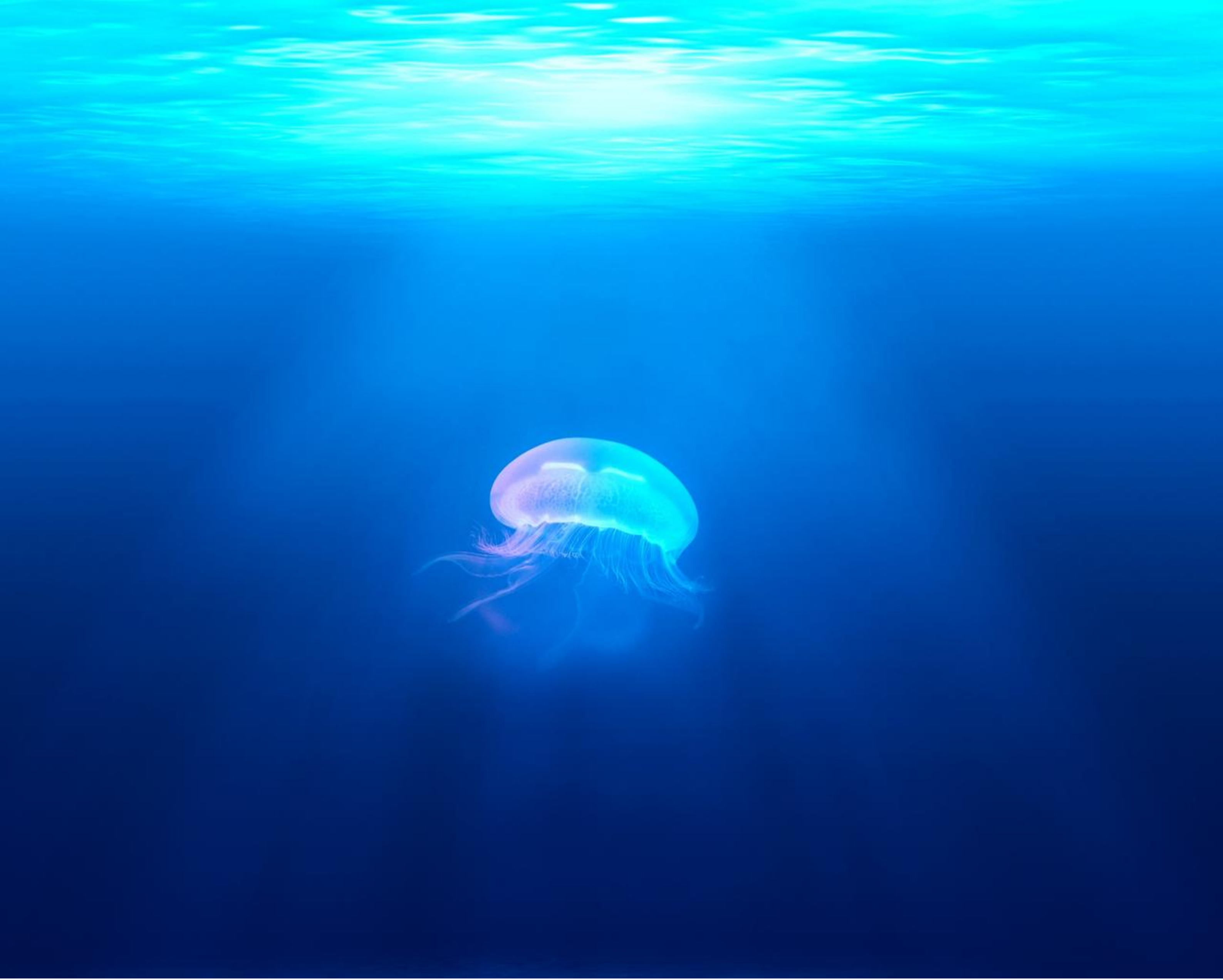
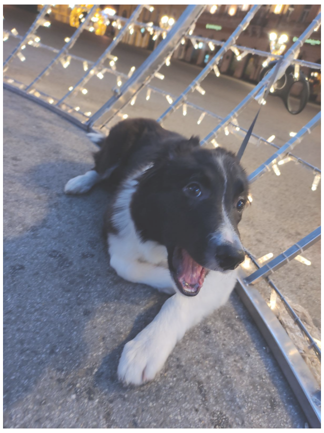


Pragmatic Node.js development



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 **1**

1.1 Command line interfaces 1

1.1.1 Command prompt 1

1.1.2 PowerShell 1

1.1.3 Linux/Mac terminals/shells 2

1.2 Package managers 2

1.2.1 Chocolatey 2

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

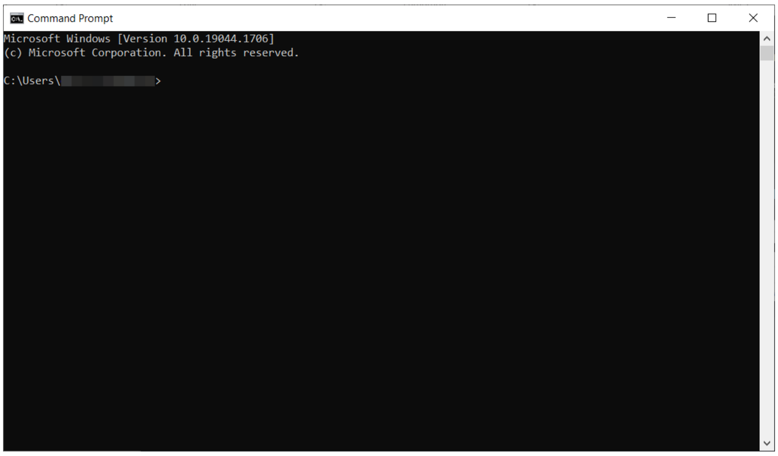
Command prompt comes preinstalled on Windows systems. It supports batch scripts and usually the file is with .bat extension.

Pros

- Available on all Windows systems
- Allows executing programs in current directory without .\ prefix

Cons

- Batch scripting language is really outdated and hard to write more complex stuff
- No command history search



```
Microsoft Windows [Version 10.0.19044.1706]
(c) Microsoft Corporation. All rights reserved.

C:\Users\[redacted]>
```

Figure 1.1: Command Prompt

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

1.1.2 PowerShell

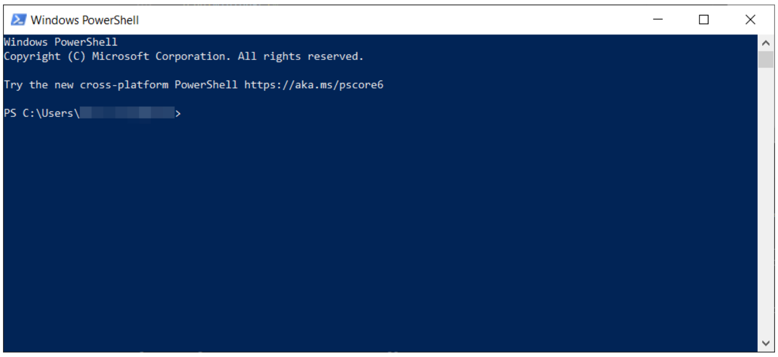
Another shell for Windows systems is called PowerShell and it is available from Windows 7 or later operating systems. Open source version PowerShell Core was released in 2016 and it is based on .Net Core which also made it cross-platform. It has better integration with various functionalities available in Windows so it is prefreed 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
- Has command history search (with F8 key)

Cons

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



```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/pscore6

PS C:\Users\[redacted]>
```

Figure 1.2: PowerShell

1.1.3 Linux/Mac terminals/shells

Linux and Mac have much bigger choice of terminals. Terminal refers to a program that allows you to run programs which are known as shells. Shells come in lots of varieties sh, bash, ksh, csh, zsh. . . . Linux has many command line programs that allow you to manipulate output of commands and offers a lot for power users.

1.2 Package managers

Package managers can be used to install additional software on your PC. They usully automate process of downloading, installing and configuring software. Later it also helps with keeping the installed software up to date or with removal.

1.2.1 Chocolatey

Chocolatey is most prominent Windows package manager. It can be downloaded from https://chocolatey.org/. Searching for packages is done with `choco search postgresql` and installation with `choco install postgresql` will install Postgres.