Pragmatic Node.js development
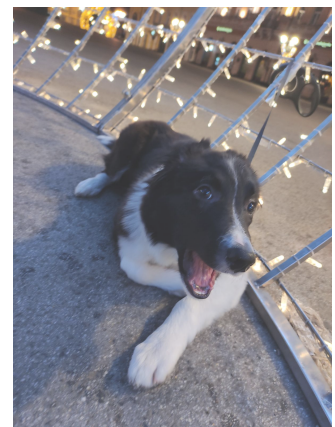
Primer in Nest.js

Author: Zlatibor Zed Veljkovic

Date: June 5, 2022

Version: 0.1

bioinfo 1: bioinfo 2

Some extra info

Contents

Chapter 1 Developer tools

Since ancient times, mankind has constantly spent effort to create new or to improve existing tools. Even now, after thousands and thousands of years we are doing the same. We are making new tools that will make us more efficient or at least to do our tasks easier. In software development there are so many tools available it is hard to choose which set should be used. Next sections will give simple overview of most prominent tools for each section.

1.1 Command line interfaces

Commad line interfaces (CLI) are programs that use textual interface and allow you to interact with it. Every operating system comes with one or more of these, Windows has Command Prompt (aka cmd.exe) and Power Shell, Linux has sh and bash with many alternatives (commonly known as shell), and Mac OS has Terminal.app. One issue that novice developers struggle is that when someone tells you to "open the terminal", they mean one for your system.

Before mentioned apps are the ones that allow you to execute some commands or run different programs. There are also some CLI that is specifically built for one purpose. One example would be Nest.js CLI which allows you to quickly create new projects, update dependencies or start the Nest app.
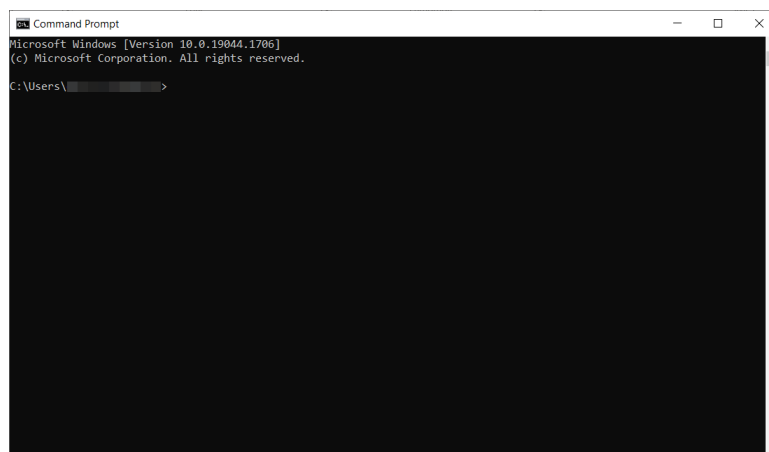
1.1.1 Command prompt

Command prompt comes preinstalled on Windows systems. It supports batch scripts and usually the file is with .bat extension.

Pros	Cons
• Available on all Windows systems	• Batch scripting language is really outdated and hard to write
• Allows executing programs in current directory without .\ prefix	more complex stuff

Figure 1.1: Command Prompt

Reference for available Command Prompt commands can be found at Windows Commands

1.1.2 PowerShell

Another shell for Windows systems is called PowerShell and it is available from Windows 7 or later operating systems. Open source version PowerShell Core was released in 2016 and it is based on .Net Core which also made it cross-platform. It has better integration with various functionalities available in Windows so it is prefeed choice over Command Prompt when working with system administration. For developer work it might be an overkill.

Pros

- Available on all Windows systems
- Better integration with Windows functionalities

Cons

- Does not allow executing programs in current directory without .\ prefix