

# **Curso Completo de Neovim: De Novato a Maestro**

Diego Saavedra

Nov 5, 2024

# Table of contents

<b>1</b>	<b>Curso Completo de Neovim: De Novato a Maestro</b>	<b>3</b>
1.1	Descripción del Curso . . . . .	3
1.2	Planificación del Curso . . . . .	3
1.2.1	<b>Módulo 1:</b> Introducción a Neovim . . . . .	3
1.2.2	<b>Módulo 2:</b> Fundamentos de Neovim y Comandos Básicos . . . . .	4
1.2.3	<b>Módulo 3:</b> Configuración y Personalización de Plugins con Lazy . . . . .	4
1.2.4	<b>Módulo 4:</b> Dominio de Edición y Navegación Avanzada en Neovim . . . . .	5
1.2.5	<b>Módulo 5:</b> Configuración Avanzada y Personalización en Neovim . . . . .	5
1.2.6	<b>Módulo 6:</b> Integración y Automatización del Flujo de Trabajo . . . . .	6
1.2.7	<b>Módulo 7:</b> Proyecto Final y Optimización de Configuración . . . . .	6
1.3	Materiales de Apoyo . . . . .	7
1.4	Tareas Adicionales y Proyectos . . . . .	7
1.5	Soporte y Comunidad . . . . .	7
<b>I</b>	<b>Unidad 0: Introducción a Neovim</b>	<b>8</b>
<b>2</b>	<b>Introduction a Neovim</b>	<b>9</b>

# 1 Curso Completo de Neovim: De Novato a Maestro

## 1.1 Descripción del Curso

Este curso tiene como objetivo llevar al estudiante desde un nivel básico de comprensión de Neovim hasta un nivel avanzado de personalización, edición eficiente y automatización del flujo de trabajo. Usaremos Lazy como gestor de plugins, junto con herramientas como LazyGit y Gitsigns para optimizar el trabajo en Git, y exploraremos cómo integrar Neovim con herramientas modernas de desarrollo.

## 1.2 Planificación del Curso

### 1.2.1 Módulo 1: Introducción a Neovim

**Objetivo:** Familiarizarse con Neovim y sentar las bases de su configuración y personalización.

**Lección 1:** Introducción al Curso y Objetivos

- Historia de Vim y Neovim.
- Beneficios y características de Neovim.
- Estructura del curso y recursos adicionales.

**Lección 2:** Instalación de Neovim y Configuración Básica

- Instalación en Windows y Linux.
- Creación de la carpeta de configuración y archivos principales (init.lua o init.vim).
- Instalación de Lazy como gestor de plugins.

### 1.2.2 Módulo 2: Fundamentos de Neovim y Comandos Básicos

**Objetivo:** Aprender los comandos esenciales de Neovim y la navegación básica.

**Lección 1:** Navegación y Modos de Neovim

- **Modos de operación:** Normal, Insert, Visual, Command.
- Navegación en el archivo y comandos básicos de movimiento.

**Lección 2:** Edición de Texto y Búsqueda en Neovim

- Comandos para edición, eliminación, y copia de texto.
- Búsqueda y reemplazo de texto en Neovim.

**Lección 3:** Configuración Inicial de Opciones

- Configuración de indentación, color de fondo y fuentes.
- Introducción al archivo `init.lua` y a las opciones de personalización.

### 1.2.3 Módulo 3: Configuración y Personalización de Plugins con Lazy

**Objetivo:** Instalar y configurar plugins clave que aumentarán la eficiencia en Neovim.

**Lección 1:** Gestión de Plugins con Lazy

- Instalación y configuración de Lazy.
- Estrategias para organizar plugins en Lazy.

**Lección 2:** Exploración de Plugins de Edición y Productividad

- Instalación de plugins como `nvim-treesitter` para resaltado avanzado de sintaxis y `comment.nvim` para gestión de comentarios.
- Configuración y personalización de cada plugin para adaptarlos al flujo de trabajo.

**Lección 3:** Integración con Git usando LazyGit y Gitsigns

- Instalación de LazyGit y configuración de accesos directos para su uso en Neovim.
- Uso de Gitsigns para la visualización de cambios en el código.
- Práctica guiada de flujos de trabajo comunes con Git.

#### 1.2.4 Módulo 4: Dominio de Edición y Navegación Avanzada en Neovim

**Objetivo:** Profundizar en técnicas de edición avanzada y optimizar la navegación.

**Lección 1:** Movimientos Avanzados y Edición con Macros

- Navegación rápida por archivos grandes y uso de marcadores.
- Creación y uso de macros para automatizar tareas repetitivas.

**Lección 2:** Uso de Bufers, Ventanas y Pestañas

- Explicación de buffers, ventanas y pestañas.
- Configuración para alternar y organizar múltiples archivos eficientemente.

**Lección 3:** Configuración de Snippets con LuaSnip

- Instalación de LuaSnip y configuración de snippets personalizados.
- Creación de snippets de código para lenguajes específicos.

#### 1.2.5 Módulo 5: Configuración Avanzada y Personalización en Neovim

**Objetivo:** Explorar configuraciones avanzadas y aprender a personalizar Neovim a un alto nivel.

**Lección 1:** Personalización de Atajos de Teclado y Mapeos

- Cómo crear mapeos personalizados en `init.lua`.
- Integración de mapeos para plugins instalados.

**Lección 2:** Configuración de LSP (Language Server Protocol)

- Explicación del concepto de LSP y su utilidad en la autocompletación y análisis de código.
- Instalación y configuración del cliente de LSP en Neovim para varios lenguajes.

**Lección 3:** Exploración de `nvim-cmp` y `telescope.nvim`

- Configuración de `nvim-cmp` para autocompletado inteligente.
- Uso de `telescope.nvim` para buscar archivos, símbolos y contenido en el proyecto.

### 1.2.6 Módulo 6: Integración y Automatización del Flujo de Trabajo

**Objetivo:** Aprender a integrar Neovim con otras herramientas y automatizar tareas para mejorar la productividad.

**Lección 1:** Integración con Terminal Embebido y Comandos Personalizados

- Cómo usar el terminal integrado en Neovim.
- Ejemplos de comandos personalizados para tareas frecuentes.

**Lección 2:** Integración con Herramientas Externas y Workflows

- Integración de Neovim con herramientas como Git, Docker, y linters.
- Ejemplos prácticos de automatización en flujos de desarrollo.

**Lección 3:** Configuración de Tareas y Scripts Automatizados

- Configuración de scripts en Neovim para realizar tareas comunes.
- Ejemplos de tareas automáticas al guardar, abrir o cerrar archivos.

### 1.2.7 Módulo 7: Proyecto Final y Optimización de Configuración

**Objetivo:** Consolidar los conocimientos adquiridos en un proyecto práctico y optimizar la configuración final de Neovim.

**Lección 1:** Creación de una Configuración Modular de Neovim

- Organización de los archivos de configuración en módulos.
- Mejores prácticas para mantener una configuración escalable y organizada.

**Lección 2:** Proyecto Final - Entorno de Desarrollo Completo en Neovim

- Los estudiantes crearán su propio entorno de desarrollo completo en Neovim.
- Integración de todas las herramientas y plugins utilizados en el curso.
- Personalización final para optimizar el flujo de trabajo.

**Lección 3:** Exportación y Respaldo de la Configuración

- Cómo versionar la configuración en GitHub.
- Compartir configuraciones y mantener un respaldo en caso de migración.

## 1.3 Materiales de Apoyo

- **Repositorio en GitHub:** Ejemplos de configuración, ejercicios y soluciones.
- **Documentación Complementaria:** Referencias a la documentación oficial de Neovim, Lazy, LSP, y otros plugins.
- **Guías de Instalación:** Instrucciones específicas para instalar LazyGit, Gitsigns y otros complementos en diferentes sistemas operativos.
- **Ejercicios Prácticos y Desafíos:** Actividades diseñadas para aplicar lo aprendido en cada módulo.

## 1.4 Tareas Adicionales y Proyectos

- Prácticas recomendadas al final de cada módulo para afianzar los conocimientos.
- Proyectos opcionales como configurar Neovim para proyectos específicos (por ejemplo, proyectos en Python, JavaScript, etc.).

## 1.5 Soporte y Comunidad

- **Acceso a foros y grupo de Discord:** Para resolver dudas, compartir configuraciones y obtener feedback.
- **Recursos adicionales:** Artículos, guías y tutoriales que complementan el curso.

## **Part I**

### **Unidad 0: Introducción a Neovim**



## 2 Introduction a Neovim

Nvim es una herramienta que te permite editar texto de forma eficiente. Es un editor de texto que se puede personalizar para adaptarse a tus necesidades. A diferencia de otros editores de texto, Nvim se puede utilizar completamente desde el teclado, lo que lo hace muy eficiente para los programadores y escritores.