Tutorial: Creating Conda Environments Using Shell Scripts

This document provides a step-by-step guide for creating reproducible Conda environments using shell (`.sh`) scripts. The goal is to help teams follow a consistent and standardized process when setting up environments for projects.

# 1. Prerequisites

- Conda (Anaconda or Miniconda) must be installed.  
- Verify installation using: `conda --version`  
- Ensure required Python version is supported by Conda.  
- Stable internet connection is needed for package installation.  
- Sufficient permissions to create environments and install dependencies.  
- A `requirements.txt` file (optional but recommended) listing dependencies.  
- On Windows: Run the script using Git Bash or WSL.

# 2. Dependencies & Tools

The script can be customized to install the following:  
- Required Python version  
- Project-specific dependencies listed in `requirements.txt`  
- Jupyter Notebook and ipykernel (if environment is to be used in Jupyter)  
- Any additional libraries specific to the project

# 3. Environment Naming Conventions

- Use clear and descriptive names (e.g., `<project\_name>\_env`).  
- Avoid spaces or special characters.  
- Ensure consistency across the team.

# 4. Shell Script Best Practices

- Always start with a shebang (`#!/bin/bash`).  
- Use variables for environment name and Python version.  
- Add echo logs for better readability.  
- Include error handling after critical steps.  
- Make the script executable before running: `chmod +x <script\_name>.sh`

# 5. Common Pitfalls to Avoid

- Forgetting to `source` Conda’s profile script before activating.  
- Typos in environment name.  
- Missing `requirements.txt` file.  
- Running the script from within another Conda environment.

# 6. Generalized Shell Script Template

#!/bin/bash  
  
# Variables (replace <> with actual values)  
ENV\_NAME="<environment\_name>"  
CONDA\_PYTHON\_VERSION="<python\_version>"  
  
echo ">>> Creating Conda environment: $ENV\_NAME with Python $CONDA\_PYTHON\_VERSION"  
  
# Step 1: Create environment  
conda create -y -n $ENV\_NAME python=$CONDA\_PYTHON\_VERSION  
if [ $? -ne 0 ]; then  
 echo "❌ Error: Failed to create Conda environment."  
 exit 1  
fi  
  
# Step 2: Activate environment  
echo ">>> Activating Conda environment..."  
source "$(conda info --base)/etc/profile.d/conda.sh"  
conda activate $ENV\_NAME  
  
# Step 3: Install dependencies (if requirements.txt exists)  
if [ -f "requirements.txt" ]; then  
 echo ">>> Installing dependencies from requirements.txt..."  
 pip install -r requirements.txt  
else  
 echo "⚠️ No requirements.txt found. Skipping pip installs."  
fi  
  
# Optional: Install Jupyter support  
conda install -y notebook ipykernel  
python -m ipykernel install --user --name $ENV\_NAME --display-name "$ENV\_NAME"  
  
echo "✅ Environment $ENV\_NAME created successfully!"

# 7. Verification

- Verify environment creation:  
 `conda env list`  
- Check Python version:  
 `python --version`  
- If Jupyter was installed, confirm kernel availability in Notebook.