DATA 503

Fundamentals of Data Engineering

Wed, Room 2, 200 Market Building
Fall 2022

Jed Rembold, PhD
jjrembold@willamette.edu
http://willamette.edu/~jjrembold/classes/data503
Collins 311
Office Hours: HW 2:30-4, TTh 2-4, or catch me anytime online!
Office Phone: (503) 370-6860

This syllabus is subject to change or adaptation as the semester progresses.

Course Description: As "big data" more and more becomes a common facet of everyday life, the bulk of attention has been focused on the analysis and usage of this information. Such a focus ignores the vital fact that, no matter how much data is gathered, little analysis is possible unless that data has been stored and organized in such a way as to be easily accessible. And as more things depend on huge repositories of data, so too does the importance of data integrity and scalability. This course focuses on the basic skills of a data engineer tasked with acquiring, storing, and maintaining such repositories of information. To this end the course is broken roughly into two parts.

The first deals with the theory behind planning, storing, and organizing large amounts of data in a way that is both efficient, reliable, and scalable. We will see that there are various large-scale approaches to how this can be done, but relational databases are one of the industry standards in storing and organizing information, and thus the other half of this class will revolve around learning how to create, manipulate, query, and maintain such relational databases using SQL. In particular, this class will focus on the open-source Postgresql variant of SQL, but the majority of learned techniques will be readily applicable to any other SQL variant. Students will leave the course having a broad theoretical foundation about the questions and trade-offs that underpin modern data storage, as well as feeling comfortable creating and utilizing a relational database to both store and query information.

Prerequisite(s): None
Note: A minimum grade of C- is required for this course to count toward university credit.
Credits: 4.0

The split nature of this class lends itself to multiple textbooks, but if you only get one, it should be the SQL book.

Text: *Practical SQL: A Beginner's Guide to Storytelling with Data* (1st or 2nd edition)
Author: Anthony DeBarros
ISBN-13: 9781593278274
Comments: This is the main book I'd suggest acquiring, especially if you haven't done much with SQL in the past. It is fairly cheap, and the older first edition will still work fine.

Optional Text 1: *Designing Data-Intensive Applications: The Big Ideas Behind Reliable, Scalable, and Maintainable Systems* (1st edition)
Author: Martin Kleppmann
ISBN-13: 978-1449373320
Comments: This is the text I used to teach the second part of the course last year. It is an excellent resource, and really delves into the minutia of different storage systems. It is not perhaps the most applicable if you don't end up needing to do a ton of data architecture, and hence why I've pivoted away from it this semester.

Optional Text 2: *Fundamentals of Data Engineering* (1st edition)
Authors: Joe Reis and Matt Housley
ISBN-13: 9781098108304
Comments: This book *just* came out at the end of July, and thus I haven't gotten a full chance to delve deep into it. By all metrics though, it seems like exactly the type of text that would nicely accompany this class, and I've loosely scheduled portions of the class around its chapters. If you are hoping to make data engineering your career, it would probably be a solid pickup.

Course Objectives:

Over the semester, students will gain working knowledge in:

1. The basic tasks of a data engineer
2. The role a data engineer plays amidst organizational and analysis pipelines
3. The fundamentals of working with and querying a relational database using SQL
4. The key factors that drive database design choices
5. More advanced database queries involving text mining or spatial relationships
6. Alternatives to relational databases and future/current developments in the field

Grade Weighting:

Homework	40%
Project	20%
Midterm	17%
Final Exam	23%

Letter Grade Distribution:

>= 92.00	A	72.00 - 77.99	C
90.00 - 91.99	A-	70.00 - 71.99	C-
88.00 - 89.99	B+	68.00 - 69.99	D+
82.00 - 87.99	B	62.00 - 67.99	D
80.00 - 81.99	B-	60.00 - 61.99	D-
78.00 - 79.99	C+	<= 59.99	F

Student Learning Objectives (SLO):

Upon completion of the course, students should be able to:

- Describe what a relational database is and what advantages or disadvantages they have over other forms of storing data.
- Design and implement a relational database, including creating multiple tables, parsing and inserting data into those tables, and including relationships between table columns.
- Query data from a database, including using advanced filters, descriptive statistics, and joins to combine information from multiple tables.
- Use SQL for analyzing more complicated types of information, including parsing text using regular expressions and analyzing spatial geometric information.
- Describe the role that a data engineer fits within a data analysis team and within a larger organization.

Course Assessment:

- **Homework**
  - There will be weekly homework sets which will be due Thursday nights at 11:59pm. Homework will involve a mix of more theoretical problems as well as SQL based problems. Solutions to theory questions should either be typed and saved as PDF or neatly (*and legibly*!) handwritten and photographed or scanned to PDF. Both PDFs and any SQL code written as part of a problem will be submitted through GitHub Classroom before the deadline. Assignments will be posted on the class webpage each week and the provided link at the top of the assignment should be followed to accept the assignment and to download any possible extra materials for that week's assignment.
- **Project**
  - There will be a project that will be partly ongoing throughout much of the semester, but will culminate in a final paper. The project will give students a chance to interact and explore the full pipeline of modern data engineering. Deliverables from the project will involve a paper describing what was done, why certain design choices were made, how the database was organized, and what basic analysis was done to answer an interesting question.

- **Tests**
  - There will be 2 tests this semester: a midterm in the center and a final at the end. Both will involve a mix of theory, writing correct SQL to achieve a goal, or interpreting what a piece of SQL is doing. I have an old midterm I can share, and there will be study guides available before each exam, so students can feel prepared for the types of questions that may appear.

Course Policies:

Late Work Policy

I understand that sometimes things come up where you are unable to get an assignment in on time, and I strive to be incredibly flexible and accepting of late work. However, there also comes a point when you get too far behind to realistically keep up with the class. In an effort to compromise between the two, my late policy allots you 3 cumulative days (72 hours) of unpenalized late work throughout the entire semester. So you can turn 3 assignments in one day late, 6 assignments in 12 hours late, etc. without penalty. Once you have used up your 3 days (72 hours), assignments will drop in worth by 20% per day. If you are approaching the point where an assignment is heavily penalized, consider just turning in what you have, so that you can move on and keep up with the class. In the case of extenuating circumstances, please just come talk to me. We'll figure out what can be done.

Incomplete Policy

An incomplete grade will only be granted in the case of prolonged illness or family emergencies that remove the student from the learning environment for an extended time period during the semester. Under no situations will an incomplete be granted due to a student falling behind through lack of motivation, understanding, or time management skills. If you are concerned about your progress and how you are doing in the class, please come visit me! We can sort out where you are struggling and work out a plan to get you back on track.

Willamette Policies:

Academic Honesty

Cheating is defined as any form of intellectual dishonesty or misrepresentation of one's knowledge. Plagiarism, a form of cheating, consists of intentionally or unintentionally representing someone else's work as one's own. Integrity is of prime importance in a college setting, and thus cheating, plagiarism, theft, or assisting another to perform any of the previously listed acts is strictly prohibited. I may impose penalties for plagiarism or cheating ranging from a grade reduction on an assignment or exam to failing the course. I can also involve the Office of the Dean for further action. For further information, visit: http://www.willamette.edu/cla/catalog/resources/policies/plagiarism_cheating.php.

Time Commitments

Willamette's Credit Hour Policy holds that for every hour of class time there is an expectation of 2-3 hours work outside of class. Thus, for a class meeting three hours a week, you should anticipate spending 6-9 hours outside of class engaged in course-related activities. Examples include study time, reading and homework, assignments, research projects, and group work.

Diversity and Disability

Willamette University values diversity and inclusion; we are committed to a climate of mutual respect and full participation. Our goal is to create learning environments that are usable, equitable, inclusive and welcoming. If there are aspects of the instruction or design of this course that result in barriers to your inclusion or accurate assessment or achievement, please notify me as soon as possible. Students with disabilities are also encouraged to contact the Accessible Education Services office in Smullin 155 at 503-370-6737 or accessible-info@willamette.edu to discuss a range of options to removing barriers in the course, including accommodations.

Tentative Course Outline:

The weekly coverage will almost certainly change as it depends on the progress of the class. However, this should serve as a rough guide.

Week	Date	Chapter	Description	Due
1	Wed, Aug 31	FoDE: Ch 1 SQL: Ch 1-2	Data Engineering Described Tables and SELECT	
2	Wed, Sep 07	FoDE: Ch 2 SQL: Ch 3-4	The Data Engineering Lifecycle Data Types and I/O	
	Thu, Sep 08			HW 1
3	Wed, Sep 14	FoDE: Ch 3 SQL: Ch 5	Designing Good Data Architecture Math and Stats with SQL	
	Thu, Sep 15			HW 2
4	Wed, Sep 21	FoDE: Ch 4 SQL: Ch 5-6	Choosing Technologies Math and Joining Tables	
	Thu, Sep 22			HW 3
5	Wed, Sep 28	FoDE: Ch 5 SQL: Ch 6	Data Generation Joining Tables	
	Thu, Sep 29			HW 4
6	Wed, Oct 05	FoDE: Ch 6 SQL: Ch 7-8	Data Storage Designing and Grouping	
	Fri, Oct 07			HW 5
7	Wed, Oct 12		**Midterm**	
8	Wed, Oct 19	FoDE: Ch 7 SQL: Ch 9	Data Ingestion Inspecting and Modifying Data	
	Thu, Oct 20			HW 6
9	Wed, Oct 26	FoDE: Ch 8 SQL: Ch 11-12	Modeling and Transforming Datetimes and Advanced Queries	
	Thu, Oct 27			HW 7
10	Wed, Nov 02	FoDE: Ch 9 SQL: Ch 13	Serving Data Mining Text	
	Thu, Nov 03			HW 8
11	Wed, Nov 09	FoDE: Ch 10 SQL: Ch 14	Security and Privacy Spatial Data with POSTGIS	
	Thu, Nov 10			HW 9
12	Wed, Nov 16	FoDE: Ch 11 SQL: Ch 15	The Future Views, Functions, and Triggers	
	Thu, Nov 17			HW 10
13	Wed, Nov 23		*Fall Break*	
14	Wed, Nov 30		**Project Presentations**	
15	Wed, Dec 07		**Final**	