Setting Up a Mojo Development Environment

This repository provides step-by-step instructions to set up **Mojo**, a cutting-edge programming language, in a Windows Subsystem for Linux (WSL2) environment using **Miniconda**. Whether you're a beginner or experienced developer, this guide ensures a seamless experience in building and using Mojo.

Table of Contents

\newpage

Prerequisites

Before starting, ensure you have thok e following:

1. **Windows 11** with WSL2 installed.

2. **Ubuntu (22.04 recommended)** running in WSL2.
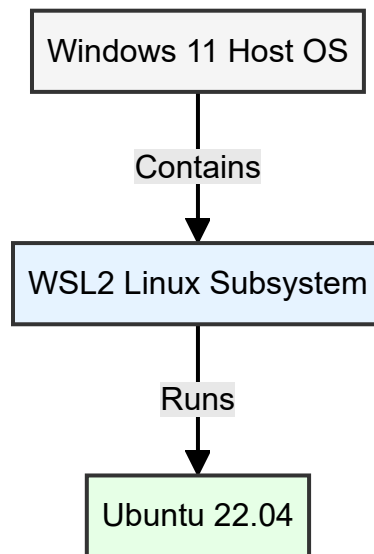
3. **Internet Access**.

\newpage

Steps to Set Up the Mojo Environment

Install WSL2 and Update Ubuntu

Ensure WSL2 and Ubuntu are installed and up to date:

```
sudo apt update && sudo apt upgrade -y
```

Diagram: System Overview

```
┌─────────────────────────┐
│    Windows 11 Host OS    │
└─────────────────────────┘
            │
        Contains
            │
            ▼
┌─────────────────────────┐
│  WSL2 Linux Subsystem   │
└─────────────────────────┘
            │
          Runs
            │
            ▼
      ┌──────────────┐
      │ Ubuntu 22.04 │
      └──────────────┘
```

Install Miniconda

Miniconda is used to manage the Mojo environment.

Download and Install Miniconda

1. Download the Miniconda installer:

   ```
   wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
   ```

2. Run the installer:

   ```
   bash Miniconda3-latest-Linux-x86_64.sh
   ```

3. Follow the prompts:
   - Accept the license by typing `yes`.
   - Confirm the installation directory (recommended: `/home/<username>/miniconda3`).
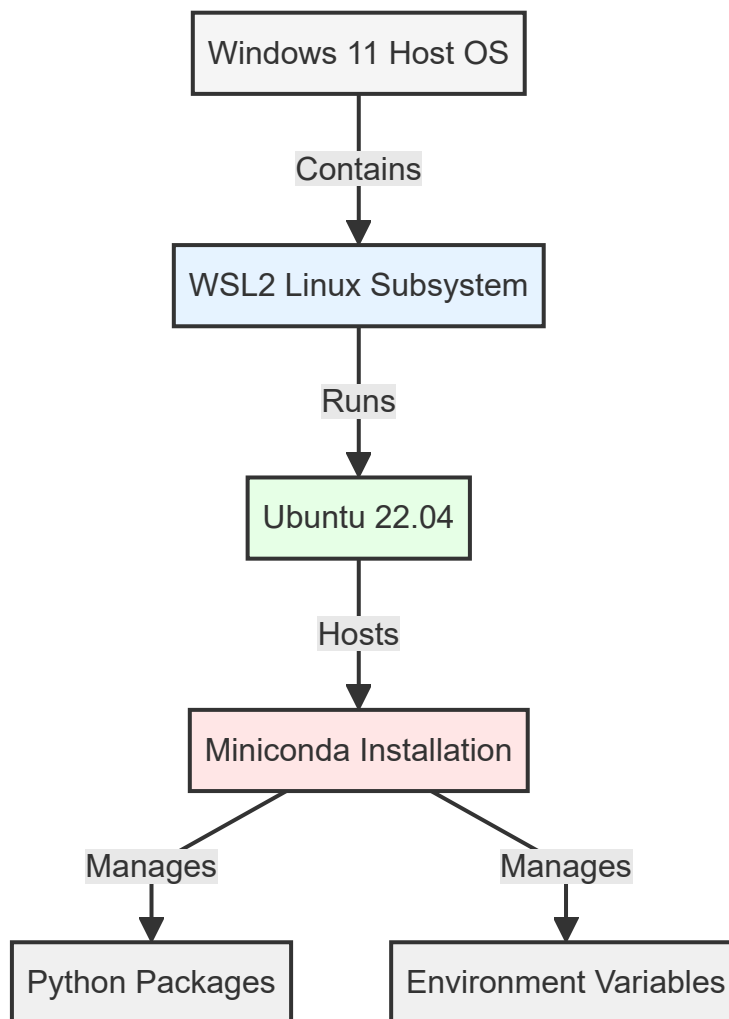   - Do not enable auto-activation unless desired.

4. Add Miniconda to your PATH:

   ```
   export PATH=~/miniconda3/bin:$PATH
   ```

5. Verify the installation:

   ```
   conda --version
   ```

Diagram: WSL2 with Miniconda Installed

```
          ┌─────────────────────┐
          │  Windows 11 Host OS  │
          └─────────────────────┘
                     │
                  Contains
                     ▼
          ┌─────────────────────┐
          │ WSL2 Linux Subsystem │
          └─────────────────────┘
                     │
                   Runs
                     ▼
          ┌─────────────────────┐
          │    Ubuntu 22.04      │
          └─────────────────────┘
                     │
                   Hosts
                     ▼
          ┌─────────────────────┐
          │ Miniconda Installation│
          └─────────────────────┘
              │              │
           Manages        Manages
              ▼              ▼
   ┌──────────────────┐ ┌──────────────────────┐
   │ Python Packages  │ │ Environment Variables │
   └──────────────────┘ └──────────────────────┘
```

\newpage

Set Up the Mojo Environment

Add Required Channels

1. Configure Conda to use the necessary repositories:

```
conda config --add channels conda-forge
conda config --add channels https://conda.modular.com/max-nightly
```

Create a Mojo Environment

2. Create a Conda environment for Mojo:

```
conda create -n mojo-env max -y
```

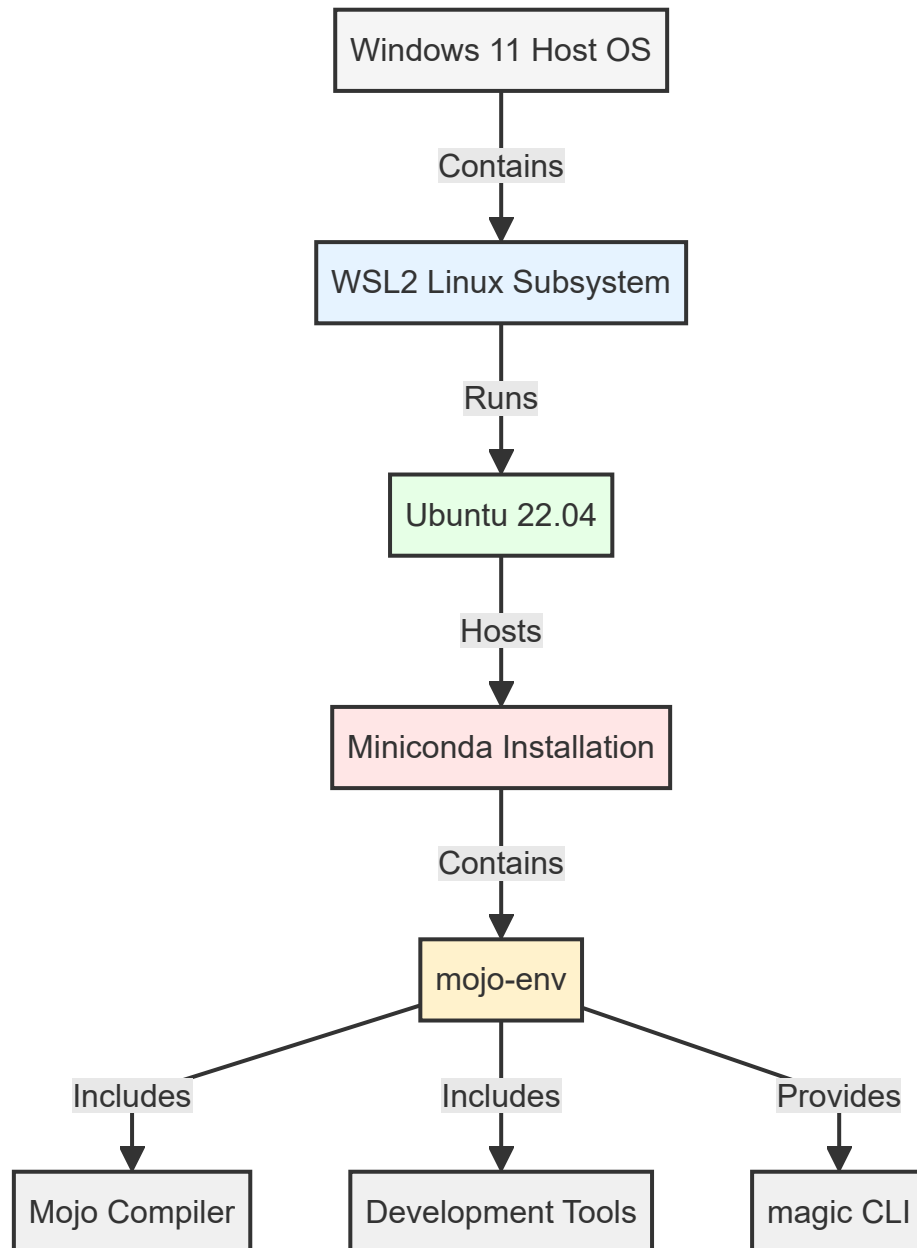3. Activate the Mojo environment:

```
conda activate mojo-env
```

4. Verify the `magic` CLI is installed:

```
magic --version
```

- You should see the version number, confirming a successful setup.

Diagram: Conda Environment for Mojo

```mermaid
flowchart TD
    A[Windows 11 Host OS] -->|Contains| B[WSL2 Linux Subsystem]
    B -->|Runs| C[Ubuntu 22.04]
    C -->|Hosts| D[Miniconda Installation]
    D -->|Contains| E[mojo-env]
    E -->|Includes| F[Mojo Compiler]
    E -->|Includes| G[Development Tools]
    E -->|Provides| H[magic CLI]
```

\newpage

Using the Mojo Environment

Activate the Environment

To enter the Mojo environment:

```
conda activate mojo-env
```

Exit the Environment

To deactivate the Mojo environment and return to the base system:

```
conda deactivate
```

Exit WSL2

To completely leave the WSL2 environment:

```
exit
```

\newpage

Additional Notes

- **Environment Isolation:** The Mojo environment is isolated from other Conda environments, ensuring dependencies do not conflict.

- **Auto-Activation:** If you want the environment to activate automatically, run:

  ```
  conda config --set auto_activate_base true
  ```

- **Uninstall Miniconda:** To completely remove Miniconda, delete its directory:

  ```
  rm -rf ~/miniconda3
  ```

This guide should help you set up and manage a reliable Mojo development environment.