

Prerequisite

-- Setup for Windows

チューターおよび予行日でのセットアップ

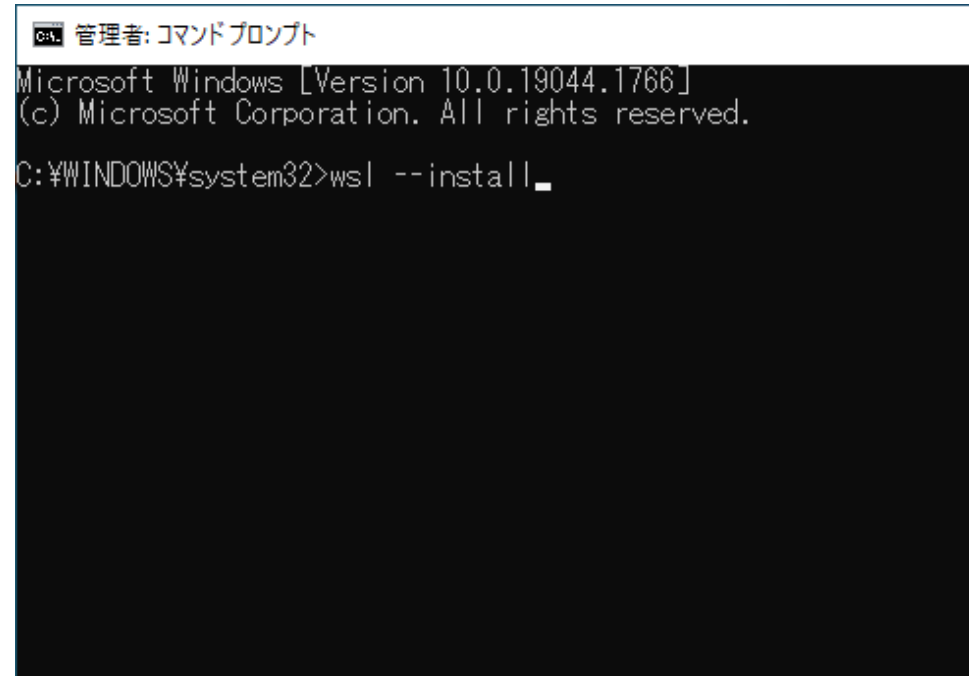
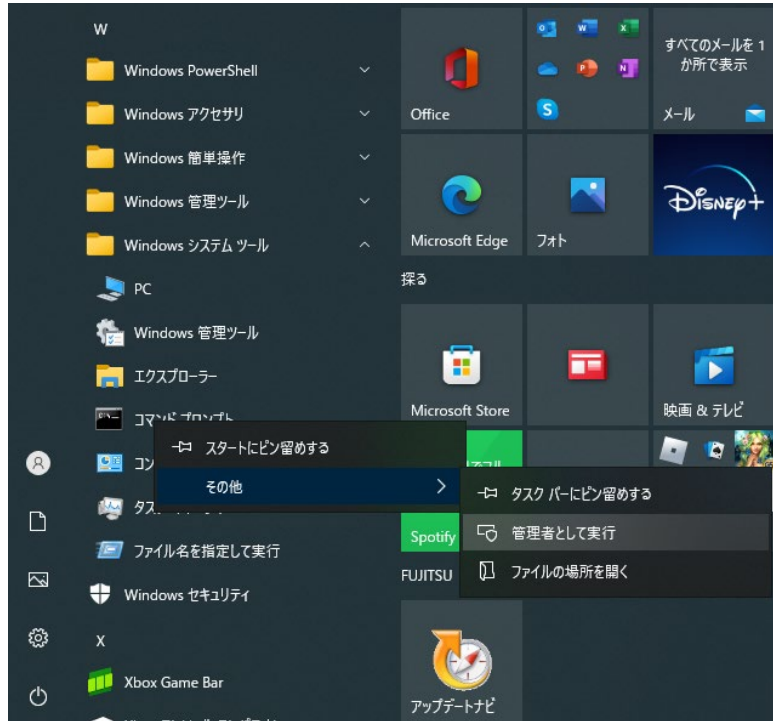
概要

- 本講座ではスーパーコンピューターシステムを利用するが、その準備としてローカル環境でUnixコマンドに馴染んでもらう必要がある。
- Windows PC上に下記三つのソフトウェアをインストールし、UnixコマンドおよびEditorの操作を練習する。
 - Windows Subsystem for Linuxの有効化およびUbuntuのインストール
Unixコマンドの実行環境
 - Windows Terminalのインストール
Unixコマンドの実行環境（2）
 - Visual Studio Codeのインストール
コードを編集するためのEditor

Windows Subsystem for Linuxの有効化 およびUbuntuのインストール

- <https://docs.microsoft.com/en-us/windows/wsl/install> を参照すること.
- 上記サイトに書いてある手順はwslがインストールしたことない場合に適用する. wslすでにインストールしているなら「Windows Terminalのインストール」まで飛ばしてください.
- p4-5は上記サイトをまとめた内容で、必要であればご参考ください.

Windows Subsystem for Linuxの有効化 およびUbuntuのインストール

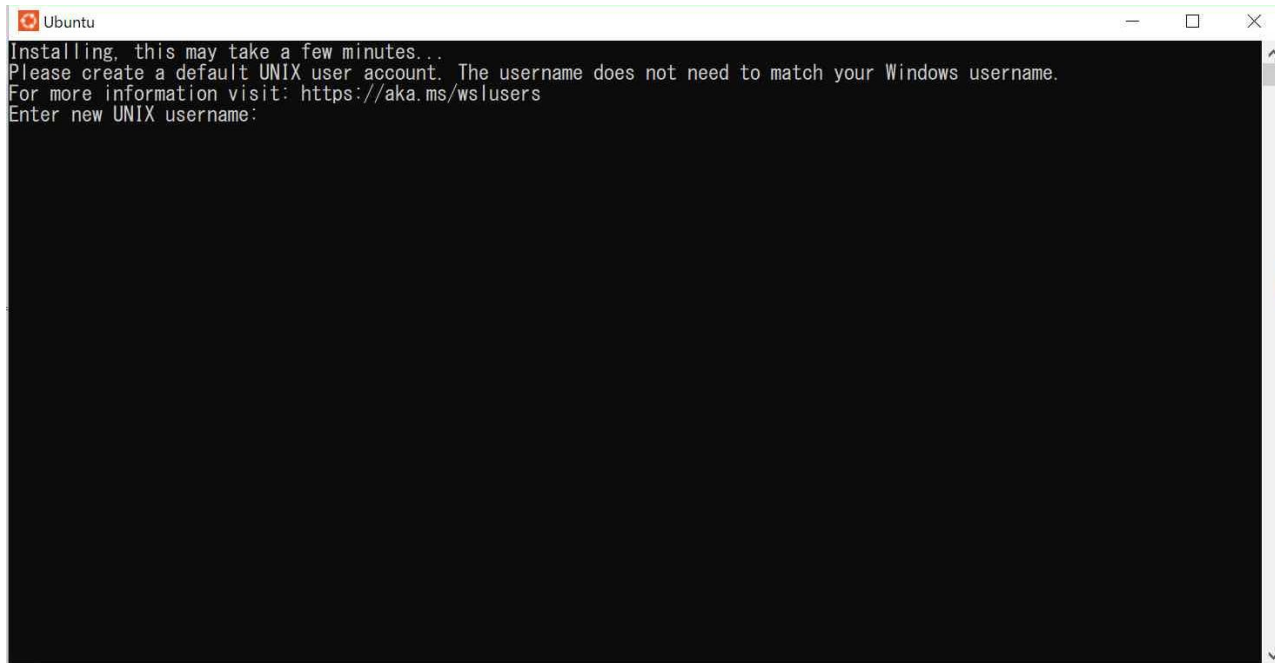


スタートメニュー → Windows システムツール → コマンド
プロンプトを右クリック，管理者として実行する

コマンドプロンプトに「wsl --install」を入力し，実行する

再起動する

Windows Subsystem for Linuxの有効化 およびUbuntuのインストール



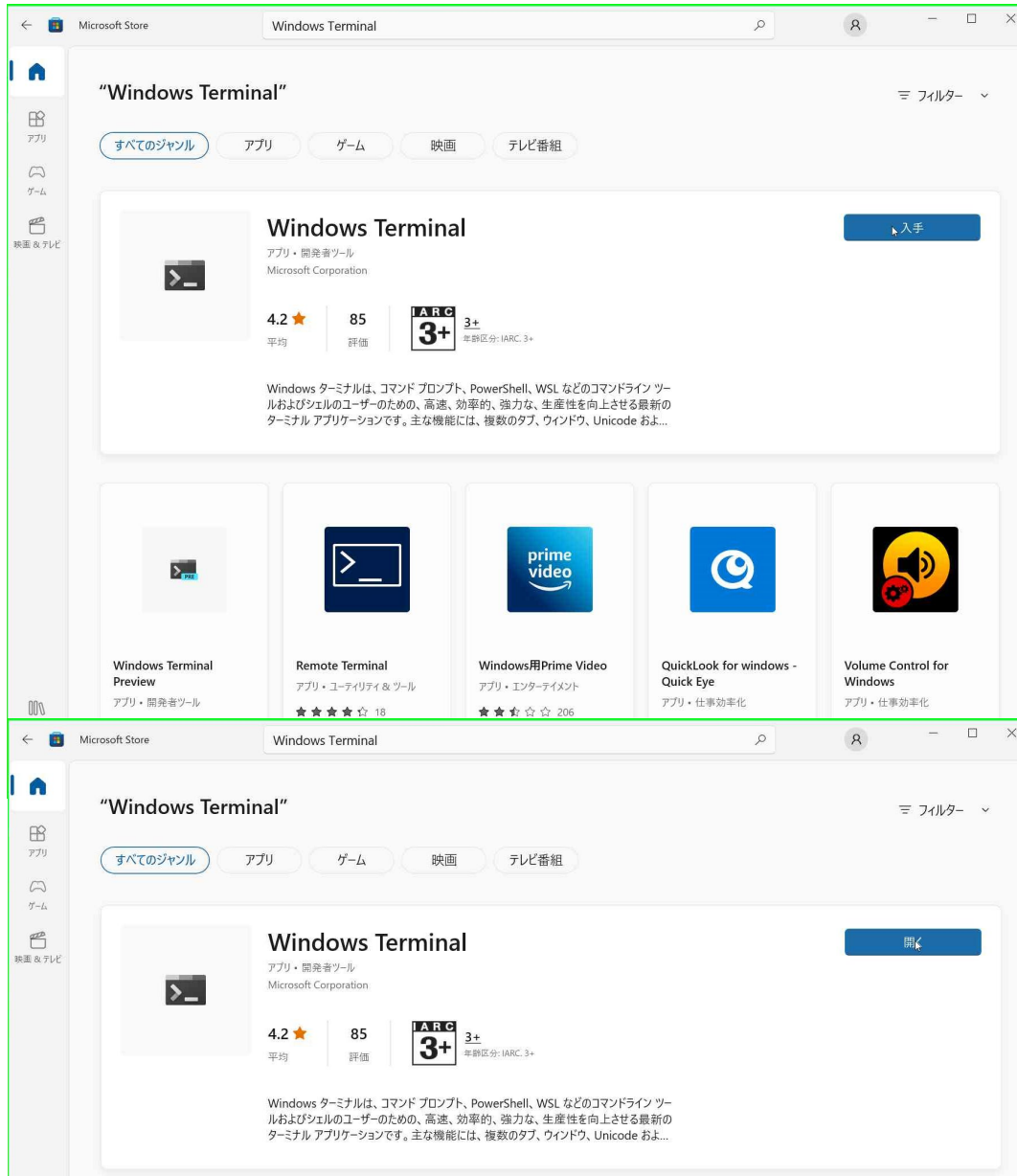
再起動すると，Ubuntuのインストール画面が自動的に立ち上がり，usernameおよびpasswordの入力が求められる．

以上Windows Subsystem for Linuxの有効化およびUbuntuのインストールが終了．

Windows Terminalのインストール

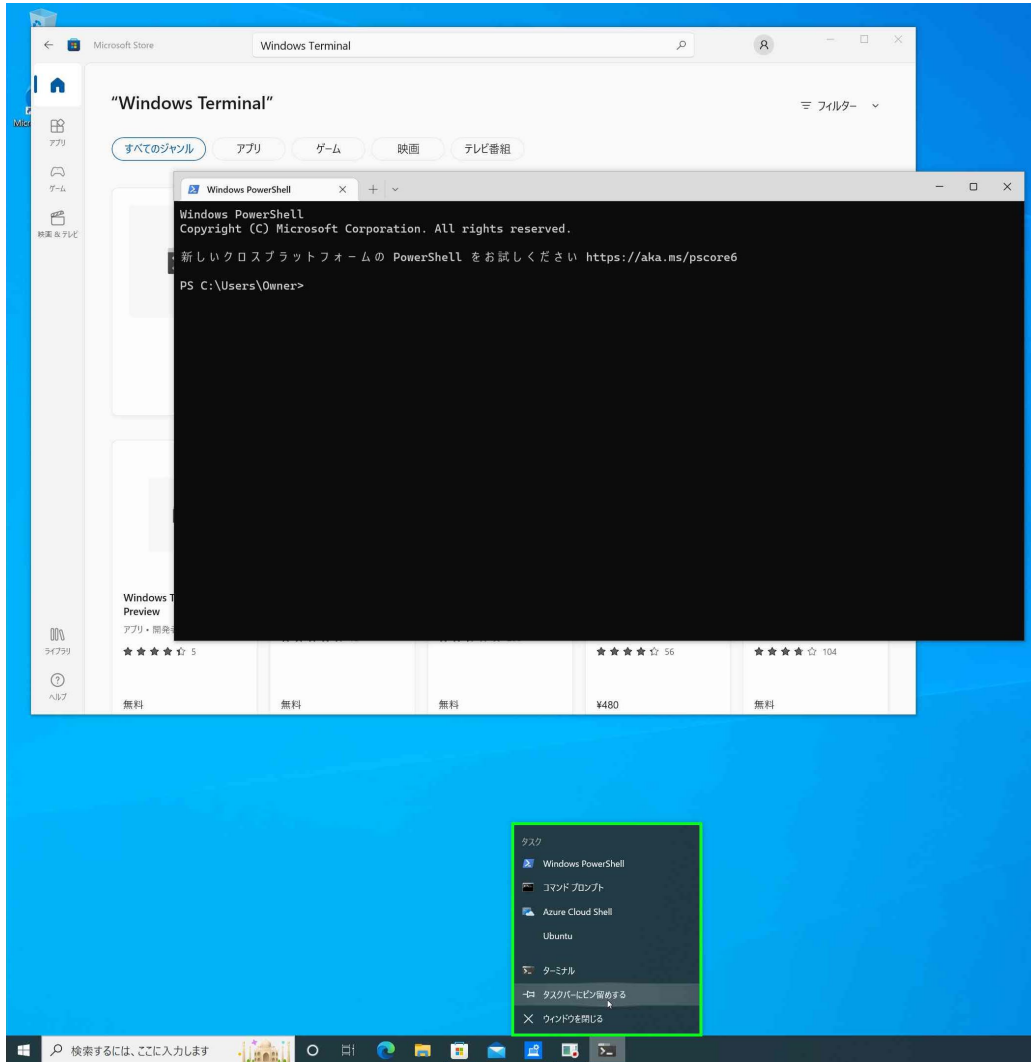
- Windows には cmd, Windows PowerShell, wsl など様々なCUIが存在する.
(CUI: Command Line Interface ; マウスで操作せず、キーボードでコマンドを入力することで操作)
- Windows Terminalからは上記すべてのCUIにアクセスできる. また履歴, フォント, 色付けなどを設定できるので, 利用しやすい環境である.

Windows Terminalのインストール



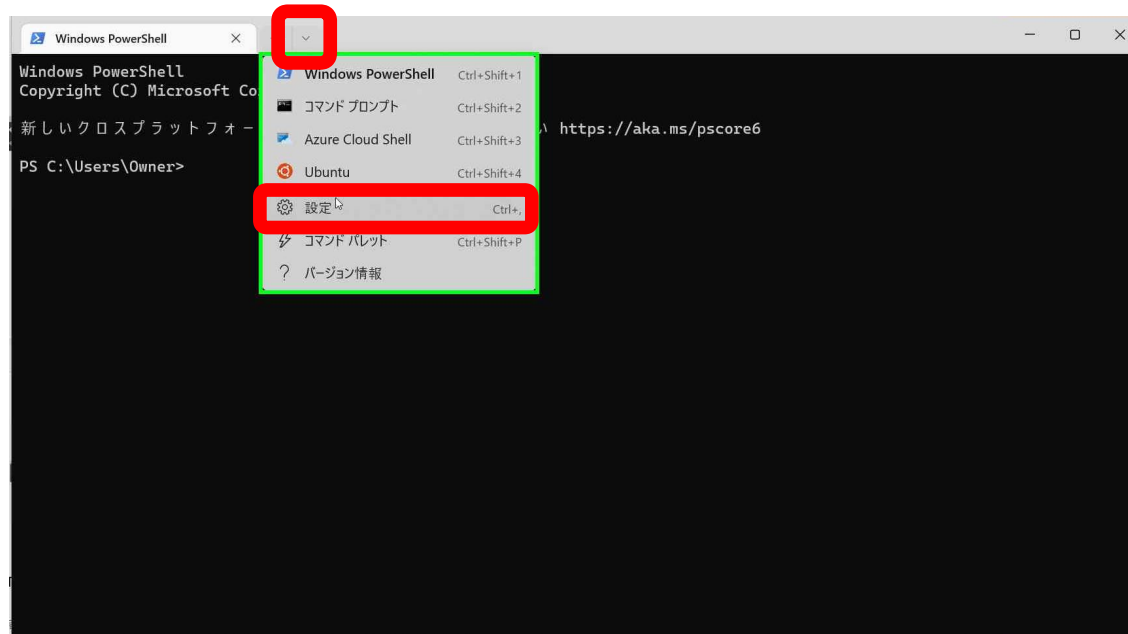
- Ubuntuインストールの時と同じく Microsoft Storeを立ち上げる
- 検索欄から「Windows Terminal」と検索（大文字に注意）
- 「入手」をクリックし、インストールが開始
- インストールが完了すると、「開く」をクリックし、Windows Terminalを開く

Windows Terminalのインストール



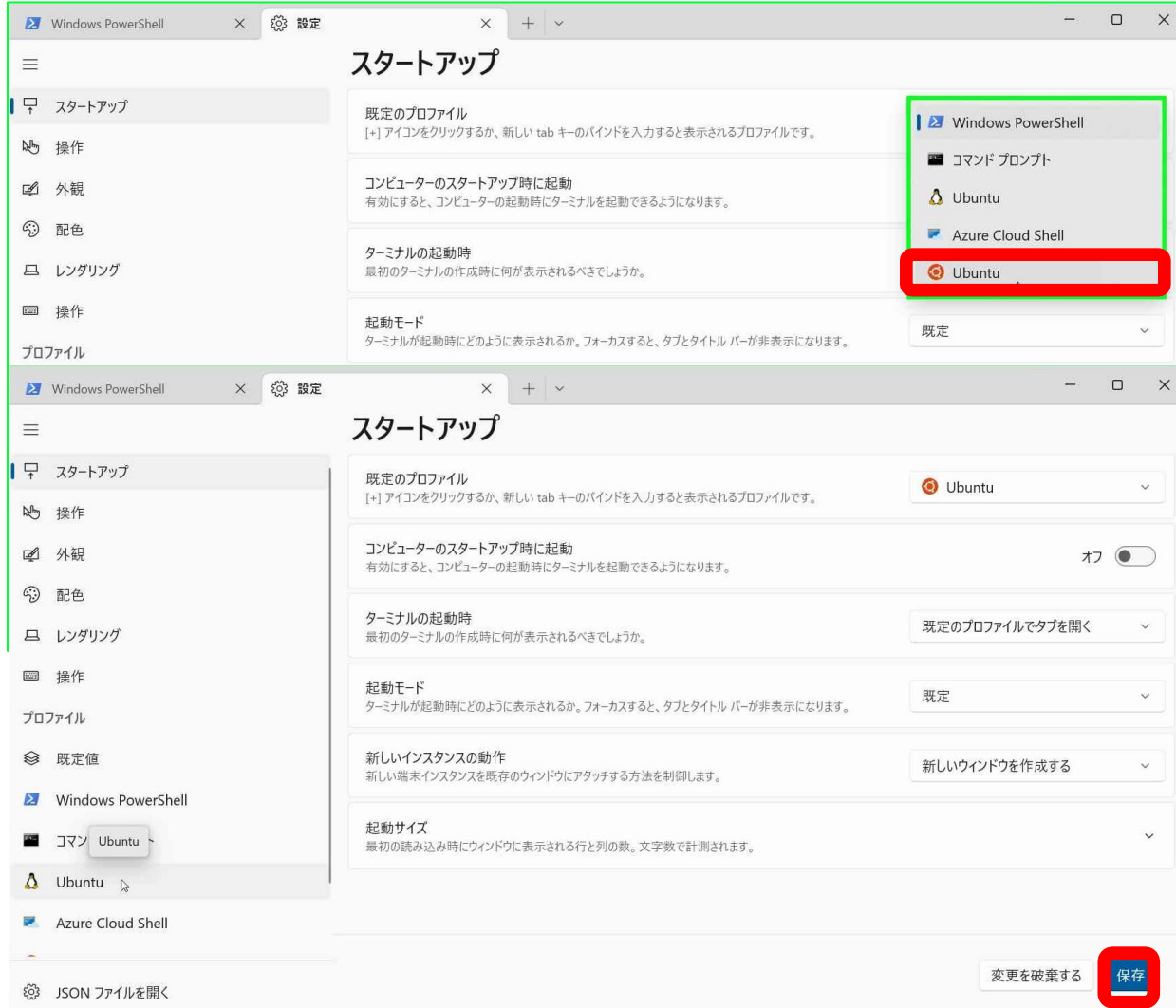
Windows Terminalを立ち上げたら、
タスクバーに固定してください。
(予行日のセットアップ)

Windows Terminalのセットアップ



Windows Terminalを起動すると、Windows PowerShellが起動する。
Windows PowerShellの右側「V」をクリックし、設定を選択。

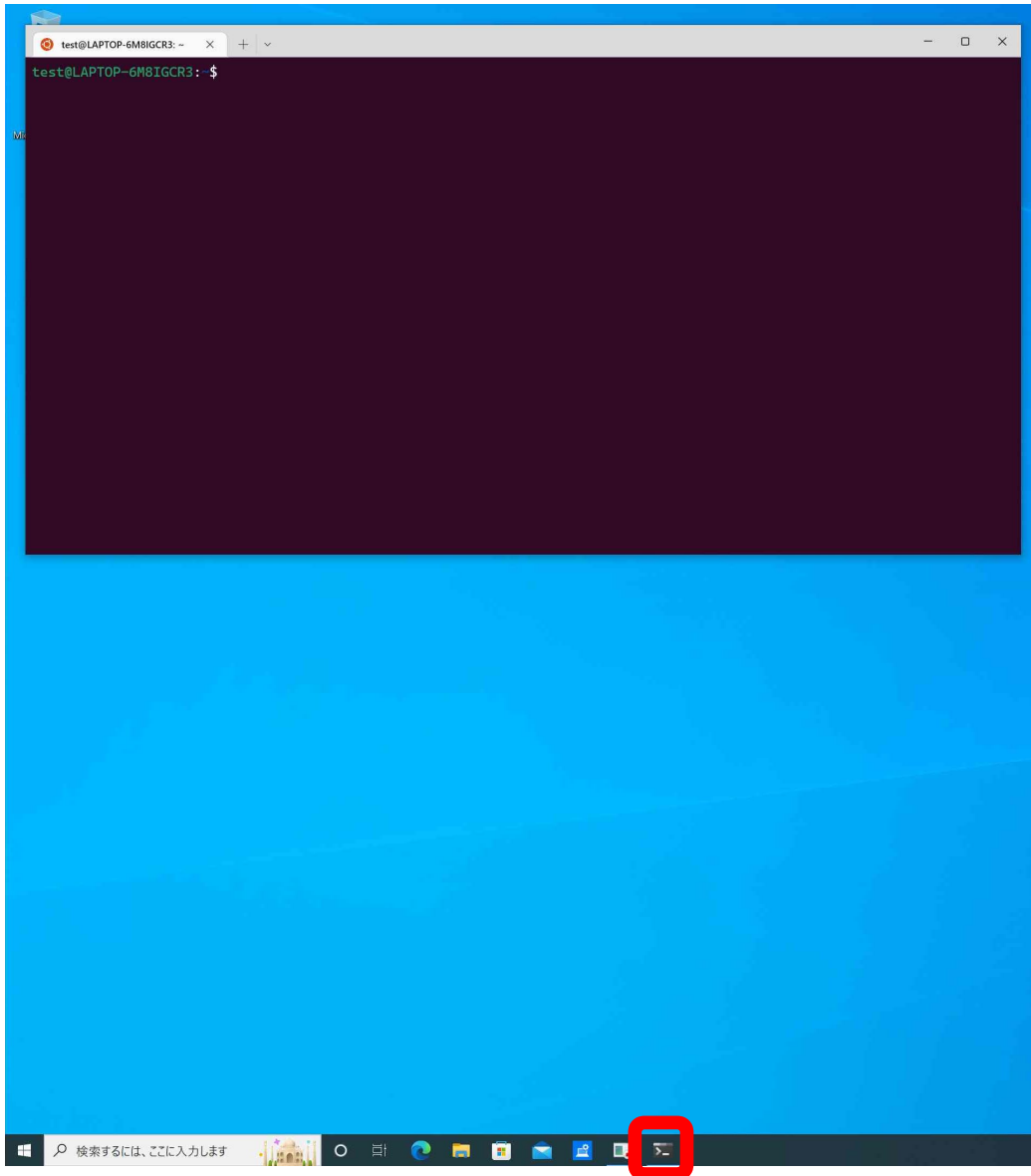
Windows Terminalのセットアップ



既定のプロファイルにUbuntu
を選択する。

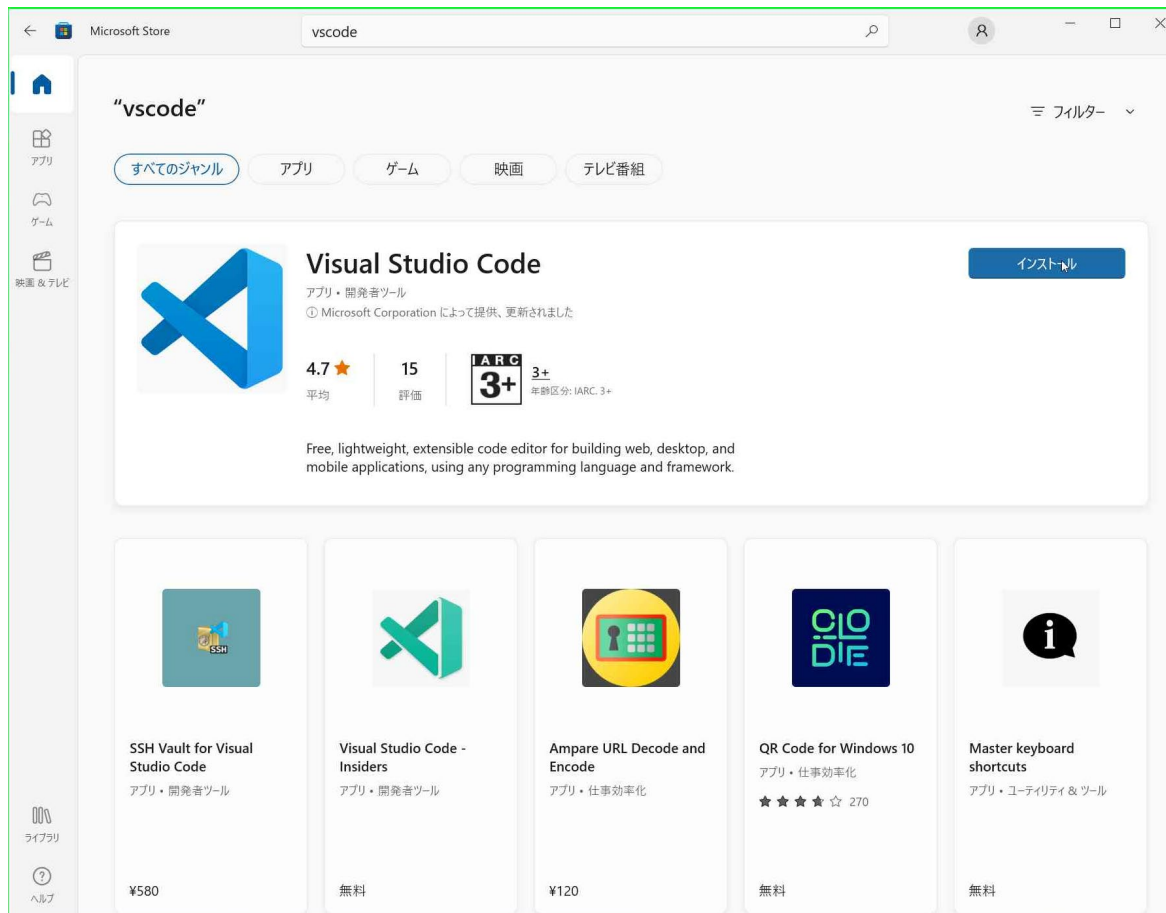
設定を保存して、Windows
Terminalを終了する。

Windows Terminalのセットアップ



これ以降，タスクバーにあるTerminalアイコンをクリックすると，Ubuntuが立ち上がる．

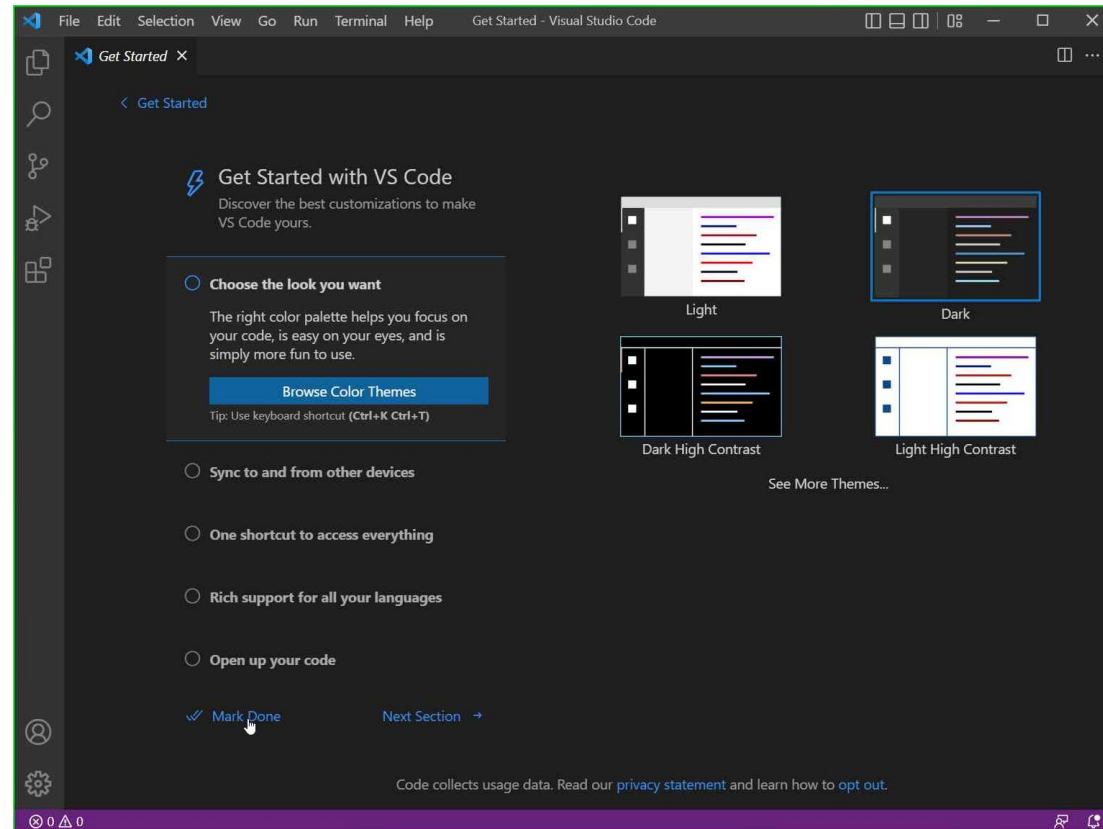
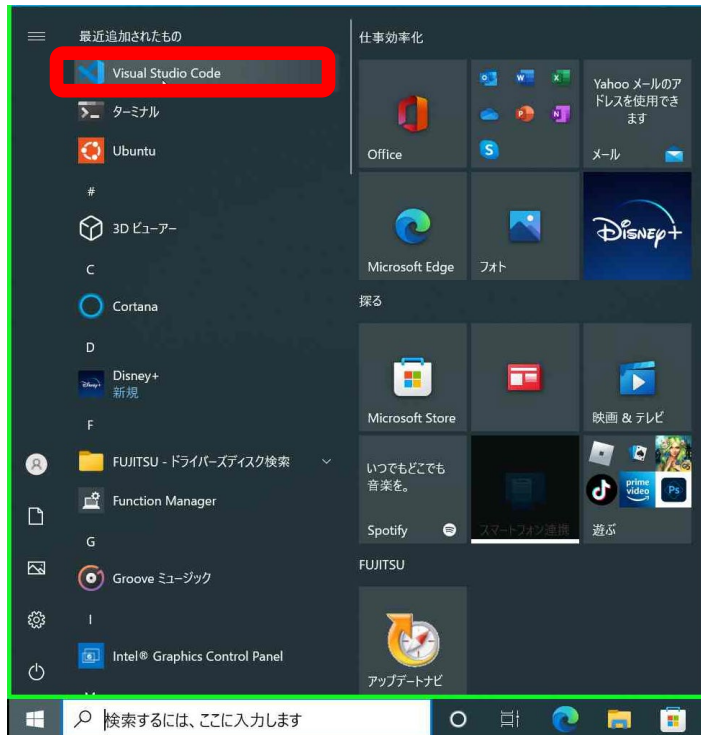
Visual Studio Codeのインストール



Microsoft Storeでvscodeを検索

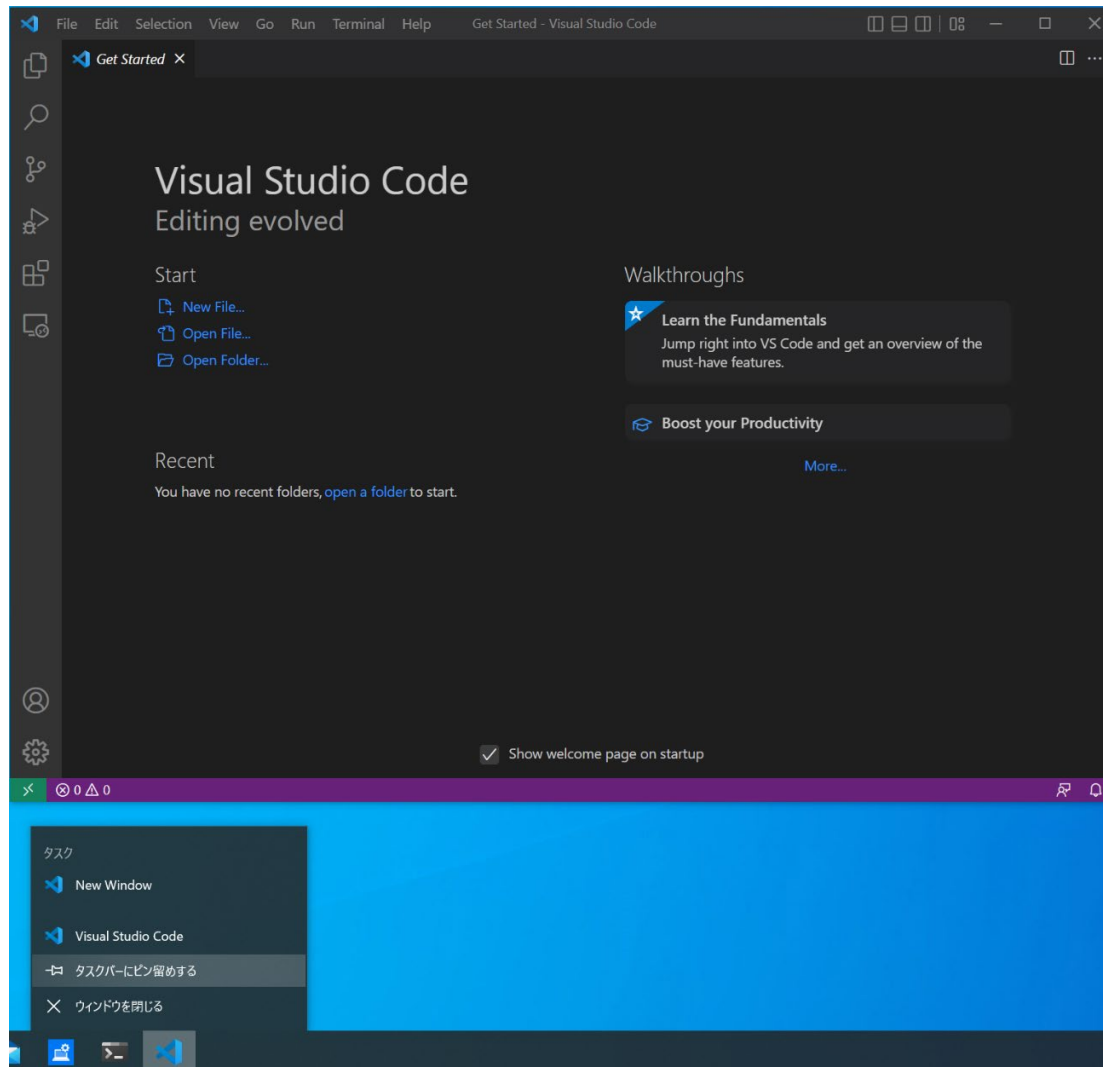
Visual Studio Codeをインストール

Visual Studio Codeのインストール



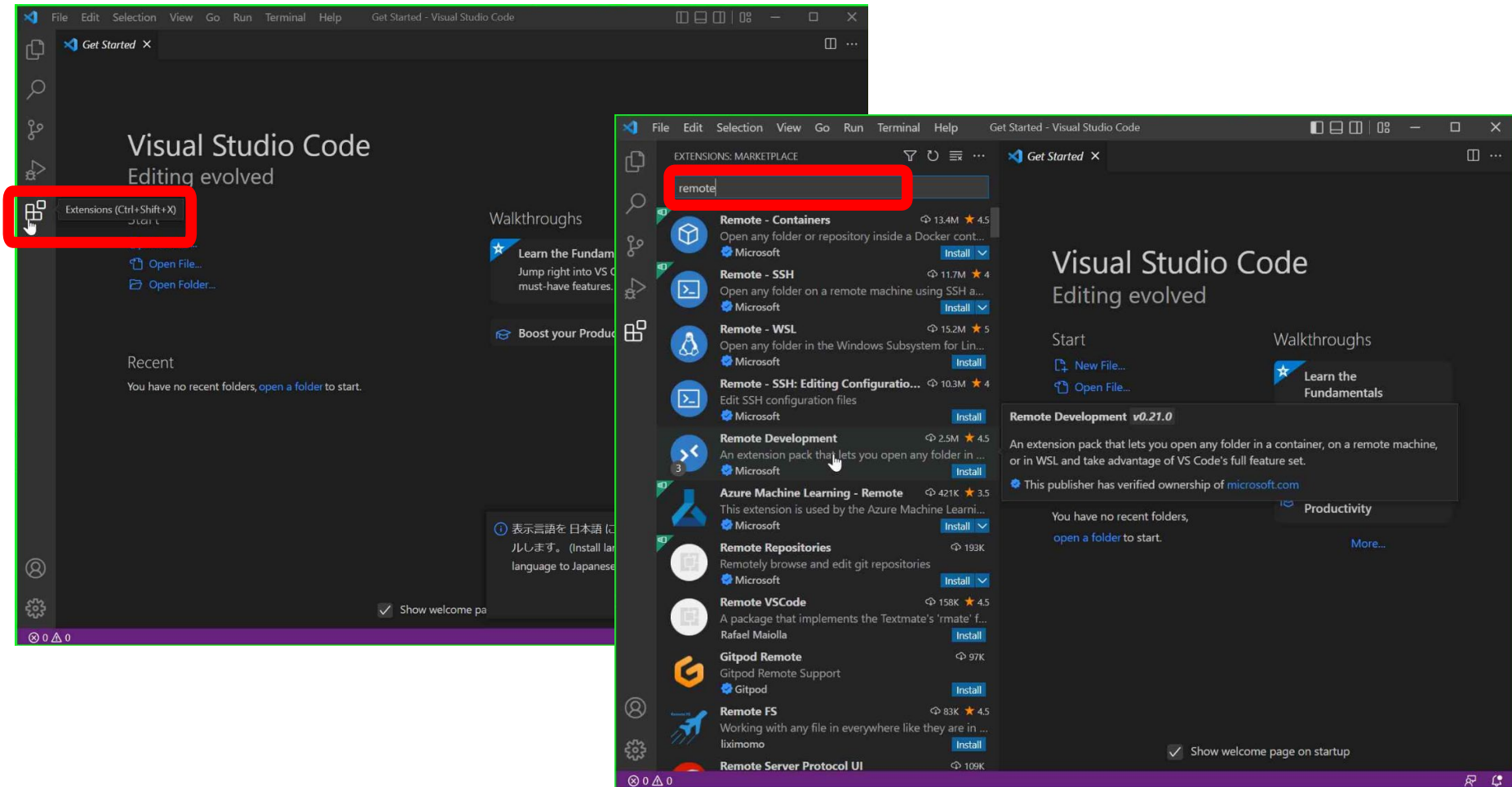
スタートメニューからVisual Studio Codeを起動する

Visual Studio Codeのインストール



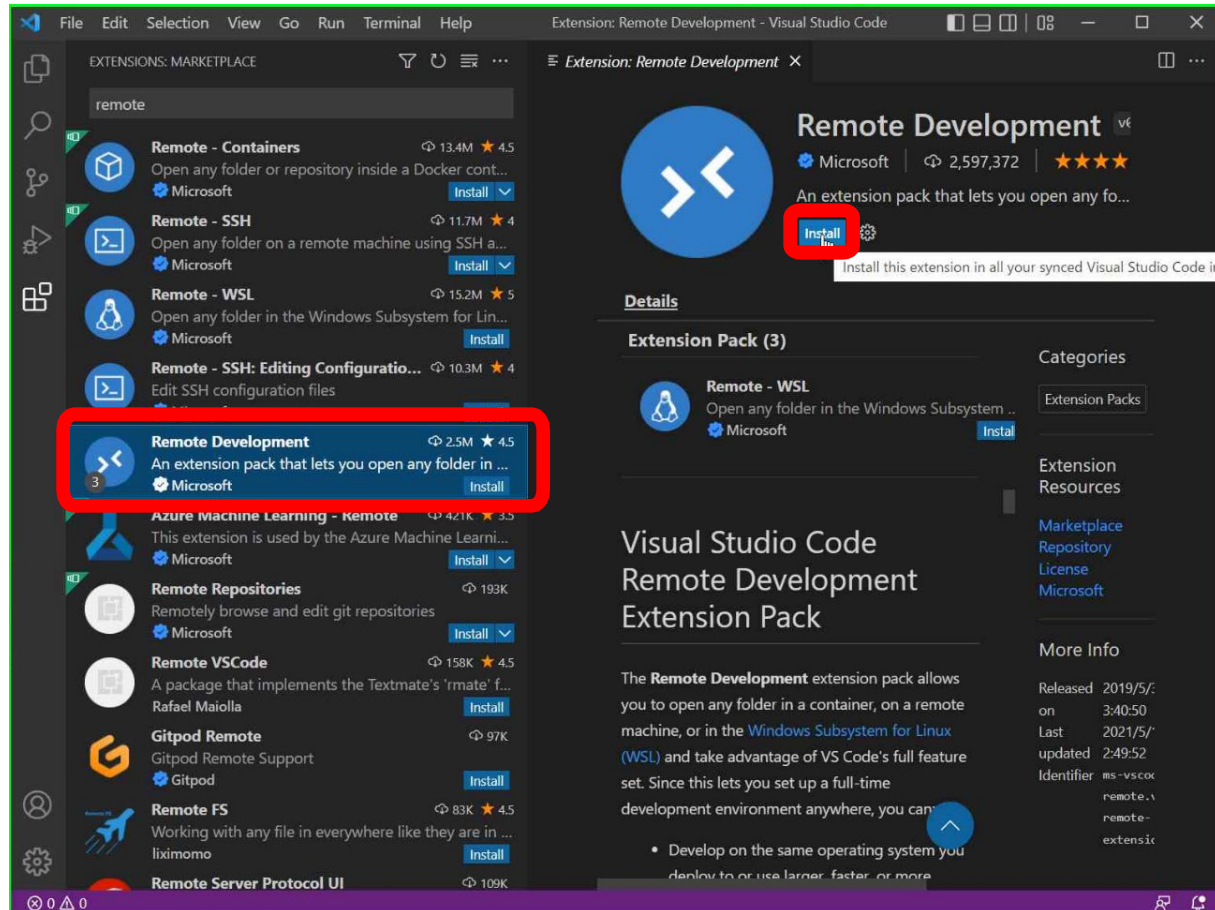
タスクバーにピン留めしてください

Visual Studio Codeのインストール



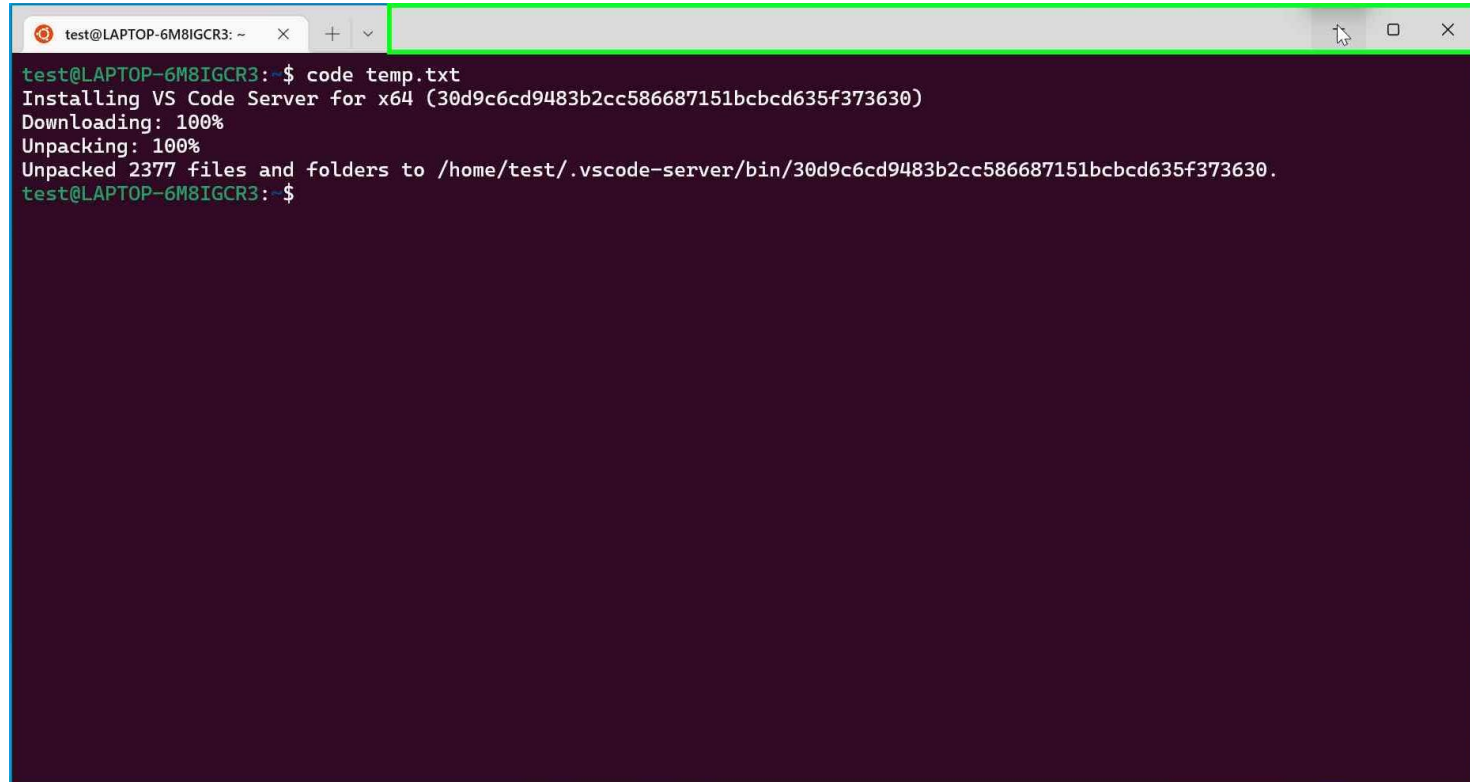
左側にあるExtensionのマークをクリックし、Extension Marketplaceに入る
Extension Marketplaceでremoteを検索する。

Visual Studio Codeのインストール



Remote Developmentを選択し、インストールする。

Visual Studio Codeのテスト

A screenshot of a terminal window with a dark purple background. The window title bar shows 'test@LAPTOP-6M8IGCR3: ~' and standard window controls. The terminal text shows the execution of 'code temp.txt', which triggers the installation of the VS Code Server for x64. The output indicates 100% download and unpacking, followed by the unpacking of 2377 files and folders to a specific path in the .vscode-server directory.

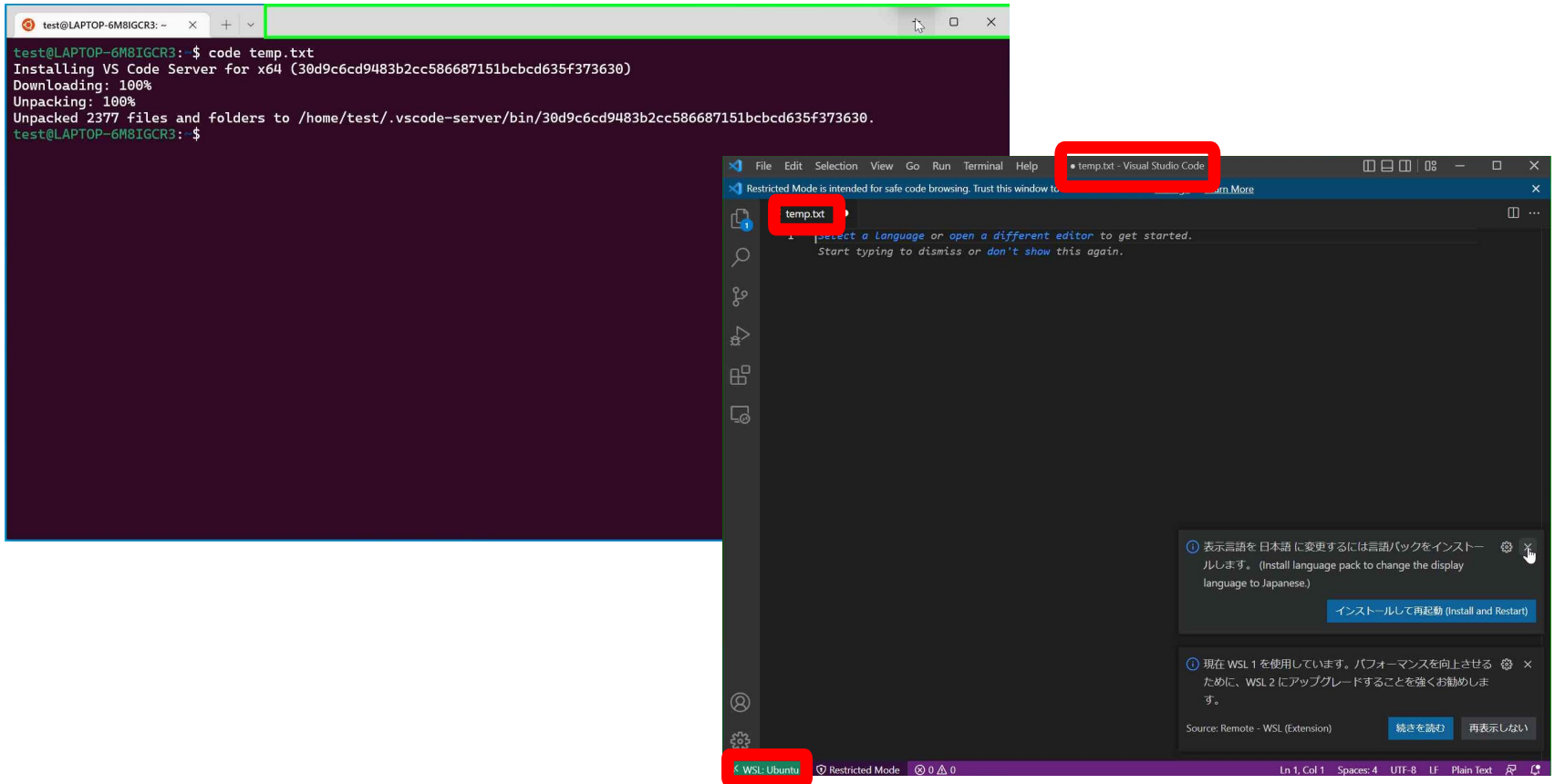
```
test@LAPTOP-6M8IGCR3: ~  
test@LAPTOP-6M8IGCR3:~$ code temp.txt  
Installing VS Code Server for x64 (30d9c6cd9483b2cc586687151bcbcd635f373630)  
Downloading: 100%  
Unpacking: 100%  
Unpacked 2377 files and folders to /home/test/.vscode-server/bin/30d9c6cd9483b2cc586687151bcbcd635f373630.  
test@LAPTOP-6M8IGCR3:~$
```

Visual Studio Codeを一回閉じ、Windows Terminalを立ち上げ、
下記コマンドを入力し

```
code temp.txt
```

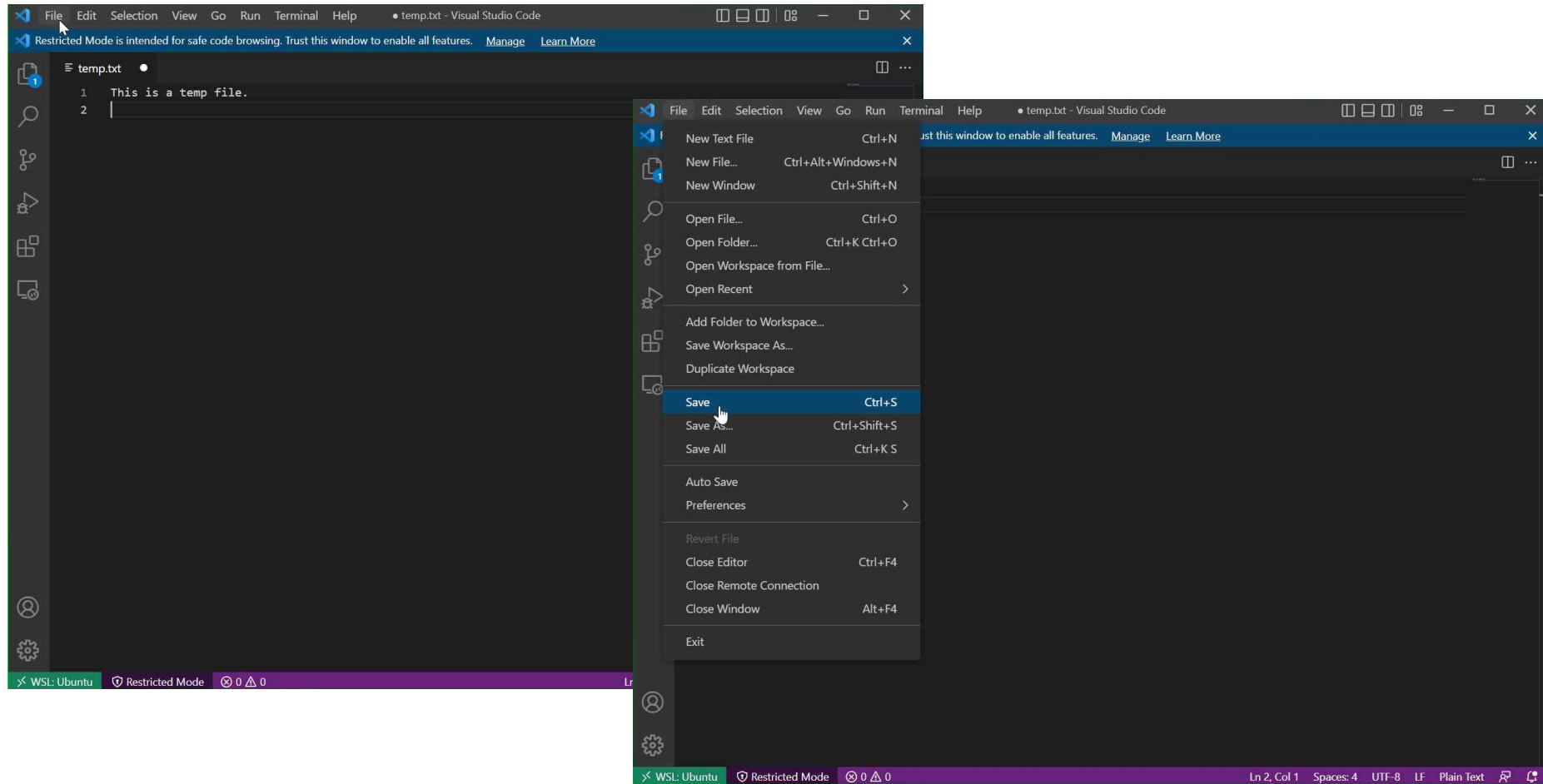
上記画面が現れ、その後Visual Studio Codeが起動される。

Visual Studio Codeのテスト



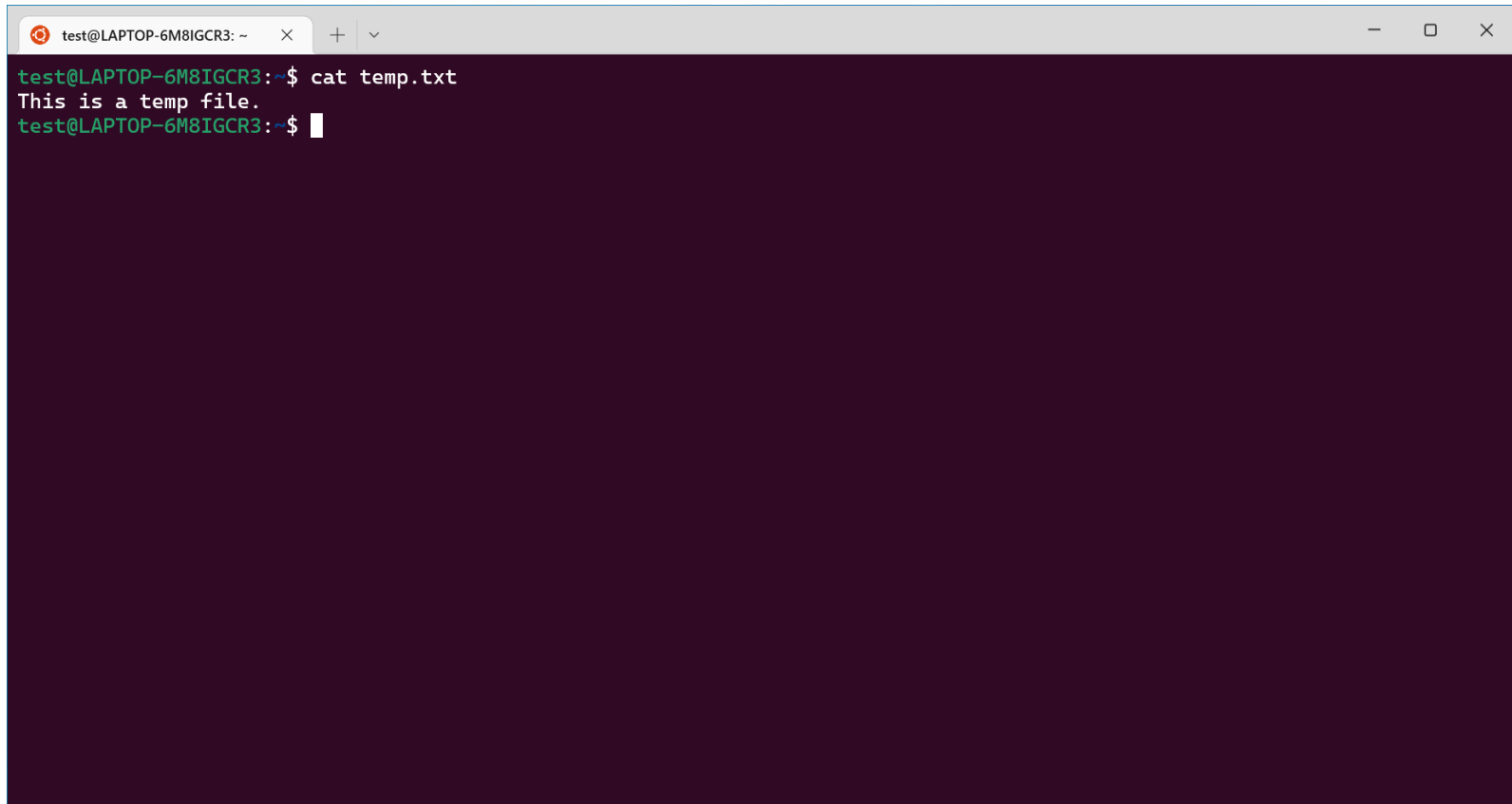
起動したVisual Studio CodeのTitleにtemp.txt含まれていることを確認．さらに左下に「WSL: Ubuntu」が表示されている．

Visual Studio Codeのテスト



適当に「This is a temp file.」と入力し、「File > Save」で保存する。

Visual Studio Codeのテスト

A screenshot of a Windows Terminal window. The title bar shows 'test@LAPTOP-6M8IGCR3: ~' and standard window controls. The terminal has a dark purple background. The command 'cat temp.txt' has been executed, and the output 'This is a temp file.' is displayed. The prompt 'test@LAPTOP-6M8IGCR3:~\$' is visible at the bottom.

```
test@LAPTOP-6M8IGCR3:~$ cat temp.txt
This is a temp file.
test@LAPTOP-6M8IGCR3:~$
```

Visual Studio Codeを閉じ、Windows Terminalを立ち上げる。
「cat temp.txt」を実行し、「This is a temp file.」と出力されたら
セットアップ成功。

コマンドのインストール

下記コマンドを実行してください

```
sudo apt update
```

```
sudo apt upgrade
```

```
wget https://bootstrap.pypa.io/get-pip.py
```

```
python3 get-pip.py
```

```
export PATH="$HOME/.local/bin:$PATH"
```

```
echo 'export PATH="$HOME/.local/bin:$PATH"' >> ~/.bashrc
```

```
pip3 install --user tldr
```

tldr python

上記コマンド実行後に右側のようにpythonの使用例が画面に表示される。
以上でコマンドのインストールが成功。

```
test@DESKTOP-7D4OT07: ~ + - v
```

```
WARNING: The script wheel is installed in '/home/test/.local/bin' which is
Consider adding this directory to PATH or, if you prefer to suppress this
on.
WARNING: The scripts pip, pip3, pip3.10 and pip3.8 are installed in '/home
TH.
Consider adding this directory to PATH or, if you prefer to suppress this
on.
Successfully installed pip-22.1.2 wheel-0.37.1
Collecting tldr
  Downloading tldr-3.1.0-py3-none-any.whl (11 kB)
Collecting shtab>=1.3.10
  Downloading shtab-1.5.5-py2.py3-none-any.whl (13 kB)
Requirement already satisfied: colorama in /usr/lib/python3/dist-packages (f
Collecting termcolor
  Downloading termcolor-1.1.0.tar.gz (3.9 kB)
  Preparing metadata (setup.py) ... done
Building wheels for collected packages: termcolor
  Building wheel for termcolor (setup.py) ... done
  Created wheel for termcolor: filename=termcolor-1.1.0-py3-none-any.whl siz
c1a5df757618729e260f71729b95ed1a1387b32eff
  Stored in directory: /home/test/.cache/pip/wheels/a0/16/9c/5473df82468f958
Successfully built termcolor
Installing collected packages: termcolor, shtab, tldr
Successfully installed shtab-1.5.5 termcolor-1.1.0 tldr-3.1.0
```

```
python
```

```
Python language interpreter.
More information: https://www.python.org.
```

- Start a REPL (interactive shell):
`python`
- Execute a specific Python file:
`python path/to/file.py`
- Execute a specific Python file and start a REPL:
`python -i path/to/file.py`
- Execute a Python expression:
`python -c "expression"`
- Run the script of the specified library module:
`python -m module arguments`
- Install a package using 'pip':
`python -m pip install package_name`
- Interactively debug a Python script:
`python -m pdb path/to/file.py`
- Start the built-in HTTP server on port 8000 in the current directory:
`python -m http.server`