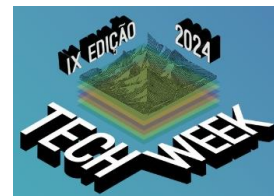


Utilizando ferramentas nativas de virtualização no Windows para o desenvolvimento e execução de aplicações Linux.

Dr. Antonio Mendes de Oliveira Neto
IFSP Campus Cubatão

e-mail: anmendes@ifsp.edu.br

AD: *Slide* de fundo branco. No alto, sobre tarja verde, TÍTULO DO TRABALHO, em branco. Abaixo, nome dos autores e instituição, em preto. No canto inferior direito, marca do logo da 9ª Techweek e do IFSP Campus Cubatão.



Virtualização

A virtualização é um processo que permite a um computador compartilhar seus recursos de hardware com vários ambientes separados digitalmente. Cada ambiente virtualizado é executado dentro dos recursos alocados, como memória, capacidade de processamento e armazenamento.

Fonte: AWS

AD: Todos *slides* de conteúdo apresentam fundo branco, tarja verde fina na lateral esquerda e textos em preto.

WSL

O Subsistema Windows para Linux (WSL) é um recurso do Microsoft Windows que permite aos desenvolvedores executar um ambiente Linux sem a necessidade de uma máquina virtual separada ou inicialização dupla.

Fonte: Wikipedia

Ubuntu

É uma distribuição Linux que possui suporte ao WSL.

Segundo página do Ubuntu para o WSL:

“Access the power of a full Ubuntu terminal environment on Windows with Windows Subsystem for Linux (WSL).”

Fonte: <https://ubuntu.com/desktop/wsl>

Pré-requisitos para a atividade:

- Windows 10 ou Windows 11 (preferência pelo W11);
- Aplicativo PowerShell recente e Terminal do Windows instalado (Não é o CMD!);
- Visual Studio Code (VSCode) recente. Instalar os plugins Python, Python Debugger, Pylance e WSL.

Como ativar o WSL e instalar o Ubuntu:

- Caso não tenha utilizado o WSL ainda, utilize o comando:

```
wsl --install
```

Obs.: Será executada a instalação tanto do WSL, como uma instalação padrão do Ubuntu (atualmente é a versão 24.04.1).

Após o processo de instalação finalizado será solicitado um nome de usuário e a senha para o usuário no Linux.

Comandos comuns do WSL para uso no Terminal do Windows:

- `wsl --list` – Lista as distribuições instaladas;
- `wsl --list --online` – Lista as distribuições que podem ser instaladas via web;
- `wsl` – Inicia a distribuição padrão;
- `wsl -d <nome_da_distribuição>` – Inicia uma distribuição específica;
- `wsl --shutdown` – Encerra todas as instâncias do WSL e remove toda a memória alocada para operações.

No Linux, atualizando o SO para a versão mais recente:

- Em qualquer SO, a primeira coisa que se deve fazer, é atualizá-lo. No terminal do Linux, pode-se fazer isso com o par de comandos:

```
sudo apt update
```

```
sudo apt upgrade
```

O primeiro atualiza as fontes de instalação dos pacotes e o segundo faz o download e atualizações dos pacotes.

Vamos instalar o Python no Linus e rodar uma aplicação:

- Instalando o suporte para o Python 3.12:
`sudo apt install python3 python3-pip python3-virtualenv`
- Criando uma pasta o projeto definindo como um diretório para trabalhar com um projeto em Python:
`mkdir projeto`
`virtualenv projeto`

“Ativando” e desativando um projeto:

Para “ativar” o projeto e trabalhar no seu desenvolvimento, digite:

```
source projeto/bin/activate – Ativa o projeto  
cd projeto – Entra no diretório do projeto  
code . – Inicia o VSCode.
```

Caso queira sair e/ou ir para outro projeto, digite “deactivate”.

Aplicação de teste. Um mini servidor WEB:

O servidor WEB proposto pode implementar diversas funcionalidades. A ideia é apresentar uma aplicação rodando em Linux pelo WSL.

A aplicação proposta utiliza o framework Flask do Python. Para instalar os pacotes do Flask, utilize a ferramenta “pip” no projeto ativo:

```
pip install flask
```

Aplicação de teste. Um mini servidor WEB:

A aplicação pode ser executada no WSL. Como é um servidor WEB, deverá ser acessada por um navegador no endereço:

<http://localhost:5000>

A solid green vertical bar is located on the far left side of the image.

FIM!