

Unlock the Power of Dev Containers: Build a Consistent Python Development Environment in Seconds!

Thomas Fraunholz

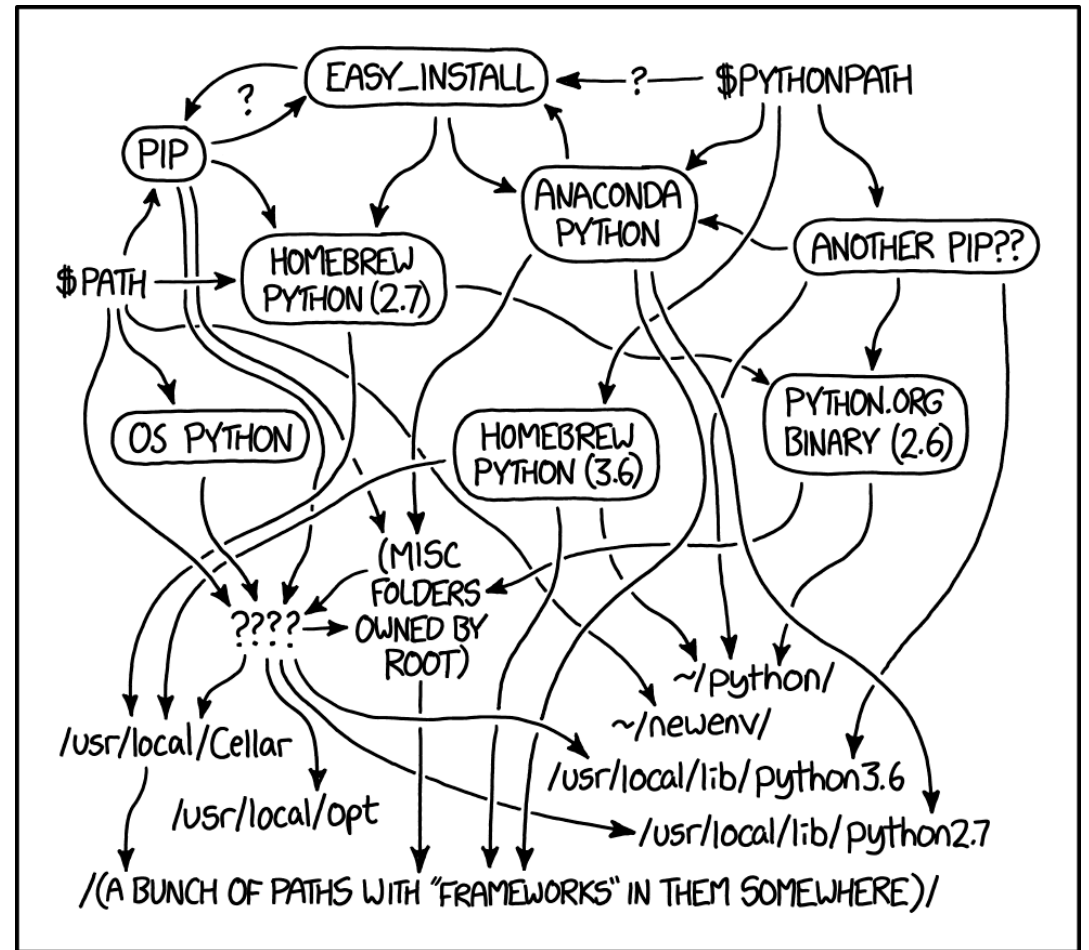
Smart Cyber Security GmbH

How it all began...

```
$ pip install something
```

How it all began...

```
$ pip install something
```



MY PYTHON ENVIRONMENT HAS BECOME SO DEGRADED
THAT MY LAPTOP HAS BEEN DECLARED A SUPERFUND SITE.

What's the Problem?



Minimalistic DevContainer Setup

```
├─ devcontainer
│  └─ .devcontainer.json
```

```
{
    "image": "mcr.microsoft.com/devcontainers/python:1-3.12-bullseye",
    "remoteUser": "vscode"
}
```

DevContainer Poetry Setup

```
|— devcontainer
   |— .devcontainer.json
```

```
{
    "image": "mcr.microsoft.com/devcontainers/python:1-3.12-bullseye",
    "remoteUser": "vscode"
    "postCreateCommand": "pip install poetry && poetry install",
}
```

```
|— pyproject.toml
|— poetry.lock
```

Git Integration

```
git clone https://<PERSONAL_ACCESS_TOKEN>@github.com/<REPOSITORY>
```

```
git clone https://<USERNAME>:<APP_PASSWORD>@bitbucket.org/<REPOSITORY>
```

```
git clone https://<TOKEN_NAME>:<PERSONAL_ACCESS_TOKEN>@gitlab.com/<REPOSITORY>
```


Environment Variables

```
|— devontainer  
  |— .devcontainer.json
```

```
{  
    ...  
    "runArgs": ["--env-file", ".env"],  
    ...  
}
```

```
|— pyproject.toml  
|— poetry.lock  
|— .env
```

Environment Variables

```
|— devontainer  
  |— .devcontainer.json  
|— pyproject.toml  
|— poetry.lock  
|— .env
```

```
GIT_AUTHOR_NAME=<YOUR_NAME>  
GIT_AUTHOR_EMAIL=<YOUR_EMAIL>  
GIT_COMMITTER_NAME=<YOUR_NAME>  
GIT_COMMITTER_EMAIL=<YOUR_EMAIL>
```

Advanced Topics

- Start services with docker-compose
- CI/CD with devcontainer-cli
- Platform Independent Development

Remote GPU Setup

```
|— devontainer
   |— .devcontainer.json
```

```
{
    "build": {
        "dockerfile": "Dockerfile",
        "context": ".."
    },
    "postCreateCommand": "bash scripts/install-dev-tools.sh",
    "runArgs": ["--gpus", "all", "--shm-size=1gb"],
    "remoteUser": "user",
}
```

```
|— Dockerfile
|— pyproject.toml
|— poetry.lock
```

```
├─ devcontainer
│  └─ .devcontainer.json
│  └─ Dockerfile
```

```
FROM nvidia/cuda:11.7.1-base-ubuntu22.04 AS base
```

```
# Install Python and Poetry
```

```
...
```

```
# Create User
```

```
ARG USER_GID=$USER_UID
```

```
RUN groupadd --gid $USER_GID user \
    && useradd --uid 1000 --gid $USER_GID -m user \
```

```
...
```

```
├─ pyproject.toml
└─ poetry.lock
```