

DevNet Associate Certification

DEVASC 200-901

Unofficial handout

Disclaimer

This is not intended as an official guide in any way, but just a best-effort collection of resources, hints and tips to support learning about DevNet Associate learning path and Exam. There is no intended or induced liability by the authors for any of the content or consequences of use of the content. There is no intended promotion or agreement with any of the suggested 3rd party resources. In addition, this collection is provided as-is and will not be updated.

Please always refer to the official online Cisco documentation for full updated reference and information.

Cisco DevNet Associate certification (DEVASC 200-901)

- 20% Understanding and Using APIs
- 20% Infrastructure and Automation
- 15% Application Deployment and Security
- 15% Cisco Platforms and Development
- 15% Network Fundamentals
- 15% Software Development and Design

Exam Number:	200-901 DEVASC
First date to test:	February 24, 2020
Associated Certifications:	DevNet Associate - Developer Certification
Duration:	120 minutes
Available Languages:	English
Exam Registration:	Pearson VUE

<https://learningnetwork.cisco.com/community/certifications/devnet-associate/devasc>

Study Guide

- Programmability Concepts
 - Data Formats
 - API
 - API Protocols
- Development Track
 - Development Models
 - Python
 - Version Control
- Application Models
 - DevOps e CI/CD
 - Containers
- Infrastructure
 - Programmability model
 - Infrastructure as a code
 - Automation Protocols (NETCONF, RESTCONF, YANG)
 - Automation Tools (VIRL, Chef, Puppet, Ansible, NSO)
- Cisco Technology
 - Cisco Products API
- Network Fundamentals
 - Networking basics
 - Network functions

Exam Guide

- Topics
- Tips & Tricks
- Labs for Practice
- Test Exam Available

Study Guide

Resources – *The Golden Path*



- DevNet Associate Exam v1.0 (200-901) Topic Guide
 - <https://developer.cisco.com/certification/exam-topic-associate/>
- Developing Applications and Automating Workflows using Cisco Core Platforms (DEVASC) v1.0 (Account required)
 - <https://digital-learning.cisco.com/#/course/61907>
- Pluralsight DevAsc 200-901 Path (Subscription needed)
 - <https://app.pluralsight.com/paths/certificate/cisco-certified-devnet-associate-devasc-200-901>
- Cisco Certified Devnet Associate Devasc 200-901 Official Cert Guide (Inglese) – 28 giugno 2020
 - <https://www.ciscopress.com/store/cisco-certified-devnet-associate-devasc-200-901-official-9780136677338>
- Useful Cisco Live Sessions (*showing just one in a bunch*)
 - <https://www.ciscolive.com/c/dam/r/ciscolive/emea/docs/2018/pdf/LTRCRT-2700.pdf>
- DevNet Associate Exam Test (Account required)
 - <https://digital-learning.cisco.com/#/course/62399>

Resources on DevNet – 1/4

- REST & API

- <https://developer.cisco.com/learning/labs/what-are-rest-apis/step/1>
- <https://developer.cisco.com/learning/lab/getting-started-rest-apis/step/1>
- <https://developer.cisco.com/learning/lab/hands-on-postman/step/1>

- Git

- <https://developer.cisco.com/learning/lab/git-basic-workflows/step/1>
- <https://developer.cisco.com/learning/lab/git-intro/step/1>
- <https://developer.cisco.com/learning/lab/git-branching/step/1>
- <https://developer.cisco.com/learning/lab/git-servers/step/1>
- <https://github.com/CiscoDevNet>

Resources on DevNet – 2/4

- Python

- <https://developer.cisco.com/learning/lab/02-python-01-home-lab-python/step/1>
- <https://developer.cisco.com/learning/lab/python-primer-1/step/1>
- <https://developer.cisco.com/learning/lab/intro-to-python-primer-2/step/1>
- <https://developer.cisco.com/learning/labs/intro-python-part1/step/1>
- <https://developer.cisco.com/learning/labs/intro-python-part2/step/1>
- <https://developer.cisco.com/learning/lab/intro-to-python-parsing-json/step/1>
- <https://developer.cisco.com/learning/lab/coding-202-parsing-json/step/1>
- <https://developer.cisco.com/learning/lab/coding-201-parsing-xml/step/1>
- <https://developer.cisco.com/learning/lab/collab-spark-calling-apis-from-python-itp/step/1>
- <https://developer.cisco.com/learning/lab/collab-spark-rest-api-mission-itp/step/1>
- <https://github.com/gto76/python-cheatsheet>

Resources on DevNet – 3/4

- RESTCONF/NETCONF & YANG

- <https://www.youtube.com/watch?v=cooE3wZ7O4I>
- <https://www.youtube.com/watch?v=txf9M4Ud9yU>
- <https://developer.cisco.com/learning/modules/intro-device-level-interfaces/intro-yang/step/1>
- <https://developer.cisco.com/learning/lab/intro-restconf/step/1>
- <https://developer.cisco.com/learning/lab/intro-netconf/step/1>

- Ansible & NX-OS Automation

- https://developer.cisco.com/learning/tracks/nxos-programmability/sdx-intro-nxos/nxos-intro-01_overview/step/1
- https://developer.cisco.com/learning/modules/sdx-ansible-intro/ansible-02_ansible-intro/step/1
- <https://developer.cisco.com/learning/lab/ansible-overview/step/1>
- <https://developer.cisco.com/learning/lab/ansible-ios-modules/step/1>
- https://developer.cisco.com/learning/lab/ansible-03_ansible-hands-on/step/1
- <https://developer.cisco.com/docs/ios-xe/#!ansible-quick-start-guide>

Resources on DevNet – 4/4

- NSO

- https://developer.cisco.com/learning/tracks/get_started_with_nso

- Programmability CoE

- <https://developer.cisco.com/site/programmability-coe/>
- https://developer.cisco.com/docs/prog-coe_resources/#!awesome-learning/
- <https://developer.cisco.com/video/net-prog-basics/>

Resources (Build a Local DevBox)

- Environment Setup

- <https://developer.cisco.com/learning/modules/dev-setup/dev-what/step/1>
- <https://developer.cisco.com/learning/lab/dev-centos/step/1>
- <https://developer.cisco.com/learning/lab/dev-ubuntu/step/1>

- VirtualBOX

- <https://www.virtualbox.org/wiki/Downloads>

- CentOS

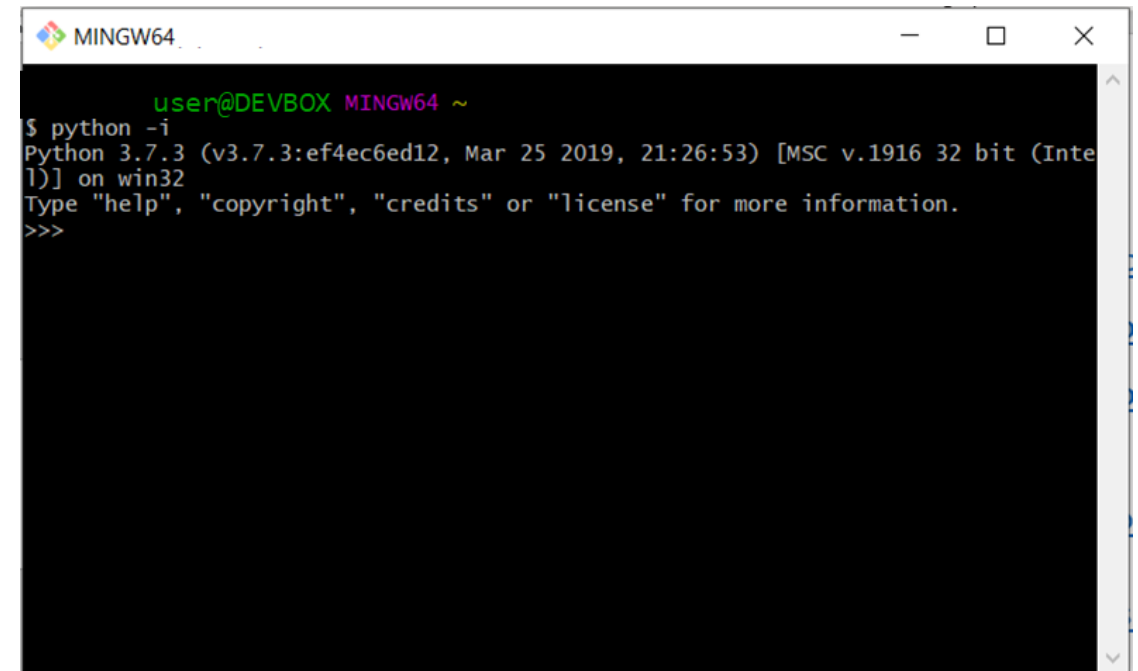
- http://isoredirect.centos.org/centos/8/isos/x86_64/
- <https://wiki.centos.org/HowTos/Virtualization/VirtualBox/CentOSguest>
- <https://developer.cisco.com/learning/modules/dev-setup/dev-centos/step/1>

- OS Images

- <https://www.osboxes.org/>
- <https://www.osboxes.org/centos/>
- <https://www.osboxes.org/ubuntu/>

- Git

- <https://git-scm.com/>
- <http://gitforwindows.org/>



```
MINGW64 ~
user@DEVBOX MINGW64 ~
$ python -i
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 21:26:53) [MSC v.1916 32 bit (Intel)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
```

More Resources

- REST & API

- Learning

- <https://www.youtube.com/watch?v=iFMLyMgCUTs&list=PLM-7VG-sgbtBBnWb2Jc5kufgtWYEmiMAw&index=1>

- Sandbox

- <https://www.getpostman.com/>

- Python

- Learning

- <https://www.pythonforbeginners.com/>
 - <https://www.practicepython.org/>

- Sandbox

- <http://pythontutor.com/>

Commercial Resources (Pluralsight)

- Nicholas Russo – YouTube channel
 - <https://www.youtube.com/watch?v=umaFfs0zsdo>
 - Study Guide (Excel Plan) for DevNet Core
 - <https://www.youtube.com/watch?v=AhPloufPDH8>
 - Study Guide (Excel Plan) for DevNet Associate
- Pluralsight (needed for above)
 - Annual Subscription needed
 - <https://app.pluralsight.com/paths/certificate/cisco-certified-devnet-associate-devasc-200-901>



Microsoft Excel
Worksheet



Microsoft Excel
Worksheet

Final Resources (for Certification)

- DevNet Certification
 - <https://developer.cisco.com/startnow/>
 - <https://developer.cisco.com/certification/exam-topic-associate>
- Pearson VUE On-line proctored Exams
 - <https://home.pearsonvue.com/cisco/onvue>

Exam Guide

Cisco DevNet Associate certification (DEVASC 200-901)

- 20% Understanding and Using APIs
- 20% Infrastructure and Automation
- 15% Application Deployment and Security
- 15% Cisco Platforms and Development
- 15% Network Fundamentals
- 15% Software Development and Design

Exam Number:	200-901 DEVASC
First date to test:	February 24, 2020
Associated Certifications:	DevNet Associate - Developer Certification
Duration:	120 minutes
Available Languages:	English
Exam Registration:	Pearson VUE

<https://learningnetwork.cisco.com/community/certifications/devnet-associate/devasc>

DEVASC 200-901 - Exam Content (1/3)

1.0 Software Development and Design

15%

- 1.1 Compare **data formats** (XML, JSON, and YAML)
- 1.2 Describe **parsing** of common data format (XML, JSON, and YAML) to Python data structures
- 1.3 Describe the concepts of test-driven development
- 1.4 Compare **software development methods** (agile, lean, and waterfall)
- 1.5 Explain the benefits of organizing code into **methods / functions, classes, and modules**
- 1.6 Identify the advantages of common **design patterns** (MVC and Observer)
- 1.7 Explain the advantages of **version control**
- 1.8 Utilize common version control operations with **Git**
 - 1.8.a Clone
 - 1.8.b Add/remove
 - 1.8.c Commit
 - 1.8.d Push / pull
 - 1.8.e Branch
 - 1.8.f Merge and handling conflicts
 - 1.8.g diff

2.0 Understanding and Using APIs

20%

- 2.1 Construct a **REST API** request to accomplish a task given API documentation
- 2.2 Describe common usage patterns related to **webhooks**
- 2.3 Identify the **constraints** when consuming APIs
- 2.4 Explain common **HTTP response codes** associated with REST APIs
- 2.5 **Troubleshoot** a problem given the HTTP response code, request and API documentation
- 2.6 Identify the parts of an **HTTP response** (response code, headers, body)
- 2.7 Utilize common **API authentication** mechanisms: basic, custom token, and API keys
- 2.8 Compare common **API styles** (REST, RPC, synchronous, and asynchronous)
- 2.9 Construct a **Python script** that calls a REST API using the requests library

DEVASC 200-901 - Exam Content (2/3)

3.0 Cisco Platforms and Development

15%

- 3.1 Construct a **Python script** that uses a Cisco SDK given SDK documentation
- 3.2 Describe the **capabilities of Cisco network management platforms and APIs** (Meraki, Cisco DNA Center, ACI, Cisco SD-WAN, and NSO)
- 3.3 Describe the **capabilities of Cisco compute management platforms and APIs** (UCS Manager, UCS Director, and Intersight)
- 3.4 Describe the **capabilities of Cisco collaboration platforms and APIs** (Webex Teams, Webex devices, Cisco Unified Communication Manager including AXL and UDS interfaces, and Finesse)
- 3.5 Describe the **capabilities of Cisco security platforms and APIs** (Firepower, Umbrella, AMP, ISE, and ThreatGrid)
- 3.6 Describe the device level **APIs and dynamic interfaces for IOS XE and NX-OS**
- 3.7 Identify the appropriate **DevNet resource** for a given scenario (Sandbox, Code Exchange, support, forums, Learning Labs, and API documentation)
- 3.8 Apply concepts of **model driven programmability** (YANG, RESTCONF, and NETCONF) in a Cisco environment
- 3.9 Construct **code to perform a specific operation** based on a set of requirements and given API reference documentation such as these:
 - 3.9.a Obtain a list of network devices by using Meraki, Cisco DNA Center, ACI, Cisco SD-WAN, or NSO
 - 3.9.b Manage spaces, participants, and messages in Webex Teams
 - 3.9.c Obtain a list of clients / hosts seen on a network using Meraki or Cisco DNA Center

4.0 Application Deployment and Security

15%

- 4.1 Describe **benefits of edge computing**
- 4.2 Identify attributes of different **application deployment models** (private cloud, public cloud, hybrid cloud, and edge)
- 4.3 Identify the attributes of these **application deployment types**
 - 4.3.a Virtual machines
 - 4.3.b Bare metal
 - 4.3.c Containers
- 4.4 Describe **components for a CI/CD pipeline** in application deployments
- 4.5 Construct a **Python unit test**
- 4.6 Interpret contents of a **Dockerfile**
- 4.7 Utilize **Docker images** in local developer environment
- 4.8 Identify **application security issues** related to secret protection, encryption (storage and transport), and data handling
- 4.9 Explain how **firewall, DNS, load balancers, and reverse proxy** in application deployment
- 4.10 Describe top **OWASP** threats (such as XSS, SQL injections, and CSRF)
- 4.11 Utilize **Bash** commands (file management, directory navigation, and environmental variables)
- 4.12 Identify the **principles of DevOps practices**

DEVASC 200-901 - Exam Content (3/3)

5.0 Infrastructure and Automation

20%

- 5.1 Describe the **value of model driven programmability** for infrastructure automation
- 5.2 Compare **controller-level to device-level management**
- 5.3 Describe the use and roles of **network simulation and test tools** (such as VIRL and pyATS)
- 5.4 Describe the **components and benefits of CI/CD pipeline** in infrastructure automation
- 5.5 Describe principles of **infrastructure as code**
- 5.6 Describe the capabilities of automation tools such as **Ansible, Puppet, Chef, and Cisco NSO**
- 5.7 Identify the **workflow being automated by a Python script** that uses Cisco APIs including ACI, Meraki, Cisco DNA Center, or RESTCONF
- 5.8 Identify the **workflow being automated by an Ansible playbook** (management packages, user management related to services, basic service configuration, and start/stop)
- 5.9 Identify the **workflow being automated by a bash script** (such as file management, app install, user management, directory navigation)
- 5.10 Interpret the results of a **RESTCONF or NETCONF** query
- 5.11 Interpret basic **YANG** models
- 5.12 Interpret a unified **diff**
- 5.13 Describe the principles and benefits of a **code review process**
- 5.14 Interpret **sequence diagram** that includes API calls

6.0 Network Fundamentals

15%

- 6.1 Describe the purpose and usage of **MAC addresses and VLANs**
- 6.2 Describe the purpose and usage of **IP addresses, routes, subnet mask / prefix, and gateways**
- 6.3 Describe the function of common **networking components** (such as switches, routers, firewalls, and load balancers)
- 6.4 Interpret a basic **network topology** diagram with elements such as switches, routers, firewalls, load balancers, and port values
- 6.5 Describe the function of **management, data, and control planes** in a network device
- 6.6 Describe the functionality of these IP Services: **DHCP, DNS, NAT, SNMP, NTP**
- 6.7 Recognize **common protocol port values** (such as, SSH, Telnet, HTTP, HTTPS, and NETCONF)
- 6.8 Identify **cause of application connectivity issues** (NAT problem, Transport Port blocked, proxy, and VPN)
- 6.9 Explain the **impacts of network constraints on applications**

Resources (Test and Labs)

- DevNet Associate Exam Test (Account required)
 - <https://digital-learning.cisco.com/#/course/62399>
- LAB & Sandbox
 - <https://devnetsandbox.cisco.com/RM/Topology>
 - <https://developer.cisco.com/netdevops/live/>

Exam Tips – #1

Getting ready for DEVASC - #1

How did I prepare for the exam?

I used mostly two resources:

- [Developing Applications and Automating Workflows using Cisco Core Platforms \(DEVASC\) v1.0](#)
- <https://developer.cisco.com/certification/exam-topic-associate/>

Personally I suggest to use the first of these links to study the “**theory**” and don’t spend too much time on the labs, as they are only click through, simply read the solution to every step and make sure you understand why that is done and why you have that outcome.

Studying the theory and going through the course and the material is good and all, but it might not be enough to have you pass the certification. Hands-on and practice is ***fundamental***.

Go through the DevNet tracks available here: [DevNet Express tracks](#). If you have time, **this is much more detailed and of course worth the extra effort.**

For those of you who don’t have experience with programming I strongly suggest to **get a *good* grasp on git**, as there will be questions about it, and not only on a high-level, but some detailed questions too.

Getting ready for DEVASC - #1

Do you have any tips or tricks for the exam?

As we all know, I can't share any details about the questions themselves, but what I can share with you is this:

- DO NOT underestimate the ***Cisco Platforms and APIs*** section, there is A LOT of material there, but it is worth to be studied and tried and understood before running on to the next chapter.
- The same goes for the ***Software development and design*** section. I personally have 5+ years of SW development experience, yet it took me quite some time to make some concepts here mine.
- Remember those ***really specific*** questions you had in CCNA / CCNP ***about that one detail*** that you hated so much? They are still there, so be ready.
- **TAKE YOUR TIME** to read the questions and the answers. They are not made to fool you, but sometimes the difference is so small that it might go unnoticed if you read quickly.
- Use your time wisely, you have **almost 3 hours** at your disposal (170 minutes), DO NOT run through the questions, but do not waste your time either. The time you are given is quite fair, make the best out of it (if you have a question where you have no idea what's the answer, don't waste time on it, pick a random one and go on).

This all being said, this is a certification that **can be achieved even with little to no Cisco related knowledge**. If you have 3+ years of SW development experience, you can get certified without too much issues.

Exam Tips – #2

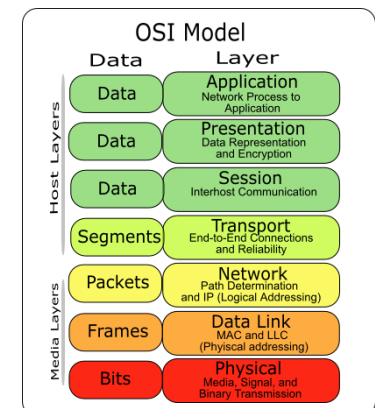
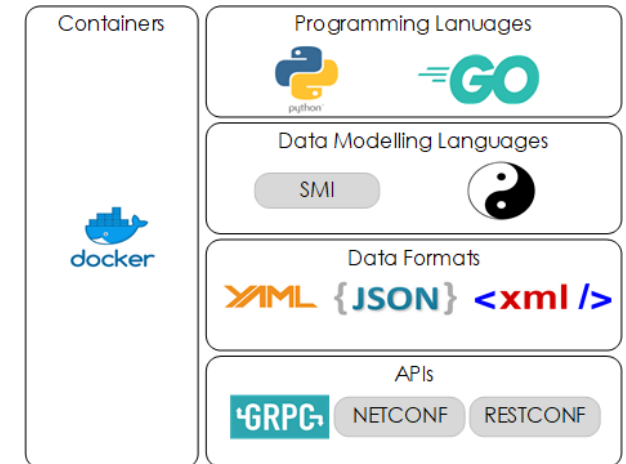
Getting ready for DEVASC – #2



Do you have any tips or tricks for the exam?

Useful Resources

- **Cisco Learning Library** (DEVASC) - <https://digital-learning.cisco.com/#/course/61907>
- **Cisco Live** sessions with tags: programmability and DEVNET
- **Cisco Documentation** for deeper understanding and clarification
- Learning labs that are thight with the exam curriculum
 - <https://developer.cisco.com/certification/exam-topic-associate/>
 - https://learningnetwork.cisco.com/s/learning-plan-detail-standard?ltui_urlRecordId=a1c3i0000005hsLAAQ<ui_urlRedirect=learning-plan-detail-standard
- Know extremely well the **difference between NETCONF e RESTCONF**
- Other **keywords**: JSON, XML, YAML, git, REST APIs, HTTP framework, Cisco APIs, Networking, Docker, Ansible, Software development approaches
- Get familiar with **Postman**
- Virtual enviroments: **VisualStudio or PyCharm, Sandboxes**
- Is not about programming, is more about **what can the code do for you**



Cisco API Collection

- Meraki API
 - <https://developer.cisco.com/meraki/api/#!/introduction/meraki-dashboard-api>
 - <https://documenter.getpostman.com/view/7928889/SVmsVg6K?version=latest#d96f78c6-ef92-45f7-b3fc-8f3f830461bf>
- Webex API
 - <https://documenter.getpostman.com/view/30210/webex-admin-api/2PMC7h?version=latest#4124911e-b77b-414b-bd0b-5bd141599e36>
 - <https://developer.webex.com/getting-started.html>
- DNA Center API
 - <https://developer.cisco.com/docs/dna-center/api/1-3-3-x/#!/intent-api-v1-3-3-x>
 - <https://developer.cisco.com/docs/dna-center/#!/using-the-cisco-dna-center-api-documentation/using-the-cisco-dna-center-api-documentation>
 - <https://documenter.getpostman.com/view/134222/SVmpZ2vT?version=latest#21eabff8-7e8b-46c8-9e65-33dad3f83994>
- Collection
 - <https://explore.postman.com/ciscodevnet>

Exam Tips – #3

Cisco OnVUE



OnVUE online proctored exams allow you to conveniently take an exam in the comfort of your home or office while being monitored by an offsite proctor. All communication with the proctor is done in English.

1 Run system test

Be sure to run your test on the same network and computer you will use on exam day.

Test my system

2 Schedule exam

Login to your account to schedule an OnVUE exam.

Sign in

[Create account](#)

3 Begin exam

When it is time to take your exam, login to your account to begin the exam.

Sign in



Now Available:
ONLINE
Certification Testing



On exam day:

We recommend logging into your account 30 minutes early to start the check-in process and to allow for any troubleshooting.

If you are more than 15 minutes late after your scheduled exam time you will be unable to begin your exam and are unlikely to receive a refund.

- Click on the Login button - <https://home.pearsonvue.com/cisco/onvue>
- Click on your scheduled exam under "Purchased Online Exams"
- Click "Begin Exam" and follow the on-screen prompts to complete the check-in process
- Once you have completed the check-in process you will be contacted by a Proctor to begin your exam

*Pluralsight - <https://app.pluralsight.com/library/>



SET A PERSONAL GOAL
Track your progress and stay motivated to hit your weekly goal.

30 minutes a week

- 10 minutes a week
- 20 minutes a week
- 30 minutes a week
- 40 minutes a week
- 50 minutes a week
- 60 minutes a week
- 2 hours a week

Introduction to the Cisco Certified DevNet Professional Certification
By Nick Russo

The DEVCOR Blueprint

- SW Dev/Design 20%
- Using APIs 20%
- Cisco platforms 20%
- App Deployment 20%
- Infra automation 20%

Table of Contents

- 1 Course Overview (1m 25s)
- 2 Steps to Success: What You Need to Know (18m 0s)

Getting Started with Software Development Using Cisco DevNet
By Nick Russo

Pillars of Good Coding

- Functional decomposition
- Error checking
- Usage of design patterns

Table of Contents

- 1 Course Overview (1m 33s)
- 2 Learning the Foundations of Software Design (43m 1s)
- Introducing Globomatics and Ot... (3m 48s)
- Demo: First Things First: Basic Ba... (8m 38s)
- Software Development Strategies (7m 26s)
- Understanding the Core Agile Ten... (2m 20s)
- The Three Pillars of Good Coding (3m 38s)
- The Power of Python pip and virt... (1m 26s)
- Demo: Setting Up a Workspace w... (3m 48s)
- The Observer Design Pattern (1m 22s)
- Demo: Homemade Observer Patt... (4m 56s)
- The Model View Controller (MVC)... (2m 31s)
- Demo: Simple MVC-based Flask ... (8m 9s)
- Module Review (0m 53s)

* subscription based

Grazie