# OpenOnDemandの対話的な開発環境の利用(Jupyter, VS Code)

# 目次

- デスクトップ環境
- Jupyter Notebook
- VS Code (Code Server)

# デスクトップ環境

OpenOnDemandのデスクトップ環境では、グラフィカルなインターフェース(GUI)を通じてアプリケーションを利用できます。

# 主な用途:

- GaussViewなどのGUIベースのソフトウェアの利用
- ファイルマネージャーでの直感的なファイル操作



## セッションの起動方法

Desktop	Desktop (launched-by-ond 7ffd71d6f12a@ccfep1)	emand-af99baad-e406-49a7-9f58- 1 node   1 core   Running	
compute nodes. You will have full access to the resources	Host: >_ ccfep1.center.ims.ac.jp		
these nodes provide. This is analogous to an interactive	Created at: 2025-01-31 17:14:41 JST		
batch job.	Time Used: less than 1 minute		
Number of hours	Session ID: 04911667-409a-4f2b-8277-7396dd9be04e		
Launch	Compression	Image Quality	
* The Desktop session data for this session can be accessed	0 (low) to 9 (high)	0 (low) to 9 (high)	
under the data root directory.	Launch Desktop	View Only (Share-able Link)	

- 1. ダッシュボードから「Desktop」アイコンをクリック
- 2. セッション設定画面で以下を設定: - Number of hours: 利用時間を指定
- 3. [Launch]ボタンをクリック
- 4. セッションの準備が完了したら、[Launch Desktop]ボタンをクリック

### セッションの管理

- 実行中のデスクトップセッションは、ダッシュボードの「My Interactive Sessions」から確認できます
- セッションを終了するには、「My Interactive Sessions」から該当するセッションの「Delete」をクリックしてください。

## 注意事項

- ブラウザのタブを誤って閉じた場合でも、セッションは維持されています。「Launch Desktop」から再接続可能です
- 指定した利用時間を過ぎると、セッションは自動的に終了します
- リソースの効率的な利用のため、作業終了後は速やかにセッションを終了してください

## GaussViewの使い方

OpenOnDemandのデスクトップ環境を使用してGaussViewを起動しすることができます。

### ■ GaussViewの起動方法

- 1. OpenOnDemandのデスクトップ環境にログインします
- 2. デスクトップ上のターミナルを開きます



3. 以下のコマンドを入力して、GaussViewを起動します:

### gview6

4. GaussViewのメインウィンドウが表示されます

# Jupyter Notebook

Jupyter Notebookは、プログラムコードと実行結果、説明文を1つのノートブック形式でまとめることができる対話的な開発環境です。 Pythonをはじめとする様々なプログラミング言語に対応しています。

Chemical_Analysis_Tutorial Last Checkpoint: 56 minutes ago							
File Edit	View Run Kernel Settings Help				Т	ruste	d
<b>B</b> + %	□ □ ► ■ C → Code ∨ JupyterLab	¢	Pyth	on 3	(ipyke	ernel)	С
[18]:	<pre>plt.figure(figsize=(6, 4)) plt.plot(wavelengths, spectrum, 'b-', alpha=0.5, label='Raw Data') plt.plot(wavelengths, smoothed_spectrum, 'r-', label='Smoothed') plt.xlabel('Wavelength (nm)') plt.ylabel('Absorbance') plt.title('UV-Vis Spectrum Analysis') plt.legend() plt.gerd(() plt.gerd(() plt.show()</pre>	$\uparrow$	$\downarrow$	+	0+	Î	•
	UV-Vis Spectrum Analysis						

## 基本的な使い方

### ■ セッションの開始

- 1. OpenOndemandのダッシュボードから「Jupyter Notebook」アイコンをクリック
- 2. 設定画面で以下の項目を指定:
  - Number of hours: セッション時間(時間単位)
- 3. 「Launch」ボタンをクリック
- 準備完了後、「Connect to Jupyter」ボタンをクリック
   新しいタブでJupyter環境が開きます

## ■ 新規ノートブックの作成

- 1. 画面右上の「New」ボタンをクリック
- 2. プログラミング言語・実行環境の選択
  - 表示される一覧から利用したい環境を選択します
  - デフォルトは「Python3 (ipykernel)」: /apl/conda/20240305 のPython 3.10環境

	- New ≜Upload C
	dza_venv
	dza_python_env
La	🗖 R
	📕 JavaScript (Node.js)
	💻 dza_conda
	💻 Python 3 (ipykernel)
	s_ Terminal
	>- Console
	📃 New File

New Folder

※作成後でも画面右上のメニューから別の実行環境に切り替えることができます。

補足:ここで選択する実行環境のことを「カーネル」と呼びます。カスタマイズした環境の追加方法については、後述の「Python環境の カスタマイズ」セクションで説明します。

### ■ ノートブックの基本操作

## セルについて

ノートブックは「セル」と呼ばれるブロックの集まりで構成されています。各セルには以下の種類があります:

- コード:プログラムを書き、実行するための領域
- マークダウン:説明文を書くための領域(Markdown形式で書式設定可能)
- Raw:書式設定なしの純粋なテキスト用の領域

## 基本的な操作方法

#### セルの操作:

- 新しいセルの追加: 画面上部の [+] ボタン
- セルの種類の変更: ツールバーのドロップダウンで「Code」、「Markdown」など選択
- セルの実行: Shift + Enterキー(または実行ボタン▶)
- コード:プログラムが実行され、結果が表示されます
   マークダウン:書式が適用された文章が表示されます
- セルの選択:セルをクリック(青枠が表示されます)

# その他の操作:

- ファイルの保存: Ctrl + S または画面上部の保存アイコン
- セルの削除:セルを選択して D キーを2回押す
- 実行順序の確認: セルの左側に表示される番号で確認できます

Python環境のカスタマイズ

## ■ デフォルト環境

デフォルトで `/apl/conda/20240305` と同じPython環境 (Python 3.10) が利用可能です。 \* 当環境にインストールされているPythonパッケージがそのまま使用できます

#### ■ カスタム環境の追加手順

独自のPython環境をJupyterで使用したい場合は、シェルにて以下の手順で設定します:

### Conda環境を使用する場合:

# 環境の作成と有効化
conda create -n myenv python=3.10 # 任意の環境名とPythonバージョン
conda activate myenv

#### # 必要なパッケージのインストール

conda install ipykernel # Jupyter用カーネル conda install <必要なパッケージ名>

## # Jupyterのカーネルとして登録 python -m ipykernel install --user --name myenv --display-name "My Python Env"

#### venv環境を使用する場合:

# 環境の有効化
python -m venv myenv
source env_name/bin/activate
# ipykernelのインストール
pip install ipykernel
pip install <必要なパッケージ名>
# Jupyterのカーネルとして登録
python -m ipykernel installusername myenvdisplay-name "My Python Env"

登録が完了すると、追加したカーネルが選択可能になります。

#### 注意事項

- ブラウザのタブを誤って閉じた場合でも、「My Interactive Sessions」から再接続可能です
- 指定した利用時間を過ぎると、セッションは自動的に終了します

## VS Code (Code Server)

VS Code は多機能なコードエディタで、プログラミングやテキスト編集を効率的に行うことができます。 ファイルの編集、プログラムの実行、デバッグなど、開発に必要な機能が統合されています。

### 基本的な使い方

#### ■ セッションの開始

- 1. OpenOndemandのダッシュボードから「Code Server」アイコンをクリック
- 2. 設定画面で以下の項目を指定:
  - Number of hours: セッション時間を指定します(時間単位)
  - Working Directory:作業ディレクトリを指定します
  - \* デフォルトではホームディレクトリ(\$HOME)
  - \*「Select Path」ボタンで別のディレクトリを選択可能
  - Codeserver Version:使用するバージョンを選択
  - \*特別な理由がない限り、最新版(4.8)を推奨
- 3. 「Launch」ボタンをクリック
- 4. 準備完了後、「Connect to VS Code」ボタンをクリック→新しいタブでVS Code環境が開きます

#### ■ 基本的な画面構成

- 左側のサイドバー:ファイル一覧や拡張機能などの各種機能にアクセス
- メインエディタ:ファイルの編集領域
- 下部パネル:ターミナル、出力、デバッグ情報などを表示



- ファイルを開く:
  - サイドバーのファイルエクスプローラーからファイルをクリック
  - または Ctrl + O で開くファイルを選択
- ファイルの保存: Ctrl + S
- 新規ファイル作成: Ctrl + K N

# ■ ターミナルの利用

ターミナルを利用してコマンドライン操作が可能です:

- ターミナルを開く:
- メニューから [Terminal] → [New Terminal] を選択
- または Ctrl + ` (バックウォート)
- 複数のターミナル:
  - 新規ターミナル : [+]ボタン
  - ターミナルの分割 : [Split Terminal]アイコン
  - ターミナルの切り替え:ターミナルパネル右のメニューを使用

## ■ 拡張機能

VS Codeでは、必要に応じて拡張機能をインストールできます。 インストールした拡張機能は、セッションを終了した後も保持されます。

- 1. サイドバーにある Extensions アイコン (四角が4つ重なったアイコン)をクリック
- 2. 検索バーに、必要な拡張機能の名前やキーワードを入力
- 3. 目的の拡張機能を見つけたら、[Install]をクリック

## ■ 便利な機能

- コマンドパレット: Ctrl + Shift + P
- VS Codeの全機能に素早くアクセス
- ファイル内検索 : Ctrl + F
- プロジェクト内検索: Ctrl + Shift + F

## 注意事項

- ブラウザのタブを誤って閉じた場合でも、「My Interactive Sessions」から再接続可能です
- 指定した利用時間を過ぎると、セッションは自動的に終了します