UDACITY
FOR ENTERPRISE

THE SCHOOL OF DATA SCIENCE

Programming for Data Science with Python

NANODEGREE SYLLABUS

Overview

This Nanodegree Degree Program is Built in Collaboration with

MODE

The Programming for Data Science with Python Nanodegree program offers you the opportunity to learn the most important programming languages used by data scientists today. Get your start into the fascinating field of data science and learn Python, SQL, terminal, and git with the help of experienced instructors. You will emerge prepared to tackle real world data analysis problems.

The Programming for Data Science with Python Nanodegree program is comprised of content and curriculum to support three projects.

Program Information

TIME
3 months
Study 10 hours/week

LEVEL
Foundational

PREREQUISITES
Ability to perform basic operations on your computer like opening files and folders, opening applications, and copying & pasting. You should also be able to read, write, and listen in English.
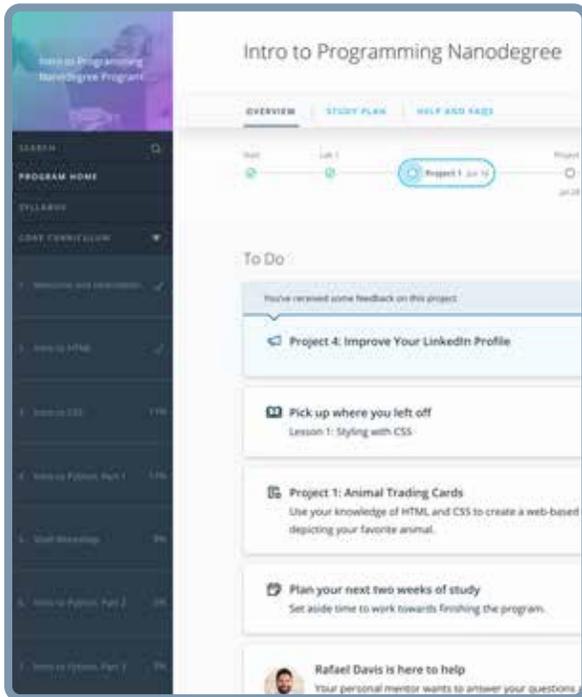
HARDWARE/SOFTWARE REQUIRED
Access to the internet and a 64-bit computer.

LEARN MORE ABOUT THIS NANODEGREE
Contact us at enterpriseNDs@udacity.com.

Our Classroom Experience

REAL-WORLD PROJECTS
Learners build new skills through industry-relevant projects and receive personalized feedback from our network of 900+ project reviewers. Our simple user interface makes it easy to submit projects as often as needed and receive unlimited feedback.

KNOWLEDGE
Answers to most questions can be found with Knowledge, our proprietary wiki. Learners can search questions asked by others and discover in real-time how to solve challenges.
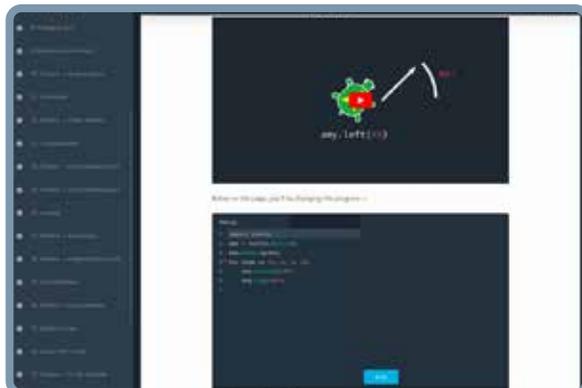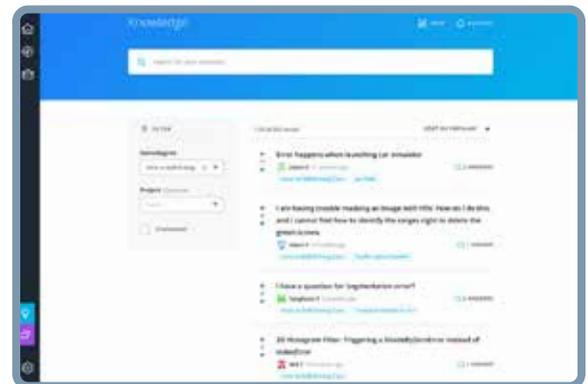
LEARNER HUB
Learners leverage the power of community through a simple, yet powerful chat interface built within the classroom. Learner Hub connects learners with their technical mentor and fellow learners.

WORKSPACES
Learners can check the output and quality of their code by testing it on interactive workspaces that are integrated into the classroom.

QUIZZES
Understanding concepts learned during lessons is made simple with auto-graded quizzes. Learners can easily go back and brush up on concepts at anytime during the course.

CUSTOM STUDY PLANS
Mentors create a custom study plan tailored to learners' needs. This plan keeps track of progress toward learner goals.

PROGRESS TRACKER
Personalized milestone reminders help learners stay on track and focused as they work to complete their Nanodegree program.

Learn with the Best

Josh Bernhard

DATA SCIENTIST AT NERD WALLET

Josh has been sharing his passion for data for nearly a decade at all levels of university, and as Lead Data Science Instructor at Galvanize. He's used data science for work ranging from cancer research to process automation.

Derek Steer

CEO AT MODE

Derek is the CEO of Mode Analytics. He developed an analytical foundation at Facebook and Yammer and is passionate about sharing it with future analysts. He authored SQL School and is a mentor at Insight Data Science.

Juno Lee

CURRICULUM LEAD AT UDACITY

Juno is the curriculum lead for the School of Data Science. She has been sharing her passion for data and teaching, building several courses at Udacity. As a data scientist, she built recommendation engines, computer vision and NLP models, and tools to analyze user behavior.

Richard Kalehoff

INSTRUCTOR

Richard is a Course Developer with a passion for teaching. He has a degree in computer science, and first worked for a nonprofit doing everything from front end web development, to backend programming, to database and server management.

Karl Krueger

COMMAND LINE INSTRUCTOR

Karl is a Course Developer at Udacity. Before joining Udacity, Karl was a Site Reliability Engineer (SRE) at Google for eight years, building automation and monitoring to keep the world's busiest web services online.

Course 1: Introduction to SQL

The first course will teach you the fundamentals of SQL such as JOINs, Aggregations, and Subqueries. Learn how to use SQL to answer complex business problems.

Project	Investigate a Database

In this project, you'll work with a relational database while working with PostgreSQL. You'll complete the entire data analysis process, starting by posing a question, running appropriate SQL queries to answer your questions and finishing by sharing your findings.

LESSON TITLE	LEARNING OUTCOMES
BASIC SQL	· Write common SQL commands including SELECT, FROM, and WHERE. · Learn how to use logical operators like LIKE, AND, and OR.
SQL JOINS	· Write JOINs in SQL, as you are now able to combine data from multiple sources to answer more complex business questions. · Understand different types of JOINs and when to use each type.
SQL AGGREGATIONS	· Write common aggregations in SQL including COUNT, SUM, MIN, and MAX. · Write CASE and DATE functions, as well as work with NULLs.
ADVANCED SQL QUERIES	· Use subqueries, also called CTEs, in a number of different situations. · Use other window functions including RANK, NTILE, LAG, LEAD along with partitions to complete complex tasks.

Course 2: Introduction to Python Programming

In this part, you'll learn to represent and store data using Python data types and variables, and use conditionals and loops to control the flow of your programs. You'll harness the power of complex data structures like lists, sets, dictionaries, and tuples to store collections of related data. You'll define and document your own custom functions, write scripts, and handle errors. You will also learn to use two powerful Python libraries - Numpy, a scientific computing package, and Pandas, a data manipulation package.

Project	Explore US Bikeshare Data

You will use Python to answer interesting questions about bikeshare trip data collected from three US cities. You will write code to collect the data, compute descriptive statistics, and create an interactive experience in the terminal that presents the answers to your questions.

LESSON TITLE	LEARNING OUTCOMES
WHY PYTHON PROGRAMMING	· Gain an overview of what you'll be learning and doing in the course. · Understand why you should learn programming with Python.
DATA TYPES AND OPERATORS	· Represent data using Python's data types: integers, floats, booleans, strings, lists, tuples, sets, dictionaries, compound data structures. · Perform computations and create logical statements using Python's operators: Arithmetic, Assignment, Comparison, Logical, Membership, Identity. · Declare, assign, and reassign values using Python variables. · Modify values using built-in functions and methods. · Practice whitespace and style guidelines.
CONTROL FLOW	· Write conditional expressions using if statements and boolean expressions to add decision making to your Python programs. · Use for and while loops along with useful built-in functions to iterate over and manipulate lists, sets, and dictionaries. · Skip iterations in loops using break and continue. · Condense for loops to create lists efficiently with list comprehensions.

Course 2: Introduction to Python, cont.

LESSON TITLE	LEARNING OUTCOMES
FUNCTIONS	· Define your own custom functions. · Create and reference variables using the appropriate scope. · Add documentation to functions using docstrings. · Define lambda expressions to quickly create anonymous functions. · Use iterators and generators to create streams of data.
SCRIPTING	· Install Python 3 and set up your programming environment. · Run and edit python scripts. · Interact with raw input from users. · Identify and handle errors and exceptions in your code. · Open, read, and write to files. · Find and use modules in Python Standard Library and third-party libraries. · Experiment in the terminal using a Python Interpreter.
NUMPY	· Create, access, modify, and sort multidimensional NumPy arrays (ndarrays). · Load and save ndarrays. · Use slicing, boolean indexing, and set operations to select or change subsets of an ndarray. · Understand difference between a view and a copy of ndarray. · Perform element-wise operations on ndarrays. · Use broadcasting to perform operations on ndarrays of different sizes.
PANDAS	· Create, access, and modify the main objects in Pandas, Series and DataFrames. · Perform arithmetic operations on Series and DataFrames. · Load data into a DataFrame. · Deal with Not a Number (NaN) values.

Course 3: Introduction to Version Control

In this course, you will learn how to use version control and share your work with other people in the data science industry.

Project	Post your work on Github

In this project, you will learn important tools that all programmers use. First, you'll get an introduction to working in the terminal. Next, you'll learn to use git and Github to manage versions of a program and collaborate with others on programming projects. In this project you will add a completed project on GitHub, work with branches, edit a README file and project files, merge branches, stage and commit your changes to your project GitHub repository.

LESSON TITLE	LEARNING OUTCOMES
SHELL WORKSHOP	· The Unix shell is a powerful tool for developers of all sorts. Get a quick introduction to the basics of using it on your computer.
PURPOSE & TERMINOLOGY	· Learn why developers use version control and discover ways you use version control in your daily life. · Get an overview of essential Git vocabulary. · Configure Git using the command line.
CREATE A GIT REPO	· Create your first Git repository with git init. · Copy an existing Git repository with git clone. · Review the current state of a repository with the powerful git status.
REVIEW A REPO'S HISTORY	· Review a repo's commit history git log. · Customize git log's output using command line flags in order to reveal more (or less) information about each commit. · Use the git show command to display just one commit.

Course 3: Introduction to Version Control, cont.

LESSON TITLE	LEARNING OUTCOMES
ADD COMMITS TO A REPO	· Master the Git workflow and make commits to an example project. · Use git diff to identify parts of a file that changed in a commit. · Mark files as "untracked" using .gitignore.
TAGGING, BRANCHING, AND MERGING	· Discover tagging, branching, and merging and organize your commits with tags and branches. · Jump to particular tags and branches using git checkout. · Learn how to merge together changes on different branches and crush those pesky merge conflicts.
UNDOING CHANGES	· Learn how and when to edit or delete an existing commit. · Use git commit and amend flag to alter the last commit. · Use git reset and git revert to undo and erase commits.

Actual learner-submitted solutions to Nanodegree course project.

Our Nanodegree Programs Include:

Pre-Assessments

Our in-depth workforce assessments identify your team's current level of knowledge in key areas. Results are used to generate custom learning paths designed to equip your workforce with the most applicable skill sets.

Dashboard & Progress Reports

Our interactive dashboard (enterprise management console) allows administrators to manage employee onboarding, track course progress, perform bulk enrollments and more.

Industry Validation & Reviews

Learners' progress and subject knowledge is tested and validated by industry experts and leaders from our advisory board. These in-depth reviews ensure your teams have achieved competency.

Real World Hands-on Projects

Through a series of rigorous, real-world projects, your employees learn and apply new techniques, analyze results, and produce actionable insights. Project portfolios demonstrate learners' growing proficiency and subject mastery.

Our Review Process

Real-life Reviewers for Real-life Projects

Real-world projects are at the core of our Nanodegree programs because hands-on learning is the best way to master a new skill. Receiving relevant feedback from an industry expert is a critical part of that learning process, and infinitely more useful than that from peers or automated grading systems. Udacity has a network of over 900 experienced project reviewers who provide personalized and timely feedback to help all learners succeed.

Vaibhav
UDACITY LEARNER

"I never felt overwhelmed while pursuing the Nanodegree program due to the valuable support of the reviewers, and now I am more confident in converting my ideas to reality."

now at

CODING VISIONS INFOTECH

All Learners Benefit From:

Line-by-line feedback for coding projects	Industry tips and best practices	Advice on additional resources to research	Unlimited submissions and feedback loops

How it Works

Real-world projects are integrated within the classroom experience, making for a seamless review process flow.

- Go through the lessons and work on the projects that follow
- Get help from your technical mentor, if needed
- Submit your project work
- Receive personalized feedback from the reviewer
- If the submission is not satisfactory, resubmit your project
- Continue submitting and receiving feedback from the reviewer until you successfully complete your project

About our Project Reviewers

Our expert project reviewers are evaluated against the highest standards and graded based on learners' progress. Here's how they measure up to ensure your success.

900+	1.8M	3	4.85 /5
Expert Project Reviewers	**Projects Reviewed**	**Hours Average Turnaround**	**Average Reviewer Rating**
Are hand-picked to provide detailed feedback on your project submissions.	Our reviewers have extensive experience in guiding learners through their course projects.	You can resubmit your project on the same day for additional feedback.	Our learners love the quality of the feedback they receive from our experienced reviewers.

UDACITY
FOR ENTERPRISE

2440 W El Camino Real, #101
Mountain View, CA 94040, USA - HQ

For more information visit: **www.udacity.com/enterprise**