

Syllabus, Fall, 2019

COMP 453-001 – 2956 (in-person)

COMP 453-002 – 6917 (online)

Title: Database Programming

Time/Place: Tuesday 4:15 – 6:45, Cuneo Hall, Room 202

Instructor: Channah F. Naiman

Email: cnaiman@luc.edu

Office: Doyle Center, 205

Office Hours: TBA, but hopefully shortly before and after class

TA: Akanksha Negi anegi@luc.edu

Orientation

Description

This section of COMP 453 will use Python to access a MySQL database both locally and remotely. The Flask web development framework is used. We will incorporate both standard SQL queries as well as SQLAlchemy as an object-relational mapping (ORM) tool. A complete website with user account support and CRUD capabilities is developed. Python will also be used to introduce MongoDB for data cleaning and querying, using the MongoDB Aggregation Framework as well as the MongoDB query language. We will use Jupyter Notebooks for interactive testing, MongoDB Atlas as a cloud-based host, and Compass as a local GUI.

Table of Contents

- [Prerequisites](#)
- [Textbooks](#)
- [Course Outcomes and Learning Objectives](#)
- [Software](#)
- [Academic Honesty](#)
- [Lateness Policy](#)
- [Student with Disabilities](#)
- [Course Components and Grading](#)
- [Course Schedule](#)

Prerequisites

COMP 271 or COMP 251 (or equivalent)

Textbooks (and other materials used in place of a text)

No single textbook covers all of the topics for the course. There are many resources available for the different components of the course:

- *Database Design and SQL*: Several of the lectures and assignments in the first few weeks will draw upon *Fundamentals of Database Systems*, by Elmasri and Navathe, 7th edition. The 7th edition is the most recent. I bought a pdf version for \$3 on ecrater.com, but I understand that it is now easily found online for free. The fundamentals, which are relational database design, do not change. I will also provide detailed labs for SQL. The text information for the print version is: ISBN-13: 978-0133970777 ISBN-10: 0133970779. However, it is strongly recommended that you purchase an inexpensive e-book online, as you will use it for a few chapters and

for reference. All text lectures are also supported by my videos, which also incorporate examples and supplementary materials that are not from the text.

- *Python*: There are too many resources on Python to list here. But here are a few:
 - For our “text” for this segment of the course, we are using a [Python web development video series](#) by Corey Schaffer. I have created my own videos emphasizing and explaining the features that we need for this course. In addition, I have created our own labs (with videos) which further explore the database-related features (and other cool stuff!)
 - An interactive web version of [How to Think Like a Computer Scientist](#) starts with pretty much the same basics as the [Hands-on Python Tutorial](#), but in a somewhat different order with a different graphics library, and goes much further, and allows you to program totally inside the browser, and has embedded videos. There is also an older, somewhat different, non-interactive pdf version.
 - <http://python.org> Home site for the Python language, where you can find out more and download the latest production version starting with a 3 (3.6.1 in March 2017). Caution: an *incompatible 2.7* version may be listed first in the downloads.
 - <http://www.pythontutor.com/> A wonderful site where you run code in your browser step by step and save and email a URL reference to any step.
 - [Extra Material on IDLE errors messages](#)
 - An online interactive presentation with online exercises, covering pretty much what I do, though they do functions much later: <http://cscircles.cemc.uwaterloo.ca/> They incorporate the pythontutor.com functionality referenced above. The page has a link to a large collection of videos, Python from Scratch.
 - <https://groklearning.com/course/intro-python-1/>: Gentle Python intro, also totally in the browser.
 - http://en.wikibooks.org/wiki/Non-Programmer's_Tutorial_for_Python_3 Another introduction to Python basics in different words. No graphics, few major examples, lots of basic syntax.
 - [Official Python 3 Tutorial](#) *moves very fast for someone with no programming background*. It covers a lot, going well past our course!
 - <http://codecademy.com/step-by-step>, but with *Python 2, not 3*.
 - [Software Carpentry - Python](#) is an intro for data scientists using the numpy library and the nice development/display environment ipython.
 - A long further list at <http://wiki.python.org/moin/BeginnersGuide/Programmers>
 - A very nice, surprisingly short book that covers many topics concisely: Mike McGrath, [Python in Easy Steps](#).
 - You may want to experiment with [PythonAnywhere](#), an online virtual machine environment!
- For Mongo, we are using some of the MongoDB website videos, as well as some publicly-available websites and example files.

Course Outcomes and Learning Objectives

Outcomes: Students will learn to design and implement a transactional relational database; to develop a web application to access and manipulate the database; and to query a NoSQL database. Students will understand the fundamental concepts of database design and implementation using both SQL and NoSQL

Learning Objectives:

- Learn fundamental principles of the relational model, normalization and functional dependencies
- Apply data modeling concepts and techniques to capture the conceptual model of an application
- Create and query a relational database, using SQL
- Implement a web database application (in this case, in Python), using a web development framework (in this case Flask) and an object-relational mapper (in this case flask-sqlAlchemy).
- Query a remote and a local NoSQL database.

Software (instructions and links for all software is detailed in the [Orientation](#))

- MySQL/MariaDB (MariaDB is a clone of MySQL, at least for the time being), Apache web server (or some other local web server), and some way to interface with the database. The easiest way to handle this

is to install a WAMP (or MAMP, for macs) stack. I am using [XAMPP](#), and I have supplied detailed instructions on the [Orientation](#).

- Python
- Flask
- Flask-sqlAlchemy
- Anaconda
- MongoDB
- Atlas free-tier hosting
- Compass (a GUI for MongoDB)

Academic Honesty

Students are expected to have read the statement on academic integrity available http://www.luc.edu/academics/catalog/undergrad/reg_academicintegrity.shtml. This policy applies to the course. The minimum penalty for academic dishonesty is a grade of F for that assignment. Multiple instances or a single severe instance on a major exam or assignment may result in a grade of F for the course. All cases of academic dishonesty will be reported to the department office and the relevant college office where they will be placed in your school record. Academic dishonesty includes, but is not limited to, working together on assignments that are not group assignments, copying or sharing assignments or exam information with other students except in group assignments, submitting as your own information from current or former students of this course, copying information from anywhere on the web and submitting it as your own work, and submitting anything as your own work which you have not personally created for this course. If you do wish to use materials that are not your own, please check with me ahead of time and cite your source clearly. When in doubt, ask first! Be aware that I have changed the values for many for the textbook problems that are used for homework problems. For those problems that require open-ended answers, please be very careful to state the answers in your own words, not in the words of the Instructor's Manual, nor in the words of students who have previously taken this course.

Lateness Policy

Assignments are due as specified in the syllabus Course Schedule and on Sakai. Any extensions in due dates will be announced in class and sent as email announcements on Sakai. In the rare event that I allow an individual student to submit an assignment late, it will be graded as half credit. Most assignments are in-class assignments and must be completed during class time.

Online Recording Policy

In this class software may be used to record live class discussions. As a student in this class, your participation in live class discussions will be recorded. These recordings will be made available only to students enrolled in the class, to assist those who cannot attend the live session or to serve as a resource for those who would like to review content that was presented. All recordings will become unavailable to students in the class when the Sakai course is unpublished (i.e. shortly after the course ends, per the Sakai administrative schedule: <https://www.luc.edu/itrs/sakai/sakaiadministrativeschedule/>). Students who prefer to participate via audio only will be allowed to disable their video camera so only audio will be captured. Please discuss this option with your professor. The use of all video recordings will be in keeping with the University Privacy Statement shown [below](#):

Privacy Statement

Assuring privacy among faculty and students engaged in online and face-to-face instructional activities helps promote open and robust conversations and mitigates concerns that comments made within the context of the class will be shared beyond the classroom. As such, recordings of instructional activities occurring in online or face-to-face classes may be used solely for internal class purposes by the faculty member and students registered for the course, and only during the period in which the course is offered. Students will be informed of such recordings by a statement in the syllabus for the course in which they will be recorded. Instructors who wish to make subsequent use of recordings that include student activity may do so only with informed written consent of the students involved or if all student activity is removed from the recording. Recordings including student activity that have been initiated by the instructor may be retained by the instructor only for individual use.

Religious Holidays

Students with religious holiday conflicts: Please let me know within the first two weeks of class if you have a religious holiday conflict with any exam or homework due date, so that we can plan on an accommodation.

Students with Disabilities

If you have a documented disability and wish to discuss academic accommodations, please contact the Student Accessibility Center (773-508-3700 and SAC@luc.edu) as soon as possible. Students with documented disabilities who provide me with a letter from the SAC office will be fully accommodated as per the terms of the letter. Students who are allowed to take their exams in the SAC office are encouraged to do so. *Should you choose to take the exam in the classroom, I cannot guarantee that the classroom environment will be quiet enough to provide you with the environment that your disability may require. If you choose to take the exam in the classroom, you are taking that risk.*

Students with Sponsorships and Scholarships

If you require a certain grade in order to satisfy a sponsor or a scholarship requirement, please be sure to monitor your grade on Sakai. I will consider only your performance in this course in calculating grades, using the grading rubric posted in this syllabus. If you cannot achieve a minimum grade that is required by a sponsor or a scholarship, I will not change your grade to help you meet that requirement. This would be unfair to other students, and not reflective of your performance in this course. You are responsible to monitor your grade and to keep apprised of the [withdrawal dates](#) posted by the registrar.

Course Components and Grading

- *Lecture/Lab:*
 - For the in-person section: The lectures and labs for the first five weeks will be during class time, in person. Some labs may not be completed in class. They are not graded, but they provide a foundation for the homework for that week, which is graded. You will definitely want to complete them after class before working on the homework. Lectures and labs are also fully supported by videos. After the first five weeks, the rest of the course is completely “flipped”, meaning that you are responsible for watching the lecture and lab videos before coming to class.
 - For the online section: The lecture videos for the first five weeks should be completed before Tuesday 4:15 p.m. of the week listed on the syllabus for that topic. Similarly to the in-person, you should complete the labs before attempting the homework for that week.
- *In-class lab assignments:* Beginning in week 6, most class sessions will include a question-and-answer session at the beginning of class to answer any questions that you may have on the lectures and labs that you watched before coming to class. I will then introduce your in-class lab assignment, which must be completed during class time. Please come to class prepared!! It will be nearly impossible to complete the in-class lab assignment without having reviewed the lecture and lab videos before coming to class!! Given that there is no homework for this part of the course, other than this class preparation and, in the later weeks, working on your project, you are expected to watch these videos before class, and no accommodations will be made for lack of preparation. You will not be allowed to complete these assignments outside of class without my express permission, which I am unlikely to grant. Come to class with your computers!! For every assignment, you will be instructed what to submit (or, if I am checking the assignment in class, you may not have to submit anything). One of assignments is longer than the rest, and it may be assigned as a homework, or it may be used as a possible second exam, see below.
- *Exam:* As of this writing, there is one exam, scheduled on October 1. It covers the material on the first five weeks (Database Design and SQL queries). If you must miss the exam, let me know well in advance. Then if you have a good reason we can possibly make other arrangements, at my discretion. I have little sympathy for people who inform me after the fact for no good reason. Exam absences will be verified. If they can't be verified, you will not be excused.

No second try: If you have an excuse for not being prepared to take the exam, but decide to take it anyway, you don't get to change your mind after you see a poor grade. Being sick is not a way to get one more chance than everyone else. I may allow you to delay an exam due to illness, but I will not let you be reexamined due to a poor grade.

- *Possible second exam:* If I am not satisfied with the performance and/or participation on the in-class assignments for the Python-ORM module of the course, then there will be a second exam, on that material. If there is a second exam, it will be announced at least two weeks before the exam will take place.
- *Project:* There is a project, which takes the place of a final exam. You are required to build a web application that accesses a database that you will design. You are required to use the skills covered in the course (Python, SQL and ORM queries, etc.). You may use our sample programs as templates—in fact, I strongly suggest that you do). But you will have to design and implement CRUD for your own database application. Projects will be presented during the Final Exam time slot. I am not sure at this point that I will allow videos instead of an in-class presentation. The project will be discussed more in class. Project Overview ([Requirements](#)). [Presentation Guidelines](#).
- *Team Work:* For some assignments, and for the project, you will be allowed (or sometimes required) to work in a team. Full participation is required!! **Do not assume that if your team completes the assignment or a project that all team members will automatically be assigned the same grade.** For the project, I will check your GitHub site for commits, and I will gauge individual participation through team meetings with me (in class and via zoom). For assignments, I will walk around the class meeting with each team and ask questions to assess your participation. At any point, I reserve the right to disband a team for an in-class assignment and require individual completion and submission.

Grading

The course is worth 1005 points. It will be graded on the basis of 1000 points. In the event that our pace does not allow us to complete everything listed, I will base the grade out of the number of points for the completed components that were actually assigned.

Course Component	Points	Course Component	Points
<i>Orientation:</i>		DB Design and Modeling:	
Tour of the Course and Syllabus screenshot	5	Homework #1	50
		Homework #2	50
XAMPP Orientation:		Homework #3	50
Screenshot of Server Installation	10	Homework #4	50
Screenshot of SQL Test Command	10	Python-Flask-ORM:	
Screenshot of terminal window	10	In-class-Lab-02-Templates	40
Python-Flask-ORM Orientation:		In-class-Lab-03-Forms	40
Screenshot of pip install flask	5	Homework-Lab-4a-connect	40
Mongo Orientation:		Lab-4c-culminating assignment	100
Screenshot of Atlas Cluster	10	MongoDB:	
Screenshot of terminal mongo commands	5	Lab 1a Questions, Compass	40
		Lab 1b Queries and text index	40
Exam on DB Design and Modeling	150	Lab 2a Local DB, citibikes query	40
		Lab 2b Aggregation	30
Project	200	Lab 3 elemMatch	30

Course Schedule. This schedule is subject to slight changes to accommodate student progress and interest. Please also keep track of important dates (including withdrawal deadlines) on the [Academic Calendar](#).

Week	Date	Topic/Text/Links	Videos	Assignments
Module #1: Database Design and SQL				
1	8/27	<p>Intro and ER Modeling</p> <ul style="list-style-type: none"> Administrative introduction, syllabus Chapter 1: DB History, Motivation, basic terminology (lecture ppts) Intro Overview: integrity, concurrency, SQL Chapter 2: Schemas, DBMS, Architectures Supplier-Parts: motivational example (actually references topics in Chapter 15.1 - 15.4) Chapter 3: ER Diagrams. (PPTs) Pilot-Plane Type (example with many flaws) Use any modeling tool you want, even Word or PPT. BETTER: try ER Assistant (Windows only); Gliffy; something like Draw.io (I don't prefer their EER notation, but no biggie); StarUML; ErDraw (ok EER notation) Dia (Windows or mac); Lucid (You can keep renewing the free trial); ERDPlus (seems cool and very clear!); worst case, draw and scan into a pdf file. That is really a worst case. Yeah, don't do that.... You can go another direction entirely, and use SQL Workbench as an alternative to PHPMyAdmin, and also for ERD design. You're on your own with that, but if you're comfortable with it, great. Discuss Orientation test software installation on laptops. It's part of your Orientation points, which also include testing PHPMyAdmin by creating a University database, and populating with this DDL and Insert file. For more detailed instructions and links, see Orientation. 	<ul style="list-style-type: none"> 1.1 : Intro 1.2, 1.3, 1.4: Famous example Supplier/Parts 1.5, 1.6, 1.7, 1.8, 1.9: text PPTs (revised for lecture) 1.10: Pilot-PlaneType 1.11: Discussion of HW#1 	<p>Install XAMPP on your laptops, if you are using a laptop. Configure PHPMyAdmin, (possible issues), (You can use any WAMP, MAMP or LAMP stack, or you can use a VM on Guacamole if you request one. But I have supplied instructions to XAMPP, which works on Windows and Mac. Check the orientation assignments on Sakai.) Test your installation as per instructions on Sakai.</p> <p>You may also want to install SublimeText, atom, Brackets, BBEdit (or any good text editor) on the mac, or SublimeText, atom or Notepad++ on Windows</p> <p>Assign: Homework #1</p> <p>Orientation: Available on Sakai. Best to complete before the course begins.</p>
2	9/03	<p>EER Diagrams</p> <ul style="list-style-type: none"> Chapter 3: continue ER Examples: EN 3.21, 3.23, 3.32, 3.33, 3.34 Chapter 4: EER Diagrams: Examples PPTs Requirements Definition possibly begin Chapter 9 	<ul style="list-style-type: none"> 3.21, 3.23, 3.32, 3.33, 3.34 Chapter 4 PPT video goes along with the PPTs (It says Ch. 8, but ignore that) Requirements Analysis HW #2 Discussion 	<p>Due: Homework #1</p> <p>Assign: Homework #2 (use EER case) resolve any installation issues on stack</p>
3	9/10	<p>Map to Relations; SQL Lab</p> <ul style="list-style-type: none"> Chapter 9: mapping ERDs to relations (PPT Slides 12-23) revisit the Supplier-Parts example Chapter 5: relational model (formalize what you already know by now) and begin SQL (PPT slides) 	<ul style="list-style-type: none"> Ch. 9 PPT Video Ch. 5 PPT Video Lab 1 Setup and DDL Lab 1: Q1, Q2, Q3 parts of video misaligned, but still readable 	<p>Due: Orientation Tasks (due before class at 4:00 p.m. for online students too)</p> <p>Due: Homework #2</p> <p>Assign: Homework #3</p> <p>Lab #1: labs do not have to be</p>

		<p>Simple SQL Queries (based on Chapter 6, but modified for MySQL)</p> <ul style="list-style-type: none"> • Lab #1: see Company Schema using DDL and Insert to create and populate the database • useful links: copy and rename a DB in PHPMyAdmin and mysqldbcopy or mysqldump for command line • Create and populate the Pine Valley database (ERD link). • LoadBigData for Pine Valley, if instructed by video 	<ul style="list-style-type: none"> • Lab 1: Q4 – Q8 • Lab 1: Q9 – Q12 • Lab 1: Q14 – Q18 • Lab 1: Q19 – Q23 • Lab 1: Q24 – Q25 • HW #3 Instructions 	<p>submitted, as they are not graded. But you should complete them before working on the homework.</p>
4-5	9/17-9/24	<ul style="list-style-type: none"> • Chapter 14: sections 14.1-14.4, normalization and functional dependencies, brief overview PPTs • Lab #2: More complex SQL Queries (based on Chapter 7, perhaps with some extra examples) user PVFC Big Version DDL, data1 	<ul style="list-style-type: none"> • 4.1: Design Guidelines4 • 4.2: FDs and Keys • 4.3: PPT Normalization • 4.4: Normalization Example • 4.5: Semantic Reconciliation • 4.6: Equi-Natural-Joins • 4.7: Outer Joins • 4.8: Big Join • 4.9: Self Join • 4.10: Subquery • 4.11: Correlated Subquery • 4.12: Derived Tables 	<p>Due : Homework #3 Lab #2: complete before working on homework. Assign: Homework #4 Hints for HW4</p>
6	10/01	<p>Exam (Midterm Review)</p> <p>Cool link: SQL Injection</p>		<p>Due: Homework #4 Due: All Python and Flask orientation assignments are due on 10/6 at 11:55 p.m. Earlier is better!!</p>
<p>Module #2: Database Web Application, using Python, Flask, mysqlAlchemy <i>Before beginning this module, please download Python-DB-Lectures-Labs.zip. For mac students using MAMP, use this link. For students using the VM, use this link. (Actually, it is already downloaded and unzipped on the VM, for students using the VM.) It includes an entire directory structure with the files that you need for this Module. I reference the appropriate files (with its file path) for each lecture/lab listed below. Videos will be linked to individually (not as a zip file) In general, find the files listed in the Topics column, and let the videos guide you as to what to look at or do.</i></p>				
7	10/15	<p>Intro and Setup for the web application</p> <ul style="list-style-type: none"> • Reference: Setup and Code from original video (totally optional!): <ul style="list-style-type: none"> ○ Code: DemoApps/01-GettingStarted • Lecture: Review Setup and Code for getting started: <ul style="list-style-type: none"> ○ FlaskDemoLectures/GettingStarted (follow video with PPTs) You are not required to submit anything for this part, but if you follow along, this will get you up and running in Flask. • Review Lab Instructions for Setup (should have completed during Orientation. I listed this for review and reference.) <ul style="list-style-type: none"> ○ LabInstructions/1-Setup.pdf (not required to submit now) 	<ul style="list-style-type: none"> • Overview • 1-Flask-Hello-World 	<p>Python-Setup Lab (Does not have to be submitted now. It should have been completed during Orientation.)</p>

		<p>Using Templates in Flask</p> <ul style="list-style-type: none"> For reference: Code from Original Application <ul style="list-style-type: none"> Code: DemoApps/02-Templates Lecture: FlaskDemoLectures/02-Templates (follow video) In-class lab assignment: LabInstructions/Lab-1-2-Templates.pdf 	<ul style="list-style-type: none"> 02-Templates 	Python Lab 1-2: Templates
8	10/22	<p>Creating and using Forms:</p> <ul style="list-style-type: none"> For reference: Code from Original Application <ul style="list-style-type: none"> Code: DemoApps/03-Forms-Validation-Final Lecture: FlaskDemoLectures/03-Forms (follow video) <ul style="list-style-type: none"> In-class lab assignment: LabInstructions/Lab-3-Forms.pdf 	03-Forms	Python Lab 3 Forms Due: If Labs-1-2 templates was not completed in class last week, it is due now. I will confirm in class.
9	10/29	<p>Python connecting to a database.</p> <ul style="list-style-type: none"> First we cover simple connections, non-flask connections, and basic flask connections, not from our DemoApp. <ul style="list-style-type: none"> Watch the video lab-connect-db Complete Lab4-a before coming to class. <ul style="list-style-type: none"> LabInstructions/Lab-4a-db-connect.pdf, using the files in the folder LAB-connect-db. Complete the parts in red. We will review Lab-4a, and we will discuss Lab 4-b, and maybe start on it. 	<ul style="list-style-type: none"> 04a-lab-connect-db 04-DB (lecture on the DemoApp) 	Due: If Lab 3 Forms was not completed in class last week, it is due now. I will confirm in class DUE: Lab 4a-connect-db .
10	11/05	<p>There is over an hour's worth of videos to watch before coming to class!! Please come to class prepared!!!</p> <p>Crud on your database, going way beyond the DemoApp original app</p> <ul style="list-style-type: none"> For reference: Code from Original Application <ul style="list-style-type: none"> Code: DemoApps/04-Database Code: DemoApps/05-Packages Code: DemoApps/08-CRUD Lecture: FlaskDemoLectures: (follow video 04-DB) This is just a lecture on the original code of the original videos. <ul style="list-style-type: none"> Code: FlaskDemoLecture/04-DB Lecture: In the folder DemoAppLabs, we will look at ALL of the code in that folder. Each app is a slight variation or improvement over the previous apps <ul style="list-style-type: none"> In-class lab (discussion, no submission): LabInstructions/Lab-4b-db-connect-CrUD.pdf. The lab instructions tell you exactly when to look at which project's code. Begin Lab 4-c, time permitting 	<ul style="list-style-type: none"> 04-DB (lecture on the DemoApp) 04-DB-trySQL 05-Packages (lecture) 05-Packages-try-ORM-model-reflect 06-bcrypt (brief explanations, just FYI) 08-CrUD (lecture, not lab. Just watch.) 08-CrUD-Choices 08-CrUD-Update-Delete-New 	
11	11/12	<ul style="list-style-type: none"> Catch-up, continue on Lab 4-b, discuss Project Culminating assignment on Module #2: Lab 4-c. Lab Instructions/Lab-4c-big-lab Project Overview (Requirements). Presentation Guidelines. 		Lab-4-b we will discuss together in class. (Online section: there are videos to support this discussion.) This is not an assignment. No submission. Yet, you still have to do it. Wow. Lab-4-c due 11/11 at 11:55 p.m.

				Due: Mongo Orientation Labs, due on 11/16, 11:55 p.m.
<p>Module #3: Querying MongoDB</p> <p>Before beginning this module, please download MongoDB-Lectures-Labs.zip. I reference the appropriate files (with its file path) for each lecture/lab listed below. Class time will be used to work on projects, with some Q & A on Mongo. The MongoDB lectures and labs are fully supported by videos, and you are expected to work on them independently and come to class with any questions.</p> <p>Online teams: You are required to set up zoom meetings with me. A signup form will be posted as we get closer to this module.</p> <p>Videos will be linked to individually (not as a zip file). In general, find the files listed in the Topics column, and let the videos guide you as to what to look at or do. Also download the jupyter notebooks that you will using for this part of the course. Place that folder in your MongoDB application folder for this course (my videos show mine as being Documents →mongodb-analytics, but you may have yours stored elsewhere).</p>				
12	11/19	<p>Setup and Basic Querying: Lecture (watch the four lecture videos before class!!:</p> <ul style="list-style-type: none"> • Make sure that you have completed the Mongo Orientation • Lecture: Follow the directions in the four lecture videos. It is strongly recommended that you duplicate the work shown in the videos • Project Check (Online teams—zoom meetings.) 	<ul style="list-style-type: none"> 1-MDB-Atlas-Cluster 2-MDB-Import-Movies initial 3-Install Compass 4-Connect-to-Atlas 	Due: Mongo Orientation Labs, due on 11/16, 11:55 p.m.
		<ul style="list-style-type: none"> • Lab: Complete MongoDB-Lectures-Labs/MDB-Lab-1a.pdf <ul style="list-style-type: none"> ○ Complete the questions answered in red, and submit • Lab: Complete MongoDB-Lectures-Labs/MDB-Lab-1b.pdf <ul style="list-style-type: none"> ○ Complete the questions answered in red, and submit the answers, and the screenshots as specified in the videos. • Project Check (Online teams—zoom meetings.) 	<ul style="list-style-type: none"> • Lab-1-b-First Query • Lab-1-b-create-text-index 	Due: Mongo Lab-1a , Lab-1-b
13	11/26	<p>Labs: MongoDB-Lecture-Labs/Lab-2-a-b-c-Student. Submit the required screenshots from Lab2a and Lab 2b.</p> <p>Hopefully, we'll have some time for a project check. If not, we'll have team zoom meetings.</p>	<ul style="list-style-type: none"> • 5-Aggregation Framework • 6-Filtering • 7-local-trips • 8-Project-reshape • 9-UpdatingDocuments • 10-Data-Types 	Lab-2a-2b There is no assignment submission for Lab 2c. Yet, you will need it. You get to do it for no points!
14	12/03	<p>Lab 3: Open the notebook as specified in the lab instructions: LabInstructions/Lab-3-student</p> <p>Complete the missing cells in the notebook Submit the screenshots of the completed cells</p>	<ul style="list-style-type: none"> • 11-FilteringOnArrayFields • 12-elemMatch 	Lab-3
		No lab for these topics	<ul style="list-style-type: none"> • 13-GeoSpatial • 14-Graphics 	
15	12/10	Project Presentations	To be discussed in class	