

[Wiki](#) > [CS190.1x Frequently Asked ...](#) >[+ Add article](#)

CS190.1x Frequently Asked Questions

[View](#)[Edit](#)[Changes](#)

Where can students go for help with the labs or if they have other questions related to the course?

- We have created an extensive set of FAQs related to different aspects of the course. Students having trouble with VM setup, IPython notebooks (Jupyter), and/or the autograder, should first check the following FAQs:
 - [VM Setup FAQ](#)
 - [IPython notebook FAQ](#)
 - [Autograder FAQ](#)
 - [Databricks FAQ](#)
- If these FAQs do not address a specific question, students should look on [Piazza](#), which is a forum where students can ask questions and get support from other students and instructors. When using Piazza, students should follow these two guidelines:
 - Students should search through existing Piazza posts, as students often run into similar problems. It's much more efficient for both students and the instructors to not ask and answer the same questions multiple times.

LAST
MODIFIED:
June 28,
2015,
9:03 p.m.

[See all
children](#)

- Students should **not post** any coding answers or partial solutions for any of the lab exercises. It's perfectly OK to ask for help, but out of respect for fellow classmates and to avoid honor code violations students should phrase questions in a general / abstract fashion.

What will students learn in this course?

- This course presents the underlying statistical and algorithmic principles required to develop scalable machine learning pipelines. We present an integrated view of data processing by highlighting the various components of these pipelines, including feature extraction, supervised learning, model evaluation, and exploratory data analysis. Students will gain hands-on experience using Spark. For more details about the course material, schedule, and logistics, please read the [course syllabus](#).

How does this course compare with [CS100.1X: Introduction to Big Data with Apache Spark](#)? Is one course a prerequisite for the other one?

- This course focuses on machine learning pipelines, whereas CS100.1X provides a broader survey of topics related to Big Data and Spark. We coordinated the development of these two courses, and together they comprise the BerkeleyX Big Data [XSeries](#). Students are encouraged to take both courses, though each course can be taken on its own.
 - The virtual machine (VM) is identical for both courses, and lab 0 is identical for both courses.
 - Both courses use the same material to introduce Spark to students, and lab 1 in CS100.1X is equivalent to lab 2 in this course.
 - Students who have already worked on lab 0 or lab 1 from CS100.1X can submit their completed

notebook to receive credit for lab 0 and lab 2 in this course.

What software environment(s) are supported in this course? Why are we using these environments?

- We support a single virtual machine environment for the following reasons: (i) Having all students work in the same environment drastically improves our ability to support students; (ii) Aside from setup and submission, students can work offline on the labs; and (iii) Students can use the VM even after the course ends to continue working on the course labs and other Spark projects. The course VM provides Spark 1.3.1 running in Jupyter notebooks via IPython 3.1.0 and Python 2.7.6. Spark is deployed in standalone mode. Under the hood the VM is running Ubuntu 14.04.2 LTS 32-bit. Databricks is also providing Databricks Cloud access to a fraction of randomly selected students.

How should students perform the Spark coding labs?

- Students must first setup the course VM environment. After performing this setup, students should download the IPython student notebook for a given lab, which can be found in [the mooc-setup GitHub repo](#). Note that this repo also contains the labs for CS100.1X; the labs for this course are lab0_student.ipynb as well as all labs starting with the "ML_" prefix. The student notebooks contain several "<FILL IN>" statements that students should replace with their own code. Students can track their progress by running the tests that are included in the IPython notebook.

How should students submit their coding labs to be graded?

- After completing the lab, students must export their

notebooks as Python files, and submit the resulting .py file to the course autograder. The autograder automatically runs submitted notebooks and grades them using the *same tests* that are included in the student notebook. Hence, students are strongly encouraged to get their notebooks working locally before submitting to the autograder (the autograder will return errors for all segments of the submission that have not been completed). If you run into any problems, test locally again by restarting your IPython kernel and running all your cells from scratch.



[About](#) [Blog](#) [News](#) [FAQs](#) [Contact](#) [Jobs](#) [Donate](#) [Sitemap](#)

[Terms of Service & Honor Code](#) [Privacy Policy](#) [Accessibility Policy](#)

© edX Inc. All rights reserved except where noted. EdX, Open edX and the edX and Open EdX logos are registered trademarks or trademarks of edX Inc.

POWERED BY
OPENedX

