Some links in this tab are only accessible to registered students.
[Download the Syllabus as a PDF](#)

About

In this class you will have the opportunity to learn the basic skills needed to

1. design and implement a simple to moderately complex database,
2. make good decisions regarding database design, and
3. document your design and queries in standardized notation.

Additionally, you will have the opportunity to practice these skills by building a simple website driven by a database backend.

Recommended books

Relational Database Design and Implementation by Jan L Harrington, 2016 Edition is the recommended textbook for CS340-400.
A rough mapping of the current weeks to the chapters of the book is given below

- Chapters 1,3 -- Week 1
- Chapters 4-5 -- Weeks 2-3
- Chapter 10,16-17, 19 -- Week 4
- Chapter 11 -- Week 5
- Chapters 7, 13-15 -- Week 6
- Chapter 6 -- Week 7

This book is available for free *online* for all the OSU students at the [OSU Library](#)

If you decide to use Python for web development in this course, the [Flask user guide](#) should be good enough.

If you decide to use node.js for web development in this course, recommended books from CS 290 should suffice.

Course Content

This course is dedicated to learning the basics of database design and use.

To accomplish this, along with 4 assignments you will work in groups on a term Project implementing the concepts that you learn each week.

There will also be small quizzes which will help you assess your own understanding of the material. With few exceptions, if you ever get a wrong answer on a quiz that means you are not understanding critical information and

it is your responsibility to get clarification. I hope you will not hesitate to ask a question on Piazza if this happens!

All the content, assignments, quizzes and project steps will be available in weekly Modules.

Expectations from Student

Prior Knowledge

Students are expected to know the following:

You should be familiar with good coding practices. Good coding style is required and not taught in this class. You should understand basic control structures. If you are unable to code a simple sorting algorithm, you will have trouble in this class as this level of coding experience is expected.

You should also have completed 290 or be a strong student currently taking the course. It is possible to take these two courses at the same time, but if you run into difficulty in 290 you can have some real trouble late in this course, so do that with some caution.

Code quality

Code must be clear and you must understand what it is doing. Having well-documented code is going to be extremely important. I or the TA may not know the platform you are using, so it is your responsibility to make sure that your work is clear enough so that we can follow what is happening.

You should also have no major errors in your program. If we can manage to get the program to throw some default error message that is usually a major issue. Errors which you handle via a clear message to the user (e.g. 'Please enter only numbers in the age field') are usually fine. On the other hand, error messages like 'Error 0x00001: Null pointer to Null found, expected pointer to Int Factory Factory' are not.

When possible, you should find a style guide and conform to it.

About the Instructor

The Instructor Introduction video is on the [Start Here - Overview](#) page.

SCHEDULE CONTENT

- Week 0 -- Syllabus & policies (This is one week before the start of the term)
- Week 1 -- Introduction to Databases & Tools Setup
- Week 2 -- ERD & Schemas
- Week 3 -- Models, Diagrams & Schemas
- Week 4 -- SQL - DML
- Week 5 -- SQL - Advanced DML, DDL
- Week 6 -- ER to DDL
- Week 7 -- Relational Algebra
- Week 8 -- Web application development
- Week 9 -- ORMs, Diagramming, Procedures & Triggers
- Week 10 -- Final Project submission

For a detailed schedule of things you are required to turn in, refer the table below.

You can take the Syllabus Quiz, once you are registered for the course.

Grade letter	Percentage floor
A	92
A-	90
B+	87
B	82
B-	80
C+	78
C	72
C-	70
D+	68
D	62
D-	60
F	0

Grading Scale

The final grade will be computed using a weighted average according to the table above and the weighting below.

Note about Partial Credit

In general assignments are not graded such that each piece is worth a fixed portion of the total credit for the assignment. For example, if 40% of the program is non-functional that generally indicates that there is some important concept that has been missed. If that is the case, then the grade will be a 0. You can then resubmit it to earn up to 70% of the assignment credit, but the revised submission must be entirely functional.

Maximizing Credit

You should treat me as a client. I will give you a set of requirements. Requirements can be interpreted differently; this is unavoidable. You have to meet my interpretation of the requirements to get an A. If you have any doubt about what a requirement means, you should ask me to clarify.

Even if you do not plan to do work on the assignment till near the deadline, you should read the requirements as soon as you can to see if anything is ambiguous so you can get clarification while there is still time to act on it.

Grade Weighting

- Quizzes - 15%
- Assignments - 25%
- Participation - 20%
- Project - 40%

Questions about grades

The TAs will grade all your submissions, unless otherwise noted. If you have any concerns, contact them **via email** as noted in the Where to go for help?. Any concerns about grades should be communicated within 7 days of receiving the grade.

Accommodations

"Accommodations are collaborative efforts between students, faculty and Disability Access Services (DAS). Students with accommodations approved through DAS are responsible for contacting the faculty member in charge of the course prior to or during the first week of the term to discuss accommodations. Students who believe they are eligible for accommodations but who have not yet obtained approval through DAS should contact DAS immediately at 541-737-4098."

Students with documented disabilities who may need accommodations, who have any emergency medical information the instructor should be aware of, or who need special arrangements in the event of evacuation, should make an appointment with the instructor as early as possible, and no later than the first week of the term. Class materials will be made available in an accessible format upon request.

Late Policy

Requests for extensions are considered on a case by case basis. Non-emergency requests must be submitted via email at least 60 hours before the due date. (Not having enough time to get the assignment done does not, by itself constitute an emergency, sorry!) If you don't know if you will need an extension but might, you should ask for one. If no extension is requested the below scores will be used for regular homework assignments. **This policy does not apply to exams, final projects, how-to guides, peer reviews, group reviews or Project Steps(Draft and Final versions). Those are not accepted past the due date without an emergency or prior agreement.** In general to get extensions on these items you need to prove unavoidable extenuating circumstances that arose after the add/drop period closed.

Emergency requests *must include* some form of documentation from a reliable 3rd party (doctors, news stories, admins of internet providers etc.).

You will lose 10% of your grade for each day past the due date.
For example, if an assignment worth 100 points was due on Sunday midnight and you turn it in on Wednesday 0100 hrs, you will lose 20 points according to the late policy.

However, no submission will be accepted 7 days after the due date in this quarter.

Communication Methods

Piazza
> This should be used for *all* questions seeking technical help, conceptual help or assignment clarification. Essentially if it is any sort of information that may benefit other students, it should be posted to Piazza. If there is a question that gets ignored for longer than 48 weekday hours you can email the instructor and TA and ask them to follow up with it. It may have just gotten missed. Look below for the Code Sharing policy.

Slack
> The Slack channel CS340-ecampus is the primary mode of conducting office hours by the TAs unless

specified. At the beginning and end of each office hour session (whether or not it is held using Slack), the TA would put up a message on the Slack channel informing so. If you are sharing code on Slack, look below at the Code Sharing policy.

Email

Email should be used for anything that contains sensitive information. So if you have a question about a grade or want to request an extension, do so via email. All emails should have the exact characters (yes, include the parantheses) `[CS340]` at the start of the subject so that they get priority in my inbox. In addition, every time you reply to an email thread it knocks it to the back of the queue because email gets processed in order of the most recent emails last. So be careful sending multiple replies as it might bump you back in the queue.

Canvas Comments

Canvas comments should *only* be used by students to add commentary *prior* to grading and by myself and the TAs to give feedback. If you need to communicate something to myself or the TAs after your assignment has been graded do so via Email. We *will not* see comments posted to your assignment submission after it has been graded.

Canvas Mail

The TAs and myself try the best we can to monitor Canvas email. But there are less options to sort and filter mail via that system so there is a much higher likely-hood that we will miss things that get sent via Canvas mail. So I suggest you do *not* use it.

TA Office Hours

The primary mode of office hours by the TAs, unless otherwise indicated, will be the CS340-ecampus Slack channel. At the beginning and end of each office hour, the TAs will inform on the above Slack channel.

Times are in PDT. The time slots below will be updated in Week 2 based on the Beginning of the Term survey.

Monday	Name - Hours	
Tuesday		
Wednesday		
Thursday		
Friday		

Contacting the TAs

- **Brijesh Bhuva -** bhuvab@oregonstate.edu
- **Chao Ma -** machao@oregonstate.edu
- **Wang Jing -** wangji9@oregonstate.edu
- **Fengfei Zheng -** zhengfen@oregonstate.edu

Contacting the Instructor

Office hours with the Instructor are available by appointment.

Email at hedaoos@oregonstate.edu, FOUR different time slots for the next 72 hours and I can almost always accommodate one of them.

All technical questions, as well as questions about assignments, quizzes, or anything that you learn in class (unless they are related to your grade), should be posted in Piazza so that all students can benefit from them.

Communication Timelines

You can expect a response to emails within 48 weekday hours. So if you send an email at 8am on Monday, you should have a response by 8am on Wednesday. If you send an email at 10pm on Thursday, you should have a response by 10pm on Monday.

Do not expect communication on the weekends, though I do my best to look for urgent questions that are preventing you from making progress and answering them, even on the weekend. The more specific the question, the more likely you are to get help.

Code Sharing

You will not get in trouble for sharing code on Piazza in order to solve problems. The communication guide actually mandates that you share portions of your code if you want to ask a good question. If you are worried that you are posting too much code, mark it private and ask me to review it. Note that this is a more permissive policy than the standard policy for the program.

You will get in a great deal of trouble if you copy code without citing it. See the policy on plagiarism. Code from lectures is not your own, code from StackOverflow is not your own, code from the node.js documentation is not your own. If it is not your code you must cite it. If you cite it, you must provide documentation in very great detail of what it is doing so that I know you understand the code you are using.

Code editor

Amount of use: Extensive. Used for the Project and some SQL assignments.

Expected knowledge: Proficient

Coverage in class: None. You should be able to use a text based code editor well.

Examples: Atom, Notepad++, Sublime Text, Emacs, Vim or TextWrangler.

Diagramming tool

Amount of use: Extensive. Used for drawing all ER diagrams for assignments.

Expected knowledge: Moderate

Coverage in class: Minimal. You should be able to figure out how to draw diagrams using the various shapes and toolbars available in the software.

Examples: Microsoft Visio for Windows which is available for free for OSU students and Dia, LibreOffice Draw which can be used on Mac/Windows/Linux and are free and open source.

FTP Program

Amount of use: Moderate. Used for the Project.

Expected knowledge: Working knowledge

Coverage in class: None. You should be able to upload files to your engr directory already. You should also know how to set permissions on those files.

Examples: MobaXTerm or FireFTP addon for Mozilla Firefox

Browser-Based Debugging Tools

Amount of use: Moderate. Used for the Project.

Expected knowledge: Limited

Coverage in class: Minimal. This is expected from 290.

SSH use.

Amount of use: Moderate. Used for the Project

Expected knowledge: Working knowledge

Coverage in class: Very limited. Expect that you know all of the basic Linux commands to navigate and interact with a file system.