http://bit.ly/LFprep

E-LEARNING: OPEN SOURCE SOFTWARE DEVELOPMENT, LINUX AND GIT

Once registered via the Coursera website, you can immediately access the courses in this specialization via your Coursera account.

You can download the welcome document for this course from here.

Coursera elearning.pdf

DOWNLOAD THIS TOOL AND VERIFY YOUR COMPUTER IS READY FOR AN LF COURSE

What do I need to be ready for the Coursera class?

Ready-for.sh v8.12

Open Source Software Development, Linux and Git

A list of HW and SW requirements for your Linux computer can be found at the bottom of this page. However, the following ready-for.sh script automates checking that your Linux computer meets minimum requirements, installs missing software, and downloads course material tarballs in order to prepare for class.

Running this before class, ideally somewhere with fast Internet, can save a lot of time during class time (we can't guarantee fast internet depending on where the class is taught).

This script only works on Linux. **It does not work on MacOS nor Windows.**

When you run the script, it might ask for a password. Enter your own login password. (If you're curious, it uses sudo - even installs and sets up sudo if needed).

1. Download ready-for.sh v8.12 (md5sum b288f9ff2ddb25c705f6173db1aee345)

```
wget http://bit.ly/LFready -O ready-for.sh
```

2. Make the script executable

```
chmod 755 ready-for.sh
```

```
./ready-for.sh --help
```

```
Usage: ready-for.sh [options] [course] [version]
    --distro              List current Linux distro
-i  --install             Install missing packages for the course
-f  --fix-missing         Skip uninstallable packages
-r  --remove [--all]      Remove installed packages for the course
-l  --list                List all supported courses
    --no-cache            Don't use previously cached output.
-C  --no-course-files     Don't install course files
-E  --no-extras           Don't download extra materials
-I  --no-install          Don't check installed packages
-R  --no-recommends       Don't install recommended packages
-S  --no-suggests         Don't install suggested packages
-N  --no-vm               Don't download virtual machine
    --update              Update to latest version of this script
    --verify              Verify script MD5sum
-V  --version             List script version
-v  --verbose             Turn on extra messages
-y  --yes                 Answer 'yes' to every question
-h  --help                What you just typed to see this
-H  --advanced-help       Even more esoteric options for debugging

Example: ready-for.sh --install Coursera

You can ask for a specific version of a course like this:
         ready-for.sh LFD460=V2.4.3
```

3. Run script with the appropriate six character course number (Coursera in the example below)

```
./ready-for.sh Coursera
```

4. If the previous step told you there were missing packages, run it with *--install* to download/install any missing packages (it will prompt you for your sudo password)

```
./ready-for.sh --install Coursera
```

IT SAYS SOMETHING WENT WRONG. WHAT DO I DO?

all ○Developer ○Embedded ○Instructor-led ○On-site ○Self-paced ○edX ○Coursera ○Sysadmin ○Virtual

Code	Available Self-Paced Courses
Coursera	Open Source Software Development, Linux and Git **(Coursera)**
LFC210	Fundamentals of Professional Open Source Management
LFD201	Intro to Open Source Development, Git, and Linux
LFD232	Cloud Foundry for Developers
LFD254	Containers for Developers and Quality Assurance
LFD259	Kubernetes for Developers
LFD271	Hyperledger Fabric Fundamentals
LFD272	Hyperledger Fabric Fundamentals for Developers
LFS101	Introduction to Linux **(edX)**
LFS103	Introduction to Apache Hadoop **(edX)**
LFS132	Introduction to Cloud Foundry and Cloud Native Software Architecture **(edX)**
LFS151	Introduction to Cloud Infrastructure Technologies **(edX)**
LFS152	Introduction to OpenStack **(edX)**
LFS153	Building Microservice Platforms with TARS **(edX)**
LFS157	Introduction to Serverless on Kubernetes **(edX)**
LFS158	Introduction to Kubernetes **(edX)**
LFS161	Introduction to DevOps: Transforming and Improving Operations **(edX)**
LFS163	Introduction to ONAP: Complete Network Automation **(edX)**
LFS164	NFV Acceleration: An Introduction to OPNFV **(edX)**
LFS165	Introduction to Open Source Networking Technologies **(edX)**
LFS170	Blockchain: Understanding Its Uses and Implications **(edX)**
LFS171	Introduction to Hyperledger Blockchain Technologies **(edX)**
LFS201	Essentials of System Administration
LFS205	Administering Linux on Azure
LFS211	Linux Networking and Administration
LFS216	Linux Security Fundamentals
LFS232	Cloud Foundry for Developers
LFS241	Monitoring Systems and Services with Prometheus
LFS242	Cloud Native Logging with Fluentd
LFS243	Service Mesh Fundamentals
LFS244	Managing Kubernetes Applications with Helm
LFS252	OpenStack Administration Fundamentals
LFS253	Containers Fundamentals
LFS258	Kubernetes Fundamentals
LFS261	Implementing Continuous Delivery
LFS263	ONAP Fundamentals
LFS264	OPNFV Fundamentals
LFS265	Software Defined Networking Fundamentals
LFS266	DevOps for Network Engineers
LFS267	Jenkins Essentials
LFS268	CI/CD with Jenkins X
LFS272	Hyperledger Fabric Administration
LFS273	Hyperledger Sawtooth Administration
LFW211	Node.js Application Development
LFW212	Node.js Services Development

Code	Available Instructor-Led Courses
LFD301	Introduction to Linux for Developers and GIT
LFD401	Developing Applications for Linux
LFD420	Linux Kernel Internals and Debugging
LFD430	Developing Linux Device Drivers
LFD432	Developing on Cloud Foundry
LFD435	Embedded Linux Device Drivers
LFD440	Linux Kernel Debugging and Security
LFD450	Embedded Linux Development
LFD459	Kubernetes for App Developers
LFD460	Building Embedded Linux with the Yocto Project
LFS300	Fundamentals of Linux
LFS301	Linux System Administration
LFS305	Deploying and Managing Linux on Azure
LFS311	Advanced Linux System Administration and Networking
LFS416	Linux Security
LFS422	High Availability Linux Architecture
LFS426	Linux Performance Tuning
LFS430	Linux Enterprise Automation
LFS452	Essentials of OpenStack Administration
LFS457	Advanced OpenStack Administration
LFS458	Kubernetes Administration
LFS462	Open Source Virtualization
LFS465	Software Defined Networking with OpenDaylight

WHAT ABOUT USING A VIRTUAL MACHINE FOR THE COURSE?

You can find prebuilt VM images for this course here.

Information on how to install the VMs, logins for the VMs, and more can be found here.

(The links may ask for a username/password which is in the document at the top of this page, or if you mouse over the link above)

ArchLinux CentOS-8 CentOS7 Debian-10 Fedora-32 Fedora-33 GENTOO openSUSE-Leap-15-2 Ubuntu-16-04 Ubuntu-18-04 Ubuntu-20-04 Ubuntu-20-10

For a more detailed explanation of all the possible methods of installation, please view a more comprehensive explanation here.

If your computer isn't powerful enough to run a virtual machine, you may elect to use a cloud service provider like AWS.

WHAT DOES MY LINUX COMPUTER NEED FOR THE COURSE?

These numbers apply to your Linux machine you will use for the course. **If you plan to use a Virtual Machine for this course, the following numbers apply to the Virtual Machine Guest, *NOT* the host operating system.**

Coursera: Open Source Software Development, Linux and Git Direct URL

Title	Open Source Software Development, Linux and Git
Attributes	Coursera,Self-paced
Internet Access	Required
OS required for class	Linux
Virtual Machine	Acceptable
Required CPU Architecture	x86_64
Preferred Number of CPUs	2 (minimum 1)
Minimum CPU Performance	2000 bogomips
Minimum Amount of RAM	1 GiB
Free Disk Space in $HOME	5 GiB
Free Disk Space in /boot	128 MiB
Distro Architecture	x86_64
Unsupported Linux Distros	CentOS-6-,Debian-8-,Fedora-30-,LinuxMint-18.3-,openSUSE-42.3-,OracleLinux,RHEL-6-,SLES-12.2-,Ubuntu-19.10-
Package List	— Select Distro —