



NANODEGREE PROGRAM SYLLABUS

AI Programming with Python



Overview

This program focuses on the fundamental building blocks you will need to learn in order to become an AI practitioner. Specifically, you will learn programming skills, and essential math for building an AI architecture. You'll even dive into neural networks and deep learning.

One of our main goals at Udacity is to help you create a job-ready portfolio. Building a project is one of the best ways to test the skills you've acquired, and to demonstrate your newfound abilities to prospective employers. In this Nanodegree program you will test your ability to use a pre-trained neural network architecture, and also have the opportunity to prove your skills by building your own image classifier.

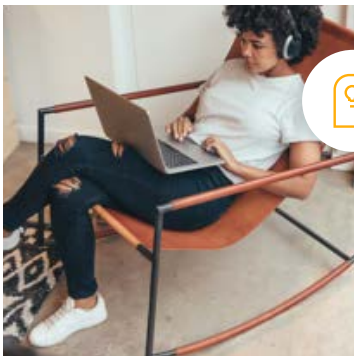
In the sections below, you'll find detailed descriptions of the projects, along with the course material that presents the skills required to complete them.



Estimated Time:
3 Months at
10 hrs/week



Prerequisites:
Basic Algebra and
Programming
Knowledge



Flexible Learning:
Self-paced, so
you can learn on
the schedule that
works best for you



Need Help?
[udacity.com/advisor](https://www.udacity.com/advisor)
Discuss this program
with an enrollment
advisor.

Course 1: Introduction to Python

Start coding with Python, drawing upon libraries and automation scripts to solve complex problems quickly.

Course Project

Using a Pre-trained Image Classifier to Identify Dog Breeds

In this Project you will be testing your newly acquired python coding skills by using a trained image classifier. You will need to use the trained neural network to classify images of dogs (by breeds) and compare the output with the known dog breed classification. You will have a chance to build your own functions, use command line arguments, test the runtime of the code, create a dictionary of lists, and more.

LEARNING OUTCOMES

LESSON ONE

Why Python Programming

- Learn why we program.
- Prepare for the course ahead with a detailed topic overview.
- Understand how programming in Python is unique.

LESSON TWO

Data Types and Operators

- Understand how data types and operators are the building blocks for programming in Python.
- Use the following data types: integers, floats, booleans, strings, lists, tuples, sets, dictionaries.
- Use the following operators: arithmetic, assignment, comparison, logical, membership, identity.

LESSON THREE

Control Flow

- Implement decision-making in your code with conditionals.
- Repeat code with for and while loops.
- Exit a loop with break, and skip an iteration of a loop with continue.
- Use helpful built-in functions like zip and enumerate.
- Construct lists in a natural way with list comprehensions.

LEARNING OUTCOMES

LESSON FOUR

Functions

- Write your own functions to encapsulate a series of commands.
- Understand variable scope, i.e., which parts of a program variables can be referenced from.
- Make functions easier to use with proper documentation.
- Use lambda expressions, iterators, and generators.

LESSON FIVE

Scripting

- Write and run scripts locally on your computer.
- Work with raw input from users.
- Read and write files, handle errors, and import local scripts.
- Use modules from the Python standard library and from third-party libraries.
- Use online resources to help solve problems.

LESSON SIX

Classes

- Object Oriented programming provides a few benefits over procedural programming. Learn the basics by understanding how to use Classes.



Course 2: Anaconda, Jupyter Notebook, NumPy, Pandas, and Matplotlib

Learn how to use all the key tools for working with data in Python: Jupyter Notebooks, NumPy, Anaconda, Pandas, and Matplotlib.

LEARNING OUTCOMES

LESSON ONE

Anaconda

- Learn how to use Anaconda to manage packages and environments for use with Python.

LESSON TWO

Jupyter Notebooks

- Learn how to use Jupyter Notebooks to create documents combining code, text, images, and more.

LESSON THREE

NumPy Basics

- Learn the value of NumPy and how to use it to manipulate data for AI problems.
- Mini-Project: Use NumPy to mean normalize an ndarray and separate it into several smaller ndarrays.

LESSON FOUR

Pandas Basics

- Learn to use Pandas to load and process data for machine learning problems.
- Mini-Project: Use Pandas to plot and get statistics from stock data.

LESSON FIVE

Matplotlib Basics

- Learn how to use Matplotlib to choose appropriate plots for one and two variables based on the types of data you have.

Course 3: Linear Algebra Essentials

Learn the foundational linear algebra you need for AI success: vectors, linear transformations, and matrices—as well as the linear algebra behind neural networks.

LEARNING OUTCOMES

LESSON ONE

Introduction

- Learn the basics of the beautiful world of Linear Algebra and learn why it is such an important mathematical tool.

LESSON TWO

Vectors

- Learn about the basic building block of Linear Algebra.

LESSON THREE

Linear Combination

- Learn how to scale and add vectors and how to visualize them in 2 and 3 dimensions.

LESSON FOUR

Linear Transformation and Matrices

- Learn what a linear transformation is and how is it directly related to matrices. Learn how to apply the math and visualize the concept.

LESSON FIVE

Linear Algebra in Neural Networks

- Learn about the world of Neural Networks and see how it relates directly to Linear Algebra.

LESSON SIX

Labs

- VECTORS LAB - Learn how to graph 2D and 3D vectors.
- LINEAR COMBINATION LAB - Learn how to computationally determine a vector's span and solve a simple system of equations.
- LINEAR MAPPING LAB - Learn how to solve problems computationally using vectors and matrices.

Course 4: Calculus Essentials

Learn the foundations of calculus to understand how to train a neural network: plotting, derivatives, the chain rule, and more. See how these mathematical skills visually come to life with a neural network example.

LEARNING OUTCOMES		
LESSON ONE	Introduction	<ul style="list-style-type: none"> • Visualize the essence of calculus. Learn why it is such a powerful concept in mathematics
LESSON TWO	Derivatives Through Geometry	<ul style="list-style-type: none"> • Learn about the derivative, one of the most important tools in calculus. • See how a derivative can measure the steepness of a function and why it is such an important indicator in the world of machine learning.
LESSON THREE	Chain Rule and Dot Product	<ul style="list-style-type: none"> • Learn how to find the derivative of a composition of two or more functions, a very important tool in training a neural network.
LESSON FOUR	More on Derivatives	<ul style="list-style-type: none"> • Learn more about derivatives while focusing on exponential and implicit functions..
LESSON FIVE	Limits	<ul style="list-style-type: none"> • Learn about the formal definition of a derivative through understanding limits.
LESSON SIX	Integration	<ul style="list-style-type: none"> • Learn about the inverse of a derivative: the integral.
LESSON SEVEN	Calculus in Neural Networks	<ul style="list-style-type: none"> • Learn more about the world of neural networks and see how it relates directly to calculus through an explicit example..

Course 5: Neural Networks

Gain a solid foundation in the hottest fields in AI: neural networks, deep learning, and PyTorch.

Course Project

Create Your Own Image Classifier

Successful software developers need to know how to incorporate deep learning models into everyday applications. Any device with a camera will be using image classification, object detection, and face recognition, all based on deep learning models. In this project you will implement an image classification application. This application will train a deep learning model on a dataset of images. It will then use the trained model to classify new images. First you will develop your code in a Jupyter notebook to ensure your training implementation works well. Then, you will convert your code into a Python application that you will run from the command line of your system.

LEARNING OUTCOMES

LESSON ONE

Introduction to Neural Networks

- Acquire a solid foundation in deep learning and neural networks. Implement gradient descent and backpropagation in Python.

LESSON TWO

Training Neural Networks

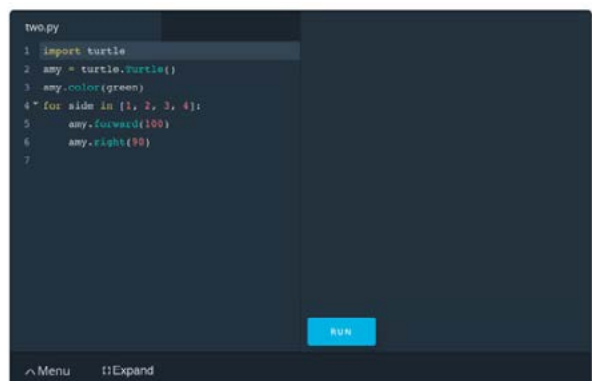
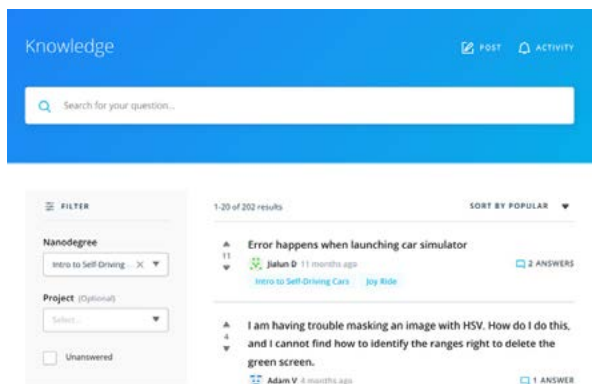
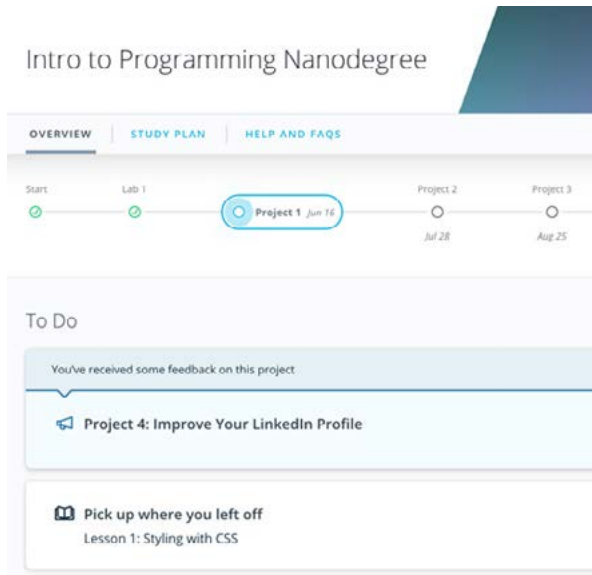
- Learn about techniques for how to improve training of a neural network, such as: early stopping, regularization and dropout.

LESSON THREE

Deep Learning with Pytorch

- Learn how to use PyTorch for building deep learning models.

Our Classroom Experience



REAL-WORLD PROJECTS

Build your skills through industry-relevant projects. Get personalized feedback from our network of 900+ project reviewers. Our simple interface makes it easy to submit your projects as often as you need and receive unlimited feedback on your work.

KNOWLEDGE

Find answers to your questions with Knowledge, our proprietary wiki. Search questions asked by other students, connect with technical mentors, and discover in real-time how to solve the challenges that you encounter.

WORKSPACES

See your code in action. Check the output and quality of your code by running them on workspaces that are a part of our classroom.

QUIZZES

Check your understanding of concepts learned in the program by answering simple and auto-graded quizzes. Easily go back to the lessons to brush up on concepts anytime you get an answer wrong.

CUSTOM STUDY PLANS

Create a custom study plan to suit your personal needs and use this plan to keep track of your progress toward your goal.

PROGRESS TRACKER

Stay on track to complete your Nanodegree program with useful milestone reminders.

Learn with the Best



Ortal Arel

CURRICULUM LEAD

Ortal Arel has a PhD in Computer Engineering, and has been a professor and researcher in the field of applied cryptography. She has worked on design and analysis of intelligent algorithms for high-speed custom digital architectures.



Luis Serrano

INSTRUCTOR

Luis was formerly a Machine Learning Engineer at Google. He holds a PhD in mathematics from the University of Michigan, and a Postdoctoral Fellowship at the University of Quebec at Montreal.



Jennifer Staab

INSTRUCTOR

Jennifer has a PhD in Computer Science and a Masters in Biostatistics; she was a professor at Florida Polytechnic University. She previously worked at RTI International and United Therapeutics as a statistician and computer scientist.



Juan Delgado

CONTENT DEVELOPER

Juan is a computational physicist with a Masters in Astronomy. He is finishing his PhD in Biophysics. He previously worked at NASA developing space instruments and writing software to analyze large amounts of scientific data using machine learning techniques.

Learn with the Best



Grant Sanderson

INSTRUCTOR

Grant Sanderson is the creator of the YouTube channel 3Blue1Brown, which is devoted to teaching math visually, using a custom-built animation tool. He was previously a content creator for Khan Academy.



Mat Leonard

INSTRUCTOR

Mat is a former physicist, research neuroscientist, and data scientist. He did his PhD and Postdoctoral Fellowship at the University of California, Berkeley.



Mike Yi

INSTRUCTOR

Mike is a Content Developer with a BS in Mathematics and Statistics. He received his PhD in Cognitive Science from the University of Irvine. Previously, he worked on Udacity's Data Analyst Nanodegree program as a support lead.



Juno Lee

CONTENT DEVELOPER

As a data scientist at Looplist, Juno built neural networks to analyze and categorize product images, a recommendation system to personalize shopping experiences for each user, and tools to generate insight into user behavior.

All Our Nanodegree Programs Include:



EXPERIENCED PROJECT REVIEWERS

REVIEWER SERVICES

- Personalized feedback & line by line code reviews
- 1600+ Reviewers with a 4.85/5 average rating
- 3 hour average project review turnaround time
- Unlimited submissions and feedback loops
- Practical tips and industry best practices
- Additional suggested resources to improve



TECHNICAL MENTOR SUPPORT

MENTORSHIP SERVICES

- Questions answered quickly by our team of technical mentors
- 1000+ Mentors with a 4.7/5 average rating
- Support for all your technical questions



PERSONAL CAREER SERVICES

CAREER SUPPORT

- Resume support
- Github portfolio review
- LinkedIn profile optimization



Frequently Asked Questions

PROGRAM OVERVIEW

WHY SHOULD I ENROLL?

AI-powered increases in safety, productivity, and efficiency are already improving our world, and the best is yet to come! As it becomes increasingly evident how impactful AI can be, demand for employees with AI skills increases—demand is in fact already skyrocketing.

The AI Programming with Python Nanodegree program makes it easy to learn the in-demand skills employers are looking for. You'll learn foundational AI programming tools (Python, NumPy, PyTorch) and the essential math skills (linear algebra and calculus) that will enable you to start building your own AI applications in just three months.

Whether you're seeking a full-time role in an AI-related field, want to start applying AI solutions in your current role, or simply want to start learning the defining technology of our time, this is the perfect place to get started.

WHAT JOBS WILL THIS PROGRAM PREPARE ME FOR?

While this is an introductory course that is not designed to prepare you for a specific job, after completing this program, you should be proficient in the skills used in the AI Industry, including but not limited to Python, machine learning, etc. If you wish to prepare for a full-time AI-related career, we recommend enrolling in our Machine Learning Engineer Nanodegree program next.

HOW DO I KNOW IF THIS PROGRAM IS RIGHT FOR ME?

Learning to program with Python, one of the most widely used languages in Artificial Intelligence, is the core of this program. You'll also focus on neural networks—AI's main building blocks. By learning foundational AI and math skills, you lay the groundwork for advancing your career—whether you're just starting out, or readying for a full-time role.

ENROLLMENT AND ADMISSION

DO I NEED TO APPLY? WHAT ARE THE ADMISSION CRITERIA?

No. This Nanodegree program accepts all applicants regardless of experience and specific background.

WHAT ARE THE PREREQUISITES FOR ENROLLMENT?

Formal prerequisites include basic knowledge of algebra and basic programming in any language. You will also need to be able to communicate fluently and professionally in written and spoken English.



FAQs Continued

TUITION AND TERM OF PROGRAM

HOW IS THIS NANODEGREE PROGRAM STRUCTURED?

The AI Programming with Python Nanodegree program is comprised of content and curriculum to support two (2) projects. We estimate that students can complete the program in three (3) months working 10 hours per week.

Each project will be reviewed by the Udacity reviewer network. Feedback will be provided and if you do not pass the project, you will be asked to resubmit the project until it passes.

HOW LONG IS THIS NANODEGREE PROGRAM?

Access to this Nanodegree program runs for the length of time specified in the payment card above. If you do not graduate within that time period, you will continue learning with month to month payments. See the [Terms of Use](#) and [FAQs](#) for other policies regarding the terms of access to our Nanodegree programs.

I HAVE GRADUATED FROM THE AI PROGRAMMING WITH PYTHON PROGRAM BUT I WANT TO KEEP LEARNING. WHERE SHOULD I GO FROM HERE?

Our graduates can continue on to our [Machine Learning](#), [Deep Learning](#), or [Artificial Intelligence Nanodegree](#) programs.

SOFTWARE AND HARDWARE

WHAT SOFTWARE AND VERSIONS WILL I NEED IN THIS PROGRAM?

We'll teach you how to install all the software required. Virtually any 64-bit operating system with at least 8GB of RAM will be suitable. Udacity does not provide any hardware.

