis Python, Psi4 untuk quantum chemistry, xTB untuk semiempirik modern, serta JupyterLab untuk eksplorasi interaktif.

```
conda install -c conda-forge numpy scipy pandas matplotlib jupyterlab rdkit openbabel -y
pip install pyscf h5py
# Opsional jika tersedia/kompatibel:
conda install -c conda-forge psi4 xtb-python -y
```

Library/Tool	Fungsi
numpy/scipy/pandas	Komputasi numerik, tabel, dan pengolahan output.
matplotlib	Membuat grafik energi, spektrum, atau descriptor.
RDKit	SMILES, struktur 3D awal, descriptor QSAR, LogP, refractivity, Gasteiger charge, volume.
Open Babel	Konversi format XYZ, MOL, SDF, PDB, CML, dan format kimia lain.
PySCF	HF, DFT, optimasi, orbital, momen dipol, dan properti kuantum tertentu.
Psi4	Alternatif backend quantum chemistry untuk HF/DFT dan fitur tertentu.
xTB	Optimasi cepat dan metode semiempirik modern seperti GFN-xTB.
JupyterLab	Notebook interaktif untuk pembelajaran dan eksplorasi hasil.

2.5 Pengujian Instalasi

Pengujian dilakukan sebelum menjalankan file molekul yang lebih besar. Mulailah dari import library, lalu lanjutkan dengan hitungan kecil seperti air menggunakan HF/STO-3G atau DFT/B3LYP dengan basis ringan.

```
python - <<'PY'
import numpy, scipy, pandas
print("Numerik OK")
try:
    import rdkit
    print("RDKit OK")
```

```

except Exception as e:
    print("RDKit belum siap:", e)
try:
    import pyscf
    print("PySCF OK")
except Exception as e:
    print("PySCF belum siap:", e)
PY

python - <<'PY'
from pyscf import gto, scf
mol = gto.M(atom="O 0 0 0; H 0 0 0.96; H 0.92 0 -0.24", basis="sto-3g")
mf = scf.RHF(mol)
print("Energi HF air =", mf.kernel())
PY

```

2.6 Troubleshooting Awal

Masalah	Solusi Praktis
ModuleNotFoundError	Environment belum aktif atau paket belum terpasang. Aktifkan conda activate kimia lalu instal ulang paket.
wsl command not found/timeout	Cek fitur Windows, koneksi internet, dan instalasi WSL melalui PowerShell Administrator.
Psi4 atau xTB gagal instal	Gunakan Conda Forge, cek versi Python, atau jadikan fitur tersebut opsional.
DFT-D3/D4 tidak tersedia	Jangan paksa angka dispersi. Tulis status not_available jika paket dispersi tidak ada.
Optimasi tidak konvergen	Gunakan basis lebih ringan, ubah struktur awal, atau jalankan MM/semiempirik sebelum DFT.
Frekuensi imajiner muncul	Struktur bukan minimum lokal. Lakukan optimasi ulang atau ubah geometri awal.

BAB 3

TUTORIAL MENGGUNAKAN TRIGGER /3py DAN MENYIAPKAN PAKET MOLEKUL

3.1 Format Dasar Trigger /3py

Trigger /3py adalah perintah utama untuk meminta AI menghasilkan paket komputasi molekul. Format dasarnya menggunakan nama molekul dan nama file XYZ dalam tanda kutip. Nama file sebaiknya tanpa spasi agar mudah dijalankan di terminal.

```
/3py "nama-molekul nama-molekul.xyz"
```

Contoh paling sederhana adalah membuat paket molekul air atau etanol. Setelah trigger dikirim, AI diarahkan menghasilkan file nama-molekul.xyz dan nama-molekul.py. File tersebut kemudian diunduh, disimpan ke folder proyek, dan dijalankan di WSL atau Python environment yang sesuai.

Contoh Trigger dari Fitur Komputasi

```
/3py "nama-molekul nama-molekul.xyz"
/3py "metilhidrazina metilhidrazina.xyz"
/3py "aspirin aspirin.xyz" --job optimize --method B3LYP --basis def2-SVP
/3py "paracetamol paracetamol.xyz" --job singlepoint --method HF --basis STO-3G
/3py "asam asetat asam-asetat.xyz" --charge 0 --spin 0
/3py "benzena benzena.xyz" --job optimize_freq --qsar --method B3LYP --basis 6-311++G(d,p)
/3py "fenol fenol.xyz" --qsar-only --basis def2-TZVPP --charge 0 --spin 0
/3py "etanol etanol.xyz" --family mm --engine rdkit --force-field MMFF94 --job optimize --dipole
/3py "nikotin nikotin.xyz" --family semiempirical --engine mopac --semiempirical-method PM7 --job optimize_freq --dipole
/3py "benzaldehida benzaldehyda.xyz" --family dft --method CAM-B3LYP --basis def2-TZVP --disp D3BJ --dipole --qsar
/3py "asam benzoat asam-benzoat.xyz" --family dft --method B3LYP --basis def2-SVP --solvent water --solvation-model pcm --dipole
/3py "benzaldehyda benzaldehyda.xyz" --family dft --method CAM-B3LYP --basis def2-TZVP --pcm-solvent methanol --solvation-model cpcm --addons uv-vis,nmr
/3py "anilin anilin.xyz" --family dft --method PBE0 --basis def2-TZVP --solvent thf --solvation-model smd --qsar
/3py "benzena benzena.xyz" --family dft --method B3LYP --basis 6-31G(d) --addons ir,raman,thermodynamics --dipole
/3py "anisole anisole.xyz" --family dft --method CAM-B3LYP --basis def2-TZVP --addons uv-vis,nmr --qsar
/3py "formaldehida formaldehyda.xyz" --engine psi4 --addons dhf-g3mp2,nbo-def2-tzpp --basis def2-TZPP
```

3.2 Membuat Struktur XYZ Tervalidasi

File XYZ adalah struktur awal molekul. Baris pertama berisi jumlah atom, baris kedua berisi keterangan molekul, dan baris berikutnya berisi simbol atom serta koordinat X, Y, Z dalam satuan Angstrom. Validasi file XYZ sangat penting karena kesalahan satu atom saja dapat menyebabkan perhitungan gagal atau menghasilkan struktur yang salah.

Bagian File XYZ	Isi	Validasi
Baris 1	Jumlah atom total	Harus sama dengan jumlah baris koordinat.
Baris 2	Nama molekul, rumus, dan keterangan	Tidak boleh menggantikan baris koordinat.
Kolom 1	Simbol atom	Harus simbol unsur yang benar: C, H, O, N, Cl, Br, dan seterusnya.
Kolom 2-4	Koordinat X, Y, Z	Harus berupa angka dan tidak kosong.
Struktur 3D	Susunan atom masuk akal	Periksa jarak ikatan, tumpang tindih atom, dan valensi.

```
9
Metilhidrazina (CH6N2) | initial 3D ready for optimize
C  -1.07735486   0.13266289   0.03740743
N   0.16450861  -0.61323608   0.22525374
N   1.19439942  -0.02010180  -0.58682865
H  -1.39615186   0.11738261  -1.01050926
H  -0.97486628   1.17269341   0.36566127
H  -1.87245272  -0.33104451   0.62971098
H   0.40202906  -0.63345638   1.22157328
H   1.57601668   0.83573163  -0.17655475
H   1.98387196  -0.66063178  -0.45000000
```

3.3 Membuat File Python Multi-Metode

File Python yang dihasilkan oleh fitur Komputasi harus bersifat siap jalan, modular, dan mampu menyimpan output. Secara ideal, file tersebut memiliki parser argumen terminal, validasi input, pemilihan metode, fungsi pembacaan XYZ, fungsi optimasi, fungsi single point, fungsi frekuensi, fungsi QSAR, fungsi solvent, fungsi add-ons, dan penulis log.

Modul/Fungsi Script	Tugas
read_xyz()	Membaca atom dan koordinat dari file XYZ.
validate_structure()	Memeriksa simbol atom, jumlah atom, koordinat, dan struktur awal.
select_method()	Memilih family, engine, method, basis, dispersi, charge, spin, dan job.
run_mm()	Menjalankan mekanika molekuler atau optimasi awal berbasis force field.
run_semiempirical()	Menjalankan metode semiempirik atau backend eksternal jika tersedia.
run_hf_dft()	Menjalankan HF, DFT, atau hybrid DFT.
run_frequency()	Menghitung frekuensi vibrasi dan mengecek minimum lokal.
run_dipole()	Menghitung momen dipol jika backend mendukung.

run_qsar()	Menghitung descriptor QSAR cepat/kuantum sesuai dukungan.
write_outputs()	Menyimpan TXT, CSV, JSON, XYZ, log, dan ringkasan.

3.4 Pemilihan Metode Komputasi

Metode dipilih sesuai tujuan. Tidak semua molekul memerlukan DFT berat; untuk struktur awal, mekanika molekuler atau semiempirik sering lebih praktis. Untuk hasil energi dan sifat elektronik yang lebih baik, HF, DFT, atau hybrid DFT dapat dipakai dengan basis set yang sesuai.

Keluarga Metode	Contoh	Kapan Dipakai
Mekanika Molekuler	MMFF94, MMFF94s, UFF, GAFF, Ghemical, OPLS, AMBER/CHARMM jika tersedia	Optimasi awal cepat, struktur organik, pembelajaran.
Semiempirik	AM1, RM1, PM3, PM4, PM5, PM6, PM7, MNDO, GFN1-xTB, GFN2-xTB	Molekul sedang/besar, screening cepat, optimasi sebelum DFT.
Ab Initio/HF	HF dengan STO-3G, 3-21G, 6-31G, def2-SVP, dan lain-lain	Dasar teori orbital, pembandingan sederhana, latihan kuantum.
DFT	LDA, PBE, BP86, BLYP, M06-L, TPSS, revPBE	Energi dan geometri dengan biaya sedang.
Hybrid DFT	B3LYP, PBE0, B97, wB97X, wB97X-D, CAM-B3LYP, M06-2X, TPSSh	Riset organik, spektrum, sifat elektronik, HOMO-LUMO.
Double Hybrid/lanjutan	B2PLYP, DSD-BLYP jika backend mendukung	Riset presisi lebih tinggi dengan biaya lebih besar.

3.5 Basis Set, Charge, Spin, Job, Dispersi, Frekuensi, dan Dipol

Argumen komputasi harus ditulis eksplisit. Charge dan spin sangat menentukan hasil. Molekul netral biasanya charge 0. Spin dalam banyak script sering dinyatakan sebagai 2S atau jumlah elektron tidak berpasangan; pastikan mengikuti format script yang dipakai.

Argumen	Contoh Nilai	Fungsi
--job	optimize, singlepoint, optimize_freq, freq, qsar-only	Jenis pekerjaan utama.
--method	HF, B3LYP, PBE0, CAM-B3LYP, M06-2X, PM7, MMFF94	Metode yang digunakan.
--basis	STO-3G, 6-31G(d), 6-311++G(d,p), def2-SVP, def2-TZVP, def2-TZVPP	Basis set untuk HF/DFT.
--charge	0, +1, -1, dan seterusnya	Muatan total molekul.
--spin	0, 1, 2, dan seterusnya sesuai konvensi script	Keadaan spin/multiplicity terkait elektron tidak berpasangan.
--disp	none, D3, D3BJ, D4 jika tersedia	Koreksi dispersi. Jangan membuat koreksi jika modul tidak tersedia.

--dipole	flag aktif	Menghitung momen dipol setelah electronic structure selesai.
--qsar / --qsar-only	flag aktif	Menghitung descriptor QSAR bersama workflow utama atau hanya QSAR.
--solvent	none, water, methanol, ethanol, acetone, thf	Memilih pelarut.
--addons	ir, thermodynamics, uv-vis, nmr, raman, dhf-g3mp2, nbo-def2-tzpp, none	Mengaktifkan fitur tambahan.

3.6 Prosedur Praktik Membuat Paket Molekul

18. Tentukan nama molekul dan pastikan penamaan file konsisten, misalnya fenol -> fenol.xyz dan fenol.py.
19. Mulai dari trigger sederhana: /3py "fenol fenol.xyz".
20. Untuk perhitungan ringan, tambahkan --method B3LYP --basis STO-3G atau --basis def2-SVP.
21. Untuk optimasi dan frekuensi, gunakan --job optimize_freq agar script melakukan optimasi lalu frekuensi.
22. Untuk solvent, tambahkan --solvent water --solvation-model pcm.
23. Untuk descriptor, tambahkan --qsar atau gunakan --qsar-only jika hanya butuh descriptor cepat.
24. Untuk spektrum atau termokimia, gunakan --addons ir,raman,thermodynamics.
25. Setelah file dihasilkan, simpan ke folder proyek WSL.
26. Jalankan script Python dan periksa running_output.log.
27. Buka file output di Avogadro/Jmol atau olah tabel CSV/JSON untuk laporan.

BAB 4

TUTORIAL FITUR LANJUTAN: SOLVENT, ADD-ONS, QSAR, FREKUENSI, DAN DIPOL

4.1 PCM Solvent dan Model Solvasi

Panel PCM Solvent menyediakan pilihan None, Water, Methanol, Ethanol, Acetone, dan THF. None berarti fase gas atau vakum. Pelarut lain diarahkan ke model solvasi implisit jika backend mendukung. Perhitungan solvent penting untuk mempelajari stabilisasi muatan, perubahan energi, efek lingkungan pada spektrum, dan perbandingan fase gas vs larutan.

Pelarut	Karakter	Kegunaan
None	Fase gas/vakum	Default paling aman dan ringan.
Water	Pelarut polar	Hidrasi, ion, biomolekul, spektrum dalam air.
Methanol	Pelarut protik polar	Perbandingan alkohol, spektrum, energi.
Ethanol	Pelarut protik umum	Kimia organik/farmasi dan pembanding air/metanol.
Acetone	Aprotik polar	Efek dielektrik pada molekul organik.
THF	Aprotik eterik	Reaksi organik, organologam, medium non-air.

Model yang diarahkan dalam kode mencakup none, PCM, CPCM, DDCOSMO, SMD, COSMO, ALPB, dan GBSA sesuai kemampuan backend. Jika model tidak tersedia, script wajib menyatakan tidak tersedia, bukan membuat energi solvasi palsu.

```
/3py "asam benzoat asam-benzoat.xyz" --family dft --method B3LYP --basis def2-SVP --  
solvent water --solvation-model pcm --dipole  
/3py "anilin anilin.xyz" --family dft --method PBE0 --basis def2-TZVP --solvent thf --  
solvation-model smd --qsar
```

4.2 Add-Ons IR, Thermodynamics, UV-Vis, NMR, Raman, Delta Hf, NBO, dan None

Add-ons adalah fitur tambahan di luar workflow utama. Pengguna dapat memilih satu atau beberapa add-ons sesuai tujuan. Dalam implementasi yang baik, add-ons tidak boleh mengganggu perhitungan utama. Jika add-on gagal, script tetap menyimpan output utama dan menulis status add-on secara jelas.

Add-On	Output yang Diharapkan	Syarat Dukungan
IR	Frekuensi vibrasi, intensitas IR jika tersedia, mode normal, file spektrum.	Memerlukan frekuensi/Hessian dan dukungan intensitas.
Thermodynamics	ZPE, koreksi termal, entalpi, entropi, Gibbs free energy.	Memerlukan frekuensi dan suhu/tekanan.

UV-Vis	Energi eksitasi, panjang gelombang, oscillator strength, transisi elektronik.	Memerlukan TD-DFT/CIS/backend eksternal.
NMR	Chemical shielding/shift untuk ¹ H, ¹³ C, ¹⁵ N, ¹⁹ F, ³¹ P, atau inti lain.	Memerlukan modul properti NMR.
Raman	Frekuensi dan aktivitas/intensitas Raman.	Memerlukan turunan polarizability atau output Raman.
Delta Hf(g3mp2)	Heat of formation berbasis model komposit.	Hanya jika engine mendukung G3MP2/G3/G4/G4MP2/CBS-QB3.
NBO def2-tzpp	Analisis Natural Bond Orbital.	Memerlukan NBO/Gennbo/backend eksternal.
None	Menonaktifkan add-ons.	Workflow utama tetap berjalan tanpa keluaran tambahan.

```
/3py "benzena benzena.xyz" --family dft --method B3LYP --basis 6-31G(d) --addons
ir,raman,thermodynamics --dipole
/3py "anisole anisole.xyz" --family dft --method CAM-B3LYP --basis def2-TZVP --addons uv-
vis,nmr --qsar
/3py "air air.xyz" --addons none --method B3LYP --basis STO-3G
```

4.3 QSAR Properties dan Descriptor Molekul

QSAR Properties adalah fitur untuk menghasilkan descriptor yang dapat dipakai dalam tabulasi data, pemodelan hubungan struktur-aktivitas, laporan praktikum, dan screening molekul. Kode menampilkan properti seperti Partial Charges, Hydration Energy, LogP, Surface Area, Refractivity, Polarizability, Volume, dan Mass.

Descriptor	Cara Perolehan	Catatan
Partial Charges	Gasteiger charge dari RDKit; Mulliken/Lowdin dari PySCF jika tersedia.	Periksa metode dan basis set karena muatan parsial bergantung metode.
Hydration Energy	Hanya dihitung jika model pendukung tersedia.	Jika tidak ada model, tulis tidak dihitung.
LogP	Crippen MolLogP dari RDKit.	Cocok untuk descriptor QSAR cepat.
Surface Area	LabuteASA atau grid/FreeSASA jika tersedia.	Pilih approx untuk cepat, grid untuk lebih rinci jika library ada.
Refractivity	Molar refractivity RDKit.	Sering dipakai bersama LogP.
Polarizability	Dihitung jika modul properti kuantum tersedia.	Jangan estimasi palsu jika tidak tersedia.
Volume	ComputeMolVolume atau fallback geometri.	Struktur 3D harus masuk akal.
Mass	Molecular weight dan exact molecular weight.	Dapat dihitung cepat dari formula.

```
/3py "benzena benzena.xyz" --job optimize_freq --qsar --method B3LYP --basis 6-311++G(d,p)
/3py "fenol fenol.xyz" --qsar-only --basis def2-TZVPP --charge 0 --spin 0
```

4.4 Frekuensi Vibrasi dan Validasi Minimum Lokal

Frekuensi vibrasi bukan hanya untuk spektrum IR/Raman. Dalam optimasi geometri, frekuensi dipakai untuk memastikan struktur akhir adalah minimum lokal. Struktur minimum lokal seharusnya tidak memiliki frekuensi imajiner. Jika ada satu frekuensi imajiner, struktur dapat berupa transition state atau geometri belum stabil.

28. Jalankan job optimize terlebih dahulu untuk memperoleh geometri akhir.
29. Lanjutkan dengan perhitungan frekuensi pada geometri akhir.
30. Periksa jumlah frekuensi imajiner.
31. Jika nol frekuensi imajiner, struktur dapat dinyatakan minimum lokal pada level teori tersebut.
32. Jika ada frekuensi imajiner, visualisasikan mode vibrasinya dan optimasi ulang mengikuti arah mode tersebut.
33. Gunakan hasil frekuensi untuk ZPE, termokimia, IR, dan Raman jika backend mendukung.

4.5 Momen Dipol dan Analisis Sifat Molekul

Momen dipol menggambarkan pemisahan muatan dalam molekul dan sering dipakai untuk membandingkan polaritas, efek solvent, dan perubahan struktur. Pada fitur Komputasi, momen dipol diarahkan sebagai flag --dipole agar script menghitung dan menyimpan hasilnya jika backend mendukung.

- Bandingkan momen dipol fase gas dan solvent untuk melihat efek pelarut.
- Gunakan basis set yang memadai jika momen dipol dipakai untuk laporan riset.
- Simpan komponen vektor dipol dan besar total dalam Debye jika tersedia.
- Catat metode dan basis karena momen dipol sensitif terhadap level teori.
- Jangan menyimpulkan polaritas hanya dari satu descriptor; kombinasikan dengan charge, geometri, dan solvent.

4.6 Integrasi Solvent, Add-Ons, QSAR, Frekuensi, dan Dipol dalam Satu Workflow

Kekuatan fitur Komputasi adalah kemampuannya menyatukan banyak kebutuhan dalam satu perintah. Namun, semakin banyak fitur diaktifkan, semakin besar biaya komputasi dan semakin tinggi kemungkinan backend tertentu belum mendukung semua output. Karena itu pengguna perlu menyusun workflow bertahap.

Tahap	Perintah/Strategi	Tujuan
Tahap 1	MM atau semiempirik optimize	Mendapat struktur awal yang lebih baik.
Tahap 2	DFT optimize basis sedang	Mendapat geometri final.
Tahap 3	Frequency	Memastikan minimum lokal dan memperoleh data vibrasi.
Tahap 4	Single point basis lebih besar	Memperbaiki energi elektronik tanpa optimasi ulang berat.

Tahap 5	Solvent dan dipole	Menganalisis efek pelarut dan polaritas.
Tahap 6	QSAR dan add-ons	Menyusun tabel descriptor dan spektrum.

BAB 5

MEMBACA OUTPUT, PRAKTIK KASUS, VALIDASI HASIL, DAN TROUBLESHOOTING

5.1 Menjalankan Script dan Menyimpan Log

Setelah file .xyz dan .py diperoleh, simpan keduanya dalam satu folder. Jalankan terminal WSL, aktifkan environment kimia, lalu jalankan script. Semua output penting sebaiknya disimpan ke file log agar jejak running dapat diperiksa ulang.

```
cd ~/kimia-komputasi/fenol
conda activate kimia
python fenol.py --xyz-file fenol.xyz --job optimize_freq --method B3LYP --basis def2-SVP -
-charge 0 --spin 0 --dipole --qsar 2>&1 | tee fenol_running_output.log
```

Perintah tee membuat output tampil di terminal sekaligus tersimpan ke file log. Jika script sudah memiliki penulis log internal, file log tetap berguna sebagai catatan eksternal.

5.2 Membaca Hasil Energi, Geometri, Orbital, dan Spektrum

Pembacaan output dilakukan secara berurutan. Jangan langsung mengambil energi akhir tanpa mengecek apakah perhitungan selesai normal. Jika status gagal, energi atau descriptor tidak boleh dipakai sebagai hasil final.

Komponen	Yang Dibaca	File
Status running	Apakah program selesai normal, gagal, atau berhenti karena error.	running_output.log
Energi akhir	Energi elektronik, energi total, koreksi termal, atau energi solvasi.	energy_summary.txt/csv/json
Geometri akhir	Koordinat atom final, ikatan, sudut, dihedral.	final_structure.xyz dan geometry_summary.txt
Frekuensi	Jumlah frekuensi imajiner, frekuensi normal, ZPE, IR/Raman jika ada.	frequency_summary.txt/csv/json
Orbital	HOMO, LUMO, gap, energi orbital, occupancy.	orbital_summary.txt/csv
QSAR	LogP, refractivity, ASA, volume, mass, charge, polarizability.	qsar_properties.csv/json
Solvent	Nama pelarut, model solvasi, energi gas/solvent, status model.	solvent_summary.txt/json
Add-ons	Status IR, UV-Vis, NMR, Raman, Delta Hf, NBO.	addons_summary.txt/json

5.3 Studi Praktik Bertahap

Bagian ini dapat dipakai sebagai latihan pembelajaran. Urutan dibuat dari molekul kecil menuju workflow yang lebih lengkap.

Latihan	Trigger	Tujuan
Praktik 1: Air	<code>/3py "air air.xyz" --addons none --method HF --basis STO-3G</code>	Memahami format XYZ, HF sederhana, dan output energi.
Praktik 2: Etanol	<code>/3py "etanol etanol.xyz" --family mm --engine rdkit --force-field MMFF94 --job optimize --dipole</code>	Optimasi cepat, momen dipol, dan struktur organik kecil.
Praktik 3: Benzena	<code>/3py "benzena benzena.xyz" --family dft --method B3LYP --basis 6-31G(d) --addons ir,raman,thermodynamics --dipole</code>	Frekuensi, IR/Raman, dan termokimia.
Praktik 4: Fenol	<code>/3py "fenol fenol.xyz" --qsar-only --basis def2-TZVPP --charge 0 --spin 0</code>	Descriptor QSAR dan tabel data.
Praktik 5: Anilin dalam THF	<code>/3py "anilin anilin.xyz" --family dft --method PBE0 --basis def2-TZVP --solvent thf --solvation-model smd --qsar</code>	Efek solvent dan QSAR lanjutan.
Praktik 6: Benzaldehida	<code>/3py "benzaldehida benzaldehida.xyz" --family dft --method CAM-B3LYP --basis def2-TZVP --pcm-solvent methanol --solvation-model cpcm --addons uv-vis,nmr</code>	Solvent, UV-Vis, dan NMR jika backend tersedia.

5.4 Checklist Validasi Hasil

- Jumlah atom pada file XYZ sama dengan jumlah baris koordinat.
- Simbol atom benar dan tidak ada baris koordinat kosong.
- Charge dan spin sesuai keadaan kimia molekul.
- Metode dan basis set tertulis lengkap pada output.
- Optimasi geometri berakhir konvergen.
- Frekuensi tidak memiliki mode imajiner jika targetnya minimum lokal.
- Jika menggunakan solvent, model solvasi tersedia dan status output bukan failed.
- Jika menggunakan add-ons, setiap add-on memiliki status jelas: success, not_available, skipped, atau failed.
- QSAR descriptor yang berasal dari RDKit dibedakan dari descriptor kuantum.
- Tidak ada angka fiktif untuk properti yang tidak dihitung.
- File TXT, CSV, JSON, XYZ, dan log berhasil terbentuk.
- Geometri akhir dibuka kembali di Avogadro/Jmol untuk memastikan tidak rusak.
- Hasil dibandingkan dengan metode/basis lain jika dipakai untuk riset.

5.5 Troubleshooting dan Pengembangan Lanjutan

Gejala	Kemungkinan Penyebab	Tindakan
Script tidak berjalan	Path salah, environment belum aktif,	Aktifkan conda, cek pwd, ls, dan

	file XYZ tidak ditemukan.	nama file.
Import library gagal	Library belum terpasang atau versi tidak cocok.	Instal dari conda-forge/pip sesuai library.
SCF tidak konvergen	Geometri buruk, basis terlalu berat, charge/spin salah.	Gunakan MM/semiempirik dulu, basis ringan, atau ubah parameter SCF.
Optimasi lama	Molekul besar atau basis set berat.	Gunakan def2-SVP/STO-3G untuk awal, lalu single point basis besar.
Frekuensi imajiner	Struktur belum minimum.	Optimasi ulang dari geometri berbeda.
Solvent gagal	Backend tidak mendukung model solvent.	Gunakan model lain atau catat not_available.
NMR/Raman/NBO kosong	Add-on memerlukan backend eksternal.	Instal engine pendukung atau nonaktifkan add-on.
Output terlalu banyak	Semua add-ons diaktifkan sekaligus.	Jalankan bertahap dan pisahkan folder output.

5.6 Rekomendasi Pengembangan Fitur Komputasi

Untuk pengembangan lanjutan, fitur Komputasi dapat ditingkatkan menjadi sistem generator paket riset yang lebih lengkap. Setiap output sebaiknya memiliki metadata yang konsisten sehingga mudah dilacak dan direproduksi.

- Tambahkan template project otomatis: input, output, log, figures, tables, dan reports.
- Tambahkan validator XYZ yang menampilkan jumlah atom, formula, jarak minimum antaratom, dan peringatan valensi.
- Tambahkan pilihan workflow preset: cepat, pembelajaran, riset ringan, riset solvent, QSAR, dan spektrum.
- Tambahkan exporter laporan otomatis ke DOCX/PDF setelah perhitungan selesai.
- Tambahkan visualisasi energi, HOMO-LUMO, spektrum IR/UV-Vis, dan tabel descriptor.
- Tambahkan database hasil agar beberapa molekul dapat dibandingkan dalam satu tabel.
- Tambahkan mekanisme resume jika running terhenti.
- Tambahkan uji integritas output agar file kosong/gagal tidak dianggap sukses.

Penutup

Fitur Komputasi ChemcomPy /3py dalam kode program ini memberikan kerangka kerja yang sangat berguna untuk pembelajaran dan riset awal kimia komputasi. Dengan satu trigger, pengguna diarahkan untuk membuat struktur XYZ, script Python multi-metode, output energi, geometri, frekuensi, QSAR, solvent, add-ons, momen dipol, dan file pendukung lainnya. Namun, kualitas hasil tetap bergantung pada validasi ilmiah pengguna, kesiapan backend, kebenaran struktur awal, pemilihan metode, serta ketelitian membaca output.

Panduan ini dapat dipakai sebagai buku pegangan praktikum, modul pelatihan, atau dokumentasi fitur Komputasi. Pengguna disarankan memulai dari molekul kecil, menggunakan metode ringan,

membaca log secara teliti, kemudian meningkatkan kompleksitas ke DFT, solvent, add-ons, dan QSAR setelah environment stabil.

Lampiran A. Ringkasan Perintah Harian

```
conda activate kimia
cd ~/kimia-komputasi/nama-proyek
python nama-molekul.py --xyz-file nama-molekul.xyz --job optimize_freq --method B3LYP --
basis def2-SVP --charge 0 --spin 0 --dipole --qsar
python nama-molekul.py --xyz-file nama-molekul.xyz --job singlepoint --method B3LYP --
basis def2-TZVP --solvent water --solvation-model pcm
```

Lampiran B. Glosarium Singkat

Istilah	Makna
Add-Ons	Fitur tambahan di luar workflow utama seperti IR, UV-Vis, NMR, Raman, Thermodynamics, Delta Hf, dan NBO.
Basis Set	Himpunan fungsi matematika untuk menggambarkan orbital atom/molekul dalam perhitungan kuantum.
DFT	Density Functional Theory, metode quantum chemistry berbasis kerapatan elektron.
Hybrid DFT	DFT yang mencampurkan sebagian pertukaran Hartree-Fock, misalnya B3LYP dan PBE0.
Frekuensi Imajiner	Frekuensi negatif/imajiner yang menandakan struktur belum minimum lokal atau merupakan transition state.
Momen Dipol	Ukuran pemisahan muatan dalam molekul, biasanya dinyatakan dalam Debye.
PCM Solvent	Model pelarut implisit berbasis polarizable continuum model.
QSAR	Quantitative Structure-Activity Relationship, hubungan kuantitatif descriptor struktur dengan aktivitas/properti.
XYZ	Format koordinat molekul sederhana: jumlah atom, komentar, lalu simbol atom dan koordinat X Y Z.
Backend	Program/library yang benar-benar menghitung properti, misalnya PySCF, Psi4, RDKit, Open Babel, atau xTB.

Catatan Sumber

Dokumen ini disusun dari analisis kode HTML yang diunggah pengguna, khususnya blok Komputasi yang berisi ChemcomPy /3py, Tutorial WSL Ubuntu, Add-Ons, PCM Solvent, QSAR Properties, pilihan metode, basis set, validasi output, dan contoh trigger.