edureka!
Discover Learning

PYTHON DJANGO TRAINING & CERTIFICATION

Course Curriculum : Your 10 module Learning Plan

https://www.edureka.co/python-django

About Edureka

Edureka is a leading e-learning platform providing live instructor-led interactive online training. We cater to professionals and students across the globe in categories like Big Data & Hadoop, Business Analytics, NoSQL Databases, Java & Mobile Technologies, System Engineering, Project Management and Programming. We have an easy and affordable learning solution that is accessible to millions of learners. With our students spread across countries like the US, India, UK, Canada, Singapore, Australia, Middle East, Brazil and many others, we have built a community of over 1 million learners across the globe.

About Course

Edureka's Django course helps you gain expertise in Django REST framework, Django Models, Django AJAX, Django jQuery etc. You'll master Django web framework while working on real-time use cases and receive Django certification at the end of the course.

Curriculum

Introduction to Python

Goal : Give brief idea of what Python is and touch on basics.

Objectives:

- Define Python
- Know why Python is popular
- Setup Python environment
- Discuss flow control
- Write your first Python program

Topics:

- Get an overview of Python
- Learn about Interpreted Languages
- List the Advantages/Disadvantages of Python
- Explore Pydoc
- Start Python
- Discuss Interpreter PATH
- Use the Interpreter
- Run a Python Script
- Discuss Python Scripts on UNIX/Windows
- Explore Python Editors and IDEs
- Use Variables, Keywords, Built-in Functions, Strings, Different literals, Math operators and expressions, Writing to the screen, String formatting, Command line parameters and Flow Control.

Hands On:

- Data types - string, numbers, dates
- Keywords
- Variables
- Literals

Sequences and File Operations

Goal : Learn different types of sequence structures, related operations and their usage. Also learn diverse ways of opening, reading, and writing to files.

Objectives:

- Define Reserved Keywords and Command Line Arguments
- Describe how to Get User Input from Keyboard
- Describe Flow Control and Sequences
- Practice Working with Files
- Define and Describe Dictionaries and Sets

Topics:

- Lists
- Tuples
- Indexing and Slicing
- Iterating through a sequence
- Functions for all sequences
- Using enumerate()
- Operators and keywords for sequences
- The xrange()function

- List comprehensions
- Generator expressions
- Dictionaries and sets.
- Working with files
- Modes of opening a file
- File attributes
- File methods

Hands On:

- List - properties, related operations
- Tuple - properties, related operations, comparison with list
- Dictionary - properties, related operations, comparison with list
- Set - properties, related operations, comparison with dictionary

Deep Dive – Functions, Sorting, Errors and Exception, Regular Expressions and Packages

Goal : Learn how to create generic python scripts, how to address errors/exceptions in code and finally how to extract/filter content using regex.

Objectives:

- Explain Functions and various forms of Function Arguments
- Explain Standard Library
- Define Modules
- Describe Zip Archives and Packaging

Topics:

- Functions
- Function Parameters
- Global variables
- Variable scope and Returning Values
- Sorting
- Alternate Keys
- Lambda Functions
- Sorting collections of collections
- Sorting dictionaries
- Sorting lists in place
- Errors and Exception Handling
- Handling multiple exceptions
- The standard exception hierarchy using Modules
- The Import statement
- Module search path
- Package installation waysModule Aliases and Regular Expressions

Hands On / Demo :

- Functions - syntax, arguments, keyword arguments, return values
- Lambda - features, syntax, options, comparison with functions
- Sorting - sequences, dictionaries, limitations of sorting
- Errors and exceptions - types of issues, remediation
- Packages and module - modules, import options, sys path

Object Oriented Programming in Python

Goal : Understand the Object-Oriented Programming world in Python and use of standard libraries.

Objectives:

- Implement Regular Expression and its Basic Functions
- Use Classes, Objects, and Attributes
- Develop applications based on Object Oriented Programming and Methods

Topics:

- The sys Module
- Interpreter information
- STDIO
- Launching external programs
- Paths
- Directories and filenames
- Walking directory trees
- Math Function
- Random Numbers
- Dates and Times
- Zipped Archives
- Introduction to Python Classes
- Defining Classes
- Initializes
- Instance methods
- Properties
- Class methods and data
- Static methods
- Private methods and Inheritance

Hands On:

- Regular expressions - regex library, search/match object, findall, sub, compile
- Classes - classes and objects, access modifiers, instance and class members
- OOPS paradigm - Inheritance, Polymorphism and Encapsulation in Python

Debugging, Databases and Project Skeletons

Goal : Learn how to debug, how to use databases and how a project skeleton looks like in Python.

Objectives:

- Debug python scripts using pdb
- Debug python scripts using IDE
- Classify Errors
- Develop Unit Tests
- Create project Skeletons
- Implement Database using SQLite
- Perform CRUD operations on SQLite database

Topics:

- Debugging
- Dealing with errors
- Using unit tests
- Project Skeleton
- Required packages
- Creating the Skeleton
- Project Directory
- Final Directory Structure

- Testing your set up
- Using the skeleton
- Creating a database with SQLite 3
- CRUD operations
- Creating a database object.

Hands On:

- Debugging - debugging options, logging, troubleshooting
- Unit testing - TDD, unittest library, assertions, automated testing
- Project skeleton - industry standard, configurations, sharable libraries
- RDBMS - Python for RDBMS, PEP 49, CRUD operations on Sqlite

Introduction to Django Web Framework

Goal : In this module, you will be introduced to Django and learn how to create views and perform URL mapping

Objectives :

- Explain Web Framework
- Explain MVC pattern
- Create a basic Django app
- Create Views
- Use HTTP request and response objects
- Use URLConf

Topics :

- Web development

- Introduction to Django Web Framework
- Features of Django
- Installing Django
- MVC model
- HTTP concepts
- Views
- URL Mapping

Hands On/Demo :

- Create a simple View using Django

Templates and Forms

Goal : In this module, you will learn how to create Templates and Forms in Django

Objectives :

- Explain the Django Template System
- Load Template Files
- Render Templates
- Create Forms
- Process Form Data
- Customize Form Field Validation

Topics :

- Django Template Language
- Utilities of Templates
- Creating Template Objects

- Tags, Variables and Filters
- Rendering Templates
- Template Inheritance
- Form Handling
- Form validation and Error Messages
- Form Display

Hands On/Demo :

- Create a Form that accepts personal data to from a user

Models and Dynamic Webpages

Goal : In this module, you will learn how to create Database Models and add Dynamic content to your webpages

Objectives :

- Define Database Models
- Use Model Fields
- Populate a Database, CRUD
- Use QuerySets for data retrieval
- Use jQuery and AJAX with Django to create Dynamic websites

Topics :

- Django Models
- Model Fields
- Model Inheritance
- CRUD on DB

- Primary keys and the Model
- Dynamic Webpages
- Toggle Hidden Content
- jQuery and AJAX integration

Hands On/Demo :

- Adding a Like button to a webpage

Serialization

Goal : In this module, you will learn how to serialize and deserialize data and create APIs

Objectives :

- Explain Data Serialization
- Use Django's REST Framework
- Use Serializers and Deserializers
- Use Model Serializers
- Use REST APIs

Topics :

- Serialization and Deserialization
- Django REST Framework
- Serializer class
- Model Serializers
- REST APIs

Hands On/Demo :

- Creating a REST API

Parsing XML and JSON with Python

Goal : In this module, you will learn how to parse data stored in XML & JSON formats using Python

Objectives :

- Explain XML and JSON file formats
- Explain XML-RPC
- Parse data stored in both XML and JSON formats
- Stores data in XML and JSON formats

Topics :

- XML-RPC
- XML, parsing object to XML and back
- JSON, parsing object to JSON and back

Hands On/Demo :

- Parse data stored in XML/JSON format to native Python type and vice-versa

Projects

What are the system requirements for this course?

The course has no special hardware requirements. It only requires a VM software (preferably VirtualBox) with an image of the Ubuntu OS in which Django 1.11 and Django's REST framework are installed. 4 GB Ram will be recommended.

How will I execute the practicals?

All practicals will be executed within the VM where an image of the Ubuntu OS in which Django 1.11 and Django's REST framework are installed and the solutions should be uploaded for evaluation.

In case if you face any issues, we have 24x7 Expert Support to help you with your queries.

Which case-studies will be a part of the course?

Create a Django Web App for the Following Case:

Pollster is an online web application on which different online polls are hosted and conducted. The application hosts various polls on any topic and votes are garnered online.

Only registered users can vote on the various polls hosted on the site. Each user can cast his/her vote only once on any given poll. The users can also create new polls of their choice on the site.

An admin site exists for the administrator to create and edit polls.

The following actions must be performed in order to complete this task:

- Create a Django project and an app within the project
- Design the right model for storing user and poll data
- Create the right views for viewing polls and poll results
- Optimize code using the Django template system