

Node support — Quick reference & troubleshooting

Author : Badr TAJINI - DevOps for SWE - ESIEE 2025-2026

This document provides steps to install and configure Node.js (LTS version) in your terminal environment, ensuring compatibility with DevOps tools and workflows.

Install a compatible Node and persist it (recommended: nvm)

1. If nvm is not installed, install it via the install script in your terminal:

```
curl -o- https://raw.githubusercontent.com/nvm-sh/nvm/v0.39.7/install.sh | bash
```

2. Load nvm into the current shell, install LTS Node (>=22), and set it as default:

```
export NVM_DIR="$HOME/.nvm"
[ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh"
[ -s "$NVM_DIR/bash_completion" ] && \. "$NVM_DIR/bash_completion"

nvm install --lts
nvm use --lts
nvm alias default 'lts/*'

node --version    # should be >= v22.x
npm --version     # should be >= 10.9.2
```

3. Persist nvm in zsh permanently (append loader to `~/.zshrc`):

```
cat >> ~/.zshrc << 'ZSH'
export NVM_DIR="$HOME/.nvm"
[ -s "$NVM_DIR/nvm.sh" ] && \. "$NVM_DIR/nvm.sh"
[ -s "$NVM_DIR/bash_completion" ] && \. "$NVM_DIR/bash_completion"
ZSH
```

4. **IMPORTANT: Reload existing terminal sessions**

After adding nvm to `~/.zshrc`, your **current terminal sessions** won't have nvm loaded yet. Choose one:

```
# Option A: Reload .zshrc in current terminal (quick)
source ~/.zshrc

# Option B: Start a fresh shell (cleaner)
```

```
exec zsh -l

# Option C: Close all VS Code terminals and open new ones
# (Ctrl+Shift+P → Developer: Reload Window)
```

Verify nvm is now available:

```
nvm --version
node --version # should show v22.x or higher
npm --version # should show 10.9.2 or higher
```

If you use bash, add the same lines to `~/.bashrc` instead and reload with `source ~/.bashrc`.

This sum-ups the steps to install Node.js using nvm and ensure it's properly configured in your terminal environment. This setup is essential for working with various DevOps tools that depend on Node.js and concludes the Node.js support for lab0 (starter)