


dev Senior Code

Python de Cero a Senior
La Ruta Maestra del Código



dev Senior Code

“El futuro lo creas tú...”

Curso Completo

“Python de Cero a Senior: La Ruta Maestra del Código”



Modalidad
Online - En Vivo



Nivel de dificultad
Inicial



Dedicación
Media - Alta



Proyecto Final.



Curso Completo “Python de Cero a Senior: La Ruta Maestra del Código”

Acerca del curso.

Bienvenidos al curso "Python de Cero a Senior: La Ruta Maestra del Código". Este curso está diseñado para llevar a los estudiantes desde los fundamentos más básicos de la programación en Python hasta el nivel experto en desarrollo de software. Con un enfoque integral en el backend utilizando Django y Flask, manejo avanzado de bases de datos, y desarrollo frontend con frameworks modernos como React o Vue.js. En un mundo donde la tecnología avanza rápidamente y la demanda de desarrolladores calificados es alta, adquirir habilidades completas en estas áreas te proporcionará una ventaja competitiva significativa en el mercado laboral.



Objetivo principal del curso.

El objetivo principal de este curso es capacitar a los estudiantes para convertirse en desarrolladores Senior en Python, capaces de diseñar, desarrollar y mantener aplicaciones completas y robustas. Esto incluye:

Dominio del Backend con Django y Flask: Aprenderás a crear y gestionar aplicaciones backend utilizando Django y Flask, dos de los frameworks más potentes y flexibles para el desarrollo de aplicaciones empresariales en Python. Desde los conceptos básicos hasta la implementación avanzada de microservicios, te proporcionaremos las herramientas necesarias para construir aplicaciones escalables y mantenibles.

1. **Manejo Avanzado de Bases de Datos:** Comprenderás tanto bases de datos relacionales como NoSQL, utilizando tecnologías como PostgreSQL, MySQL y MongoDB. Aprenderás a gestionar datos de manera eficiente, optimizar consultas y garantizar la integridad de los datos en aplicaciones complejas.
2. **Desarrollo Frontend con React y Vue.js:** Te enseñaremos a desarrollar interfaces de usuario modernas y dinámicas utilizando React o Vue.js, dos de las bibliotecas más populares para el desarrollo frontend. Aprenderás a crear componentes reutilizables, manejar el estado de la aplicación y conectar el frontend con el backend para ofrecer una experiencia de usuario fluida.



3. Integración de Conocimientos: A lo largo del curso, integrarás tus habilidades en backend y frontend para construir aplicaciones completas, desde la planificación y el diseño hasta el despliegue y la optimización. También te familiarizarás con herramientas de DevOps, CI/CD y pruebas para asegurar la calidad y la eficiencia de tus desarrollos.

Metodología del Curso

Este Bootcamp está estructurado en 12 módulos mensuales, cada uno abordando diferentes aspectos y niveles de complejidad. Cada módulo está compuesto por 8 clases, con una duración de 2 horas cada una, además de una tutoría semanal de 2 horas para resolver dudas y reforzar los conocimientos.

La metodología incluye:

- **Teoría y Práctica:** Cada clase combina teoría con prácticas y ejercicios que refuerzan los conceptos aprendidos. Los estudiantes tendrán la oportunidad de trabajar en proyectos reales y casos de estudio que simulan escenarios del mundo laboral.
- **Proyectos Integradores:** A lo largo del curso, se llevarán a cabo proyectos integradores que abarcan tanto el desarrollo backend como frontend. Estos proyectos permitirán a los estudiantes aplicar sus conocimientos en contextos reales y construir aplicaciones completas con Python, integrando tanto Django y Flask en el backend como React o Vue.js en el frontend.
- **Evaluaciones Continuas:** Se realizarán evaluaciones periódicas para medir el progreso de los estudiantes y asegurar que los conceptos se han comprendido y aplicado correctamente. Estas evaluaciones incluirán tanto pruebas teóricas como ejercicios prácticos que pondrán a prueba las habilidades adquiridas.
- **Preparación para el Mercado Laboral:** En el último mes del curso, se incluirá preparación específica para entrevistas y oportunidades laborales, ayudando a los estudiantes a presentar sus habilidades y proyectos de manera efectiva. Se realizarán simulaciones de entrevistas técnicas para mejorar las habilidades de comunicación y presentación.



➤ Beneficios del Curso

Al completar este Bootcamp, los estudiantes estarán equipados con:

- **Habilidades Técnicas Avanzadas:** Conocimiento profundo en Python, frameworks backend como Django y Flask, bases de datos relacionales y NoSQL, y frameworks frontend como React o Vue.js.
- **Experiencia Práctica:** Experiencia en el desarrollo de aplicaciones completas, desde la concepción hasta el despliegue, utilizando Python y tecnologías modernas en entornos reales.
- **Preparación para el Mercado Laboral:** Herramientas y estrategias para destacar en el campo de la programación y el desarrollo de software, con proyectos tangibles que demostrarán sus habilidades a futuros empleadores.

Duración del Curso: 12 Meses

- Clases por mes: 8 clases (2 por semana).
- Duración de cada clase: 2 horas.
- Tutoría semanal: 2 horas para aclarar dudas y reforzar conocimientos.





Estructura del Curso por Módulos y Clases

Mes 1: Fundamentos de Programación en Python

Clase 1: Introducción a python y configuración del entorno de desarrollo

- Instalación de Python, IDEs (PyCharm, VSCode) y configuración de Git / GitHub.

Clase 2: Variables, tipos de datos y operadores.

- Operadores aritméticos, relacionales y lógicos en Python.

Clase 3: Estructuras de control de flujo.

- Condicionales y bucles básicos (if, else, for, while).

Clase 4: Funciones y modularidad.

- Declaración y uso de funciones.

Clase 5: Introducción a listas, tuplas y diccionarios.

- Manipulación de estructuras de datos básicas.

Clase 6: Conceptos básicos de programación funcional.

- Uso de lambda, map, filter y reduce.

Clase 7: Ejercicio práctico integrador con estructuras de control.

Clase 8: Repaso y mini -proyecto básico en Python.

Mes 2: Programación Orientada a Objetos (POO) en Python

Clase 1: Introducción a POO: Clases y objetos.

- Creación de clases, métodos y objetos.

Clase 2: Encapsulamiento y herencia.

- Uso de modificadores de acceso y herencia en Python.

Clase 3: Polimorfismo e interfaces

- Implementación de polimorfismo y clases abstractas.

Clase 4: Patrones de diseño básicos: Singleton y Factory.

- Implementación y uso en aplicaciones Python.

Clase 5: Composición Vs Herencia y mejores prácticas.

- Diseño orientado a objetos efectivo.

Clase 6: Introducción a patrones de diseño avanzados (Observer, Strategy)

Clase 7: Clases anónimas, internas y lambda expressions.

- Uso práctico en aplicaciones reales.

Clase 8: Proyecto práctico de aplicación de POO en Python.



Mes 3: Manejo de Excepciones y Depuración

Clase 1: Excepciones y manejo de errores en Python.

- Uso de try, except, finally y creación de excepciones personalizadas.

Clase 2: Logs y depuración.

- Uso de la biblioteca logging.

Clase 3: Técnicas avanzadas de depuración en IDEs.

- Uso de breakpoints, evaluación de expresiones y seguimiento.

Clase 4: Gestión avanzada de errores en aplicaciones.

- Estrategias para manejo robusto de excepciones.

Clase 5: Pruebas unitarias con unittest y pytest.

- Pruebas básicas y manejo de errores.

Clase 6: Depuración y resolución de problemas comunes.

Clase 7: Taller práctico de manejo de errores y logging.

Clase 8: Mini -proyecto con enfoque en manejo de excepciones y depuración.

Mes 4: Colecciones y Manipulación de Datos

Clase 1: Introducción a colecciones de Python: Listas, Tuplas, Sets y Diccionarios.

Clase 2: Iteradores y generadores.

- Introducción a los generadores y su uso eficiente.

Clase 3: Comprensiones de listas y diccionarios.

- Manipulación de datos con list comprehension.

Clase 4: Uso avanzado de itertools y collections.

- Técnicas avanzadas con estas bibliotecas.

Clase 5: Streams de datos con Python (map, filter, reduce).

Clase 6: Operaciones avanzadas con colecciones.

Clase 7: Proyecto práctico: Manipulación avanzada de datos.

Clase 8: Taller práctico de optimización de colecciones.



Mes 5: Bases de Datos Relacionales y NoSQL

Clase 1: Introducción a bases de datos relacionales (PostgreSQL, MySQL).

- CRUD básico y SQL.

Clase 2: Conexión a bases de datos con sqlite3 y psycopg2.

- CRUD básico con conexiones de bases de datos.

Clase 3: ORM con SQLAlchemy y Django ORM.

Clase 4: Relaciones entre entidades en bases de datos relacionales.

Clase 5: Introducción a bases de datos NoSQL (MongoDB).

Clase 6: Consultas avanzadas en bases de datos NoSQL.

Clase 7: Proyecto práctico con bases de datos relacionales y NoSQL.

Clase 8: Taller práctico de optimización de bases de datos.

Mes 6: Desarrollo Backend con Flask

Clase 1: Introducción a Flask y configuración inicial del proyecto.

- Creación de un proyecto básico con Flask.

Clase 2: Controladores y manejo de peticiones HTTP.

- Rutas, métodos GET y POST en Flask.

Clase 3: Servicios y lógica de negocio en Flask.

- Uso de `@app.route` y `@app.errorhandler`.

Clase 4: Conexión a bases de datos en Flask.

Clase 5: Seguridad básica con autenticación.

Clase 6: Testing en Flask.

Clase 7: Proyecto práctico de backend con Flask.

Clase 8: Repaso y optimización.



Mes 7: Desarrollo Backend con Django

Clase 1: Introducción a Django y configuración inicial del proyecto. Creación de un proyecto básico con Django.

Clase 2: Modelos y ORM en Django.

Clase 3: Vistas y URL routing en Django.

Clase 4: Formularios y validación de datos.

Clase 5: Seguridad avanzada con autenticación y permisos.

Clase 6: APIs RESTful con Django REST Framework.

Clase 7: Testing en Django.

Clase 8: Proyecto práctico de backend avanzado con Django.

Mes 8: Desarrollo Frontend con React

Clase 1: Introducción a React y configuración del entorno.

- Creación de un proyecto con Create React App.

Clase 2: Componentes en React.

- Creación de componentes funcionales y de clase.

Clase 3: Manejo de eventos y formularios en React.

Clase 4: React Hooks (useState, useEffect).

Clase 5: Rutas en React con React Router.

Clase 6: Consumo de APIs en React.

Clase 7: Manejo de estado global con Redux.

Clase 8: Proyecto práctico: Aplicación frontend con React.



Mes 9: Integración Full Stack (Django + React)

Clase 1: Conexión entre backend (Django) y frontend (React).

Clase 2: Creación de un CRUD completo.

Clase 3: Gestión de autenticación y sesiones.

Clase 4: Diseño de UI con Material-UI o Bootstrap.

Clase 5: Optimización de rendimiento en aplicaciones full stack.

Clase 6: Despliegue en producción (AWS).

Clase 7: Testing end-to-end.

Clase 8: Proyecto final de integración full stack.

Mes 10: Testing y Calidad de Código

Clase 1: Pruebas unitarias con unittest y pytest.

Clase 2: Testing de integración en backend (Flask/Django).

Clase 3: Testing de frontend con Jest y React Testing Library.

Clase 4: Pruebas de rendimiento.

Clase 5: Análisis de calidad de código con linters.

Clase 6: Cobertura de pruebas y herramientas de medición.

Clase 7: Desarrollo guiado por pruebas (TDD).

Clase 8: Proyecto práctico de testing avanzado.

Mes 11: DevOps y CI/CD con Python

Clase 1: Introducción a CI/CD y DevOps.

Clase 2: Pipelines con Jenkins y GitLab CI.

Clase 3: Despliegue con Docker.

Clase 4: Orquestación con Kubernetes.

Clase 5: Automatización de pruebas y despliegue.

Clase 6: Monitorización y logging.

Clase 7: Proyecto práctico de CI/CD.

Clase 8: Taller de despliegue y monitorización.



Mes 12: Proyecto Final y Preparación para el Nivel Senior

Clase 1: Definición y planificación del proyecto final.

- Especificación de requisitos y diseño arquitectónico.

Clase 2: Desarrollo del backend del proyecto.

- Implementación de servicios y APIs en Flask/Django.

Clase 3: Integración de bases de datos y servicios.

- Configuración de persistencia y transacciones.

Clase 4: Seguridad y autenticación en el proyecto.

- Implementación de seguridad avanzada con JWT o OAuth.

Clase 5: Pruebas y optimización del proyecto.

- Validación de rendimiento y corrección de errores.

Clase 6: Despliegue en un ambiente de producción.

- Uso de contenedores Docker y configuración de CI/CD.

Clase 7: Revisión de código y retroalimentación.

- Mejores prácticas y refactorización.

Clase 8: Preparación para entrevistas y oportunidades laborales.

Competencias de Aprendizaje Adquiridas en el Curso " Python de Cero a Senior: La Ruta Maestra del Código "

Al finalizar el curso " Python de Cero a Senior: La Ruta Maestra del Código", los estudiantes habrán adquirido un conjunto integral de competencias que les permitirán destacar en el campo del desarrollo de software. A continuación, se detallan las competencias clave que los estudiantes desarrollarán:

1. Dominio del Backend con Django y Flask:

- o Al finalizar el curso, serás capaz de crear aplicaciones backend sólidas utilizando los frameworks más potentes y flexibles disponibles en Python, como Django y Flask. Aprenderás desde los conceptos básicos de creación de proyectos hasta la implementación de sistemas avanzados y escalables, utilizando buenas prácticas de diseño y arquitectura de software.



- Adquirirás habilidades en el manejo de la lógica de negocio, la gestión de controladores, y la interacción con bases de datos, lo que te permitirá construir aplicaciones empresariales de gran envergadura. Además, estarás capacitado para diseñar microservicios que aseguren la modularidad y escalabilidad de tus proyectos, permitiendo una arquitectura eficiente y mantenible.

2. Manejo Avanzado de Bases de Datos:

- Aprenderás a gestionar bases de datos relacionales (PostgreSQL, MySQL) y bases de datos NoSQL (MongoDB) para aplicaciones modernas. Desde la creación de bases de datos hasta la implementación de operaciones CRUD (Crear, Leer, Actualizar, Borrar), serás capaz de manejar grandes volúmenes de datos de manera eficiente.
- Te enfocarás en la optimización de consultas, el diseño de relaciones entre entidades, y la utilización de ORMs (ObjectRelational Mapping) como SQLAlchemy y Django ORM, que te permitirán mapear estructuras complejas de bases de datos en aplicaciones Python. Además, aprenderás técnicas avanzadas para garantizar la integridad y la seguridad de los datos, lo cual es fundamental para desarrollar aplicaciones confiables.

3. Desarrollo Frontend con React

- Obtendrás experiencia práctica en el desarrollo de interfaces de usuario dinámicas y responsivas con dos de los frameworks más populares del frontend: React. Serás capaz de desarrollar componentes reutilizables y manejar el estado de la aplicación de manera eficiente, permitiendo la creación de interfaces interactivas y bien estructuradas.
- Te especializarás en la conexión entre el frontend y el backend mediante el consumo de APIs, lo que permitirá crear aplicaciones web completas. Además, aprenderás a gestionar eventos, formularios, rutas, y estado global mediante Redux en React, herramientas que son esenciales para el desarrollo de aplicaciones de gran escala y altamente interactivas.

4. Integración de Conocimientos Full:Stack

- A lo largo del curso, adquirirás la capacidad de integrar conocimientos tanto de backend como de frontend para desarrollar aplicaciones full stack . Serás capaz de diseñar y construir proyectos completos desde la planificación hasta el despliegue, utilizando frameworks de backend (Django o Flask) y frontend (React), lo que te permitirá crear aplicaciones robustas y escalables.
- Además, te familiarizarás con las herramientas y procesos de DevOps, como la implementación de pipelines de CI/CD (Integración Continua y Entrega Continua), el uso de contenedores Docker y la orquestación de aplicaciones con Kubernetes. Estas habilidades te permitirán no solo desarrollar aplicaciones, sino también gestionarlas y optimizarlas para su funcionamiento en entornos de producción.



5. Preparación para el Mercado Laboral :

- o A lo largo del Bootcamp, trabajarás en proyectos reales que simulan escenarios del mundo laboral, lo que te permitirá adquirir experiencia práctica en el desarrollo de software. Estos proyectos no solo te ayudarán a aplicar los conocimientos adquiridos, sino que también te permitirán construir un portafolio tangible que podrás presentar a futuros empleadores.
- o Al finalizar el curso, participarás en simulaciones de entrevistas técnicas que te prepararán para enfrentar procesos de selección en empresas tecnológicas. Aprenderás a comunicar tus habilidades de forma efectiva y a presentar tus proyectos de manera que demuestres tu competencia técnica. También recibirás orientación sobre las tendencias del mercado y cómo destacarte entre otros profesionales, aumentando significativamente tus oportunidades laborales.

Estas competencias cubren todas las áreas esenciales del desarrollo de software, desde la programación backend y frontend hasta el despliegue y la gestión de aplicaciones en producción, brindándote una preparación integral para convertirte en un desarrollador Python senior

Metodología del Bootcamp "Python de Cero a Senior: La Ruta Maestra del Código"

El Bootcamp "Python de Cero a Senior: La Ruta Maestra del Código" está diseñado para ofrecer una experiencia de aprendizaje integral, con un enfoque práctico en el desarrollo de software utilizando Python, Django, Flask, bases de datos, y frameworks frontend como React y Vue.js. El curso se imparte en un formato 100% virtual a través de Zoom, facilitado por docentes expertos de la industria, quienes guiarán a los estudiantes en cada etapa de su formación.

Formato del Curso:

1. Formación Virtual a través de Zoom

El Bootcamp "Python de Cero a Senior: La Ruta Maestra del Código" se imparte completamente en línea mediante sesiones en vivo a través de Zoom. Esta metodología ofrece la flexibilidad de aprender desde cualquier lugar, permitiendo a los estudiantes participar activamente sin necesidad de desplazarse. Las sesiones virtuales incluyen:

- **Clases Interactivas:** Los instructores especializados imparten clases en tiempo real, explicando tanto los conceptos teóricos como guías prácticas. A través de ejemplos aplicados y casos de estudio, los estudiantes pueden interactuar directamente con los docentes, lo que permite una resolución inmediata de dudas y discusiones dinámicas sobre los temas.
- **Tutorías Semanales :** Además de las clases, se ofrecen tutorías semanales para profundizar en los contenidos, aclarar dudas y reforzar los conocimientos adquiridos. Estas sesiones brindan la oportunidad de recibir retroalimentación personalizada sobre los ejercicios y proyectos que los estudiantes estén desarrollando.



2. Enfoque Práctico y Experiencial

El Bootcamp sigue un enfoque eminentemente práctico, donde los estudiantes aplican los conceptos aprendidos mediante ejercicios y proyectos reales. Esto incluye:

- **Proyectos Integradores** : Cada módulo del curso incluye proyectos que abarcan tanto el desarrollo backend como frontend, con frameworks como Django, Flasky React. Estos proyectos permiten a los estudiantes aplicar sus habilidades en la creación de aplicaciones completas, simulando el entorno de trabajo real.
- **Ejercicios y Casos de Estudio** : A lo largo del curso, los estudiantes resolverán ejercicios prácticos y trabajarán en casos de estudio basados en problemas reales que enfrentan los desarrolladores de software. Estos ejercicios refuerzan el aprendizaje, ayudando a los estudiantes a adquirir experiencia en la solución de problemas complejos.

3. Docentes Expertos de la Industria

El Bootcamp cuenta con un equipo de docentes altamente capacitados, con amplia experiencia en la industria del desarrollo de software. Los beneficios de contar con expertos incluyen:

- **Conocimiento Actualizado**: Los docentes están en contacto continuo con las últimas tendencias y tecnologías del desarrollo de software, garantizando que el contenido del curso esté alineado con las demandas actuales del mercado.
- **Experiencia Práctica**: Al ser profesionales activos en la industria, los instructores comparten sus experiencias reales de proyectos, brindando una perspectiva práctica que enriquece la enseñanza. Esto permite que los estudiantes vean cómo se aplican los conocimientos en proyectos reales.
- **Red de Contactos y Oportunidades** : Además de la enseñanza técnica, los instructores pueden proporcionar orientación sobre oportunidades de empleo y establecer contactos en la industria, apoyando a los estudiantes en su inserción en el mercado laboral.

4. Evaluaciones y Retroalimentación Continua

El progreso de los estudiantes es monitoreado a través de evaluaciones periódicas diseñadas para asegurar que los conceptos clave se comprendan y se puedan aplicar en situaciones prácticas. Este proceso incluye:

- **Pruebas y Evaluaciones**: A lo largo del curso se realizarán pruebas teóricas y prácticas que evalúan el dominio de los conceptos de programación, frameworks y tecnologías aprendidas. Esto ayudará a los estudiantes a identificar sus puntos fuertes y áreas de mejora.
- **Retroalimentación Constructiva** : Los estudiantes recibirán comentarios detallados sobre sus proyectos y ejercicios, ayudándoles a mejorar sus habilidades y ajustar sus enfoques. Las tutorías semanales brindan un espacio adicional para revisar el trabajo realizado y obtener más orientación.

5. Preparación para el Mercado Laboral

El Bootcamp está diseñado no solo para enseñar habilidades técnicas avanzadas, sino también para preparar a los estudiantes para ingresar al mercado laboral. Esto se logra mediante:



- **Desarrollo de Proyectos Reales** : Los estudiantes trabajarán en proyectos que podrán incluir en su portafolio profesional, mostrando a futuros empleadores sus competencias en el desarrollo de aplicaciones completas utilizando Python y tecnologías modernas.
- **Preparación para Entrevistas** : En el último mes del curso, se brindará formación específica para entrevistas técnicas, incluidas simulaciones de entrevistas de código en tiempo real y consejos prácticos para destacar en los procesos de selección de personal en empresas tecnológicas.

La combinación de clases interactivas en vivo, proyectos prácticos, expertos de la industria y una preparación enfocada en el mercado laboral asegura que los estudiantes no solo adquieran conocimientos técnicos avanzados, sino que también estén preparados para enfrentar los desafíos del desarrollo de software en el mundo real. ¡Estamos emocionados de acompañarte en esta travesía hacia el dominio completo de Python y el desarrollo de software!



dev Senior Code

