COURSE SYLLABUS

Python Developer Certificate (Self-Paced)

Build the skills needed to become a Python Developer, moving from Python programming fundamentals through web development with Django and Django REST. You'll develop in-demand skills with Django, a popular Python web framework used for back-end web development, and put together a portfolio of projects with guidance from the instructor.

Group classes (Live Online) and onsite training are available for this course. For more information, email corporate@nobledesktop.com or visit: https://www.nobledesktop.com/certificates/python-developer-self-paced



hello@nobledesktop.com • (212) 226-4149

Course Outline

This package includes these courses

- Python Programming Bootcamp (Self-Paced) (30 Hours)
- Python Web Development with Django (Self-Paced) (60 Hours)
- Python Developer Capstone Projects (18 Hours)

Choose two of the classes below as free electives (contact us after registration).

- Python for Data Science Course Online (Self-Paced)
- Python for AI Course Online (Self-Paced)
- Python Data Visualization & Interactive Dashboards Online (Self-Paced)
- Python Machine Learning Course Online (Self-Paced)

Python Programming Bootcamp (Self-Paced)

Learn Python programming from the ground up, building a strong foundation in core concepts, real-world applications, and coding best practices. This course prepares you to solve technical problems with confidence while developing Python projects you can showcase in a professional portfolio.

- Learn Python fundamentals such as variables, data types, loops, and functions
- Strengthen logical programming skills using conditionals, indexing, and slicing
- Work with files by reading, writing, and appending data in text files
- Explore computer science basics, including Big-O notation and common sorting algorithms
- Build scalable applications using object-oriented programming (OOP) principles
- Use Git and GitHub to manage code, track changes, and create a polished coding portfolio

Python Web Development with Django (Self-Paced)

Learn to build and deploy web applications using Django, a powerful Python web framework. This course covers back-end development, API creation, and automation techniques for scalable applications.

- Set up Django projects and work with URL dispatchers, templates, and database models.
- Develop full-stack web applications using Django's MVC architecture and generic views.
- Implement authentication, user management, and secure API endpoints with Django REST.
- Automate tasks with Django, including background processes and scheduled jobs.
- Work with APIs, including building and consuming RESTful web services.
- Deploy Django applications and APIs to production environments for real-world use.

Python Developer Capstone Projects

Python Web Development with Django Capstone

- Analyze structured datasets (e.g., Stack Overflow Developer Survey) to identify global tech salary trends influenced by factors like location, job role, experience level, and remote work.
- Use pandas for data cleaning and preprocessing (handling missing values, currency conversions, and outliers), and visualize key insights with Matplotlib/seaborn charts, including salary comparisons and remote versus on-site trends.
- Develop an interactive Django web application featuring dynamic salary filtering, interactive charts, and dashboards, using Django models, views, templates, and Bootstrap for front-end styling.

Python for AI Capstone *(Choose One of Two)*

- AI Chat Assistant: Build a Flask and JavaScript-powered AI chat assistant embedded in a live website, integrating OpenAI's API to provide context-aware responses about products or services with client-side session memory.
- Collectibles Identification Web App: Develop an interactive Flask-based web app allowing users to upload images of collectibles, utilizing OpenAI's API for item identification, metadata generation, and dynamic session-based item logging.
- Prepare a professional presentation demonstrating your app's user flow, technical integration (Flask, JavaScript, OpenAI), prompt engineering strategies, insights, limitations, and opportunities for future improvements.

See examples of Python capstone projects from students.