Csci 4131 Internet Programming

Lecture 17, March 20th
Spring 2024

Instructor: Dr. Dan Challou

Logistics (Csci 4131, Lecture 17, March 20th)

- Homework 4 due this Friday 3/22
- Zybooks HW 8 due Sunday 3/24 (***topics are key to doing HW 5 and 6 successfully !!!***)
- **Exam 2 next Wednesday 3/27** – emphasis on topics covered since the last exam in Week 5
- Homework 5 will be out this week

Readings/Tutorials: Node.js, JSON, Fetch, AJAX – For HW 5!

Node.js References and Tutorials:
Your zyBook
https://www.w3schools.com/nodejs/
https://codeburst.io/the-only-nodejs-introduction-youll-ever-need-d969a47ef219
Video intro: https://www.youtube.com/watch?v=TlB_eWDSMt4

JSON References / Tutorials:
Your zyBook
https://www.w3schools.com/js/js_json_intro.asp
https://www.w3schools.com/js/js_json.asp
www.json.org
Optional: Chapter 10.3.3 Sebesta

FETCH References / Tutorials:
Your Zybook
https://www.w3schools.com/js/js_api_fetch.asp
https://javascript.info/fetch

AJAX References / Tutorials:
Your Zybook
https://www.w3schools.com/xml/ajax_intro.asp
Optional: Sebesta, Chapter 10

Questions ?

Technologies needed for HW 5

- All the stuff from the first ½ of the course plus:
- Node.js (You will use it to construct a HTTP server)
- JSON
- FETCH

Agenda

- Last Time
  - Final HW4 items
  - Json (JSON) wrapped up
  - AJAX and Fetch
- Today
  - AJAX, Fetch, and Node.js revisited
  - Exam 2 review
  - Introduction to RDBMS and SQL

Last Time

- AJAX and Fetch: Purpose/Use, Examples, Review, Comparison

- AJAX vs Fetch: Comparison
  - JSONreq.html
  - FetchJsonLat.html
  - locations.txt

Lecture 17, Exercise 1 Think/Pair – and then submit your answer to the Lecture 17 Exercise submission link in the week 9 module

- Consider Fetch and Ajax
- (read through the article at the following link: https://medium.com/@reemshakes/is-ajax-getting-replaced-by-fetch-api-55207234793f)
    - Is one more efficient than the other?
    - If so, how? If not, why not?

Working Exercise – **Every Person has to do this**

(Lecture 17 Exercise 2)/ Update Server

If you have node.js installed on your computer:

1) Open a browser, log into Canvas, Download the file StudentFileServerAF.zip from the week 9 module on Canvas

2) Create a folder named L17_node

3) Copy StudentFileServerAF.zip into the Folder

4) Extract the contents of the file into the folder named L17_node

Otherwise Log into Vole or a CSELabs Machine via another utility (e.g. putty), SSH, ???

And repeat steps 1 – 4 above

PHASE 2

List the contents of the Folder L17_node

You should see the files below in the listing:

FetchJsonLat.html
FetchJsonTut.html
JSONrequex.html
myTutorials.txt
locations.txt
index.html
StudentFileServerAF.js

Lets have a look at

- The node.js server in the file:
  - **StudentFileServerAF.js**,

PHASE 3: UPDATE and RUN the Node.js Server

1) Edit the file **StudentFileServerAF.js**, and **change the port number to (8 or 9) + last 3 digits of your x.500 id**
   (mine is chal0006, thus the port number 8006)
   save your changes

2) In a terminal, Run your server: node StudentFileServerAF.js

3) Open a browser (Chrome or FireFox). In the address (url bar) type;
   http://localhost:port_number_from_step_1_above/FetchJsonLat.html
   You should see the latitudes and longitudes from the file locations.txt in your browser window.

4) Kill your server (Cntl-C)

5) Finally, add the code necessary to the enable the server (**StudentFileServerAF.js**) to return the files:
   FetchJsonTut.html, and
   myTutorials.txt

(see the comments in the file StudentFileServerAF.js for locations where you have to add code)

PHASE 4: The Exercise (Lecture 17, Exercise 2):

i) Test out your changes by running your server

```
node    StudentFileServerAF.js
```

ii) Open a browser (Chrome or FireFox). In the address (url bar) type:

http://localhost:port_number_from_step_1_above/FetchJsonTut.html

You should see the a list of links to tutorials on W3 Schools myTutorials.txt in your browser window

Submit the file: **StudentFileServerAF.js** to the Lecture 17, exercise 2 link when you are done!!!!

Exam 2 Logistics, Scope, Format

- Exam 2 will be held in this classroom, during class, next Wednesday 3/27 from 11:15 – 12:30pm, here (in this classroom)
- Exam will be on paper.
- Allowed: One Computer, anything on paper (books, notes, etc)
- Not allowed – internet (turn off wifi and Bluetooth connectivity) , phone, etc

Format

- True / False
- Mutiple Choice
- Short Answer
- Write short code / method /functions

Scope

- HW 3
- HW 4
- Lecture 10 though lecture 18 (this coming Monday)
- Zybooks – HW 5,6,7, 8
- In-class Lecture Exercises

Key Topics

- JavaScript
  - Animation
  - Closures
  - Race Conditions
  - JSON
  - Ajax / Fetch
  - Interaction with DOM
  - Events, Event handling

HTTP Protocol:

- URLs
- HTTP Messages (GET, POST, Variants on GET)
- HTTP Errors

Node.js

Basic RDBMS and SQL Topics (covered in zybook)

Likely used/required in the exam, but not the focus of the exam: HTML, CSS

Finally

- If something happens that keeps you from taking the exam
- Notify us immediately (csci4131s1s24-help@umn.edu)
- Get Documentation in support of the University Sanctioned Reason you have for missing the exam. Only University Sanctioned Reasons will be accepted, otherwise a grade of zero will be assigned for the exam.

Introduction to RDBMS and SQL

- Zybook Chapter 11, sections 6 and 7
- Zybook Chapter 14
- Tutorials:
  - https://www.w3schools.com/sql/default.asp
  - https://www.tutorialspoint.com/sql/index.htm
- Optional Text: Nice overview in Sebesta: Chapter 13.1, 13.2

But why RDBMS and SQL – isn't that a dead topic????

- The Relational model is dead, SQL is dead, and I don't feel so good myself:
  - https://sigmodrecord.org/publications/sigmodRecord/1306/pdfs/11.reports.atzeni.pdf

- https://blog.timescale.com/blog/why-sql-beating-nosql-what-this-means-for-future-of-data-time-series-database-348b777b847a/

Answer

- SQL and Dr Dan may be graying, but they are not yet dead or dying – they are alive and in use. (SQL is in WIDESPREAD USE)
- That is essentially what the previous two articles conclude about SQL (please read them)

From the Blog…

- Initially seduced by the dark side, the software community began to see the light and come back to SQL.
- First came the SQL interfaces on top of Hadoop (and later, Spark), leading the industry to “back-cronym” NoSQL to “Not Only SQL” (yeah, nice try).
- Then came the rise of NewSQL: new scalable databases that fully embraced SQL. **H-Store** (published 2008) from MIT and Brown researchers was one of the first scale-out OLTP databases. Google again led the way for a geo-replicated SQL-interfaced database with their first **Spanner** paper (published 2012) (whose authors include the original MapReduce authors), followed by other pioneers like **CockroachDB** (2014).
- At the same time, the **PostgreSQL** community began to revive, adding critical improvements like a JSON datatype (2012), and a potpourri of new features in PostgreSQL 10 (better native support for partitioning and replication, full text search support for JSON, etc.) and PostgreSQL 11 (added parallelized data definition capabilities, introduced just-in-time complilation, etc.), with even more to come in PostgreSQL 12. Other companies like **CitusDB** (2016, now owned by Microsoft) and yours truly (**TimescaleDB**, released in 2017) found new ways to scale PostgreSQL for specialized data workloads.

Plus, the database we will use for the course MySQL is free and in widespread use!

Relational Database Model (or Relational Data-Base Management System - RDBMS)

- Data is organized as a table--rows and columns.

- Each row in a table represents some related set of data items.

- Primary key: One of the column values is used for indexing in the table.

  This value is unique for each row in the table.

- A database may contain multiple tables, with some values common in different tables.

  The values specify a relationship between different tables

- SQL (Structured Query Language) is used to query or update the tables.

Use a Relational Database When:

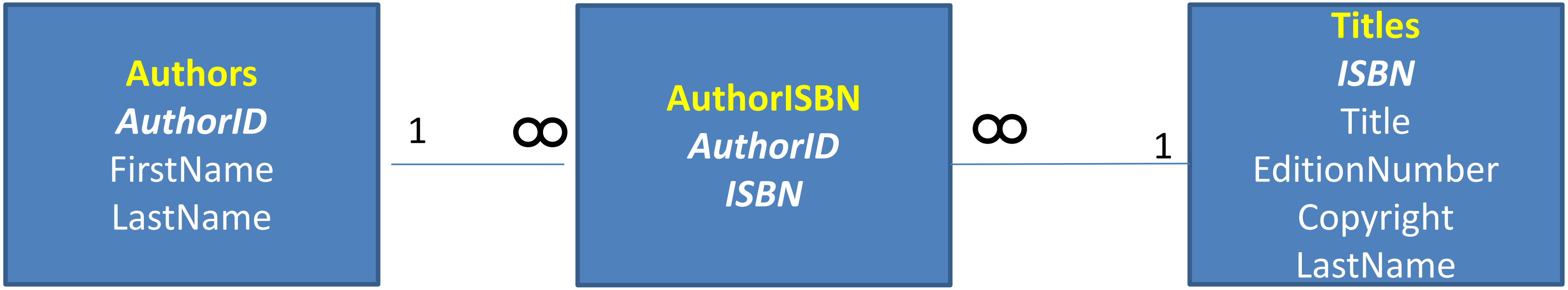
- You have relational data, e.g., you have a customer who purchases your products and those products have a supplier and manufacturer.

- You have large amounts of data and you need to be able to locate relevant information quickly.

- You need to start worrying about issues such as: scalability, reliability, ACID compliance. In computer science, ACID (Atomicity, Consistency, Isolation, Durability) is a set of properties that guarantee that database **transactions** are processed reliably.

- You need to use reporting or intelligence tools to work out business problems.

Step 1 – Design your Database Entity – Relationship Diagram (name of each entity (DB table) is in **Yellow**)



Authors
AuthorID
FirstName
LastName

AuthorISBN
AuthorID
ISBN

Titles
ISBN
Title
EditionNumber
Copyright
LastName

Primary Key:
AuthorID

Primary Key:
Composite:
AuthorID + ISBN

Primary Key
ISBN

- Three tables defined in our example:
  - Authors table
  - AuthorISBN table
  - Titles table

Example: Authors Table (1)

Three fields are defined:

authorID: **This is the primary key, defined as Integer; auto-increment assigns next integer value for this field whenever a new row is added. This field value has to be unique for each row.**

firstName: **a String containing author's first name**

lastName: **a String containing author's last name**

Example: Author’s Table (2)

authorID	firstName	lastName
1	Harvey	Deitel
2	Paul	Deitel
3	Tem	Nieto
4	Kate	Steinbuhler
5	Sean	Santry
6	Ted	Lin
7	Praveen	Sadhu
8	David	McPhie
9	Cheryl	Yaeger
10	Marina	Zlatkina
11	Ben	Wiedermann
12	Jonathan	Liperi

AuthorISBN table

- isbn String containing the ISBN of a book

- authorID ID number of one of the authors of a book whose ISBN is stored in the row

In this table, these two fields together form a unique value and therefore they are a composite primary key

AuthorISBN table

isbn	authorID
0130125075	1
0130125075	2
0130161438	1
0130161438	2
0130161438	3
0130284173	3
0130284173	6
0130284173	7
0130284181	8

Titles table

isbn	String
title	String
editionNumber	String
copyright	Year of copyright -Integer

Next Time

- HW 5
- More on RDBMS and SQL?