

CODE 301

Intermediate Software Development

AGENDA

- Assignment review
- SQL Introduction
- CRUD -- Basic database operations
- Web SQL

CRUD

Code 301

WHY PERSISTENCE?

- It would be frustrating and disappointing if all your data kept disappearing.
- Web applications need a way to store data. This process is also called persistence.
- Persistence is typically on the server, but can also be in the browser.



WHAT IS A DATABASE?

- A database is an organized collection of data.
- Database Management Systems (DBMS) have a wide variety of internal architectures, but typically they are composed of tables of data.
- Another rapidly growing alternative are documents.
- We will stick to table based databases because they are still the most common.



DATABASE TABLES

.....

id	Name	Age	Billing Rate	Hours
01	Keesha	28	75.00	40
02	Mark	42	100.00	20
03	Pam	35	123.35	10

STRUCTURED QUERY LANGUAGE (SQL)

- Structured Query Language (SQL) is a special-purpose programming language designed for managing data. Developers use SQL for inserting new data, retrieving data, updating data, and deleting data.
- SQL statements are made up of clauses, expressions, and predicates as you can see in the image below:

The diagram illustrates the components of an SQL statement using the example: `UPDATE country SET population = population + 1 WHERE name = 'USA';`

- UPDATE clause**: Points to the `UPDATE` keyword.
- country**: The table name.
- SET clause**: Points to the `SET` keyword.
- population = population + 1**: An **Expression** that defines the new value for the `population` column.
- WHERE clause**: Points to the `WHERE` keyword.
- name = 'USA'**: A **Predicate** that specifies the condition for which rows to update.
- Statement**: A bracket on the right side groups the entire query as a single SQL statement.

QUERIES

- A query retrieves data from one or more tables, or expressions.

```
SELECT isbn,  
       title,  
       price,  
       price * 0.06 AS sales_tax  
FROM Book  
WHERE price > 100.00  
ORDER BY title;
```


DATA DEFINITION LANGUAGE

- Data Definition Language (DDL) manages the table and index structure.
- The most basic statements in DDL are:
 - CREATE (http://www.w3schools.com/sql/sql_create_table.asp)
 - ALTER (http://www.w3schools.com/sql/sql_alter.asp)
 - DROP (http://www.w3schools.com/sql/sql_drop.asp)
 - TRUNCATE (http://www.w3schools.com/sql/sql_drop.asp)

DATA DEFINITION LANGUAGE

- Here's an example of create:

```
CREATE TABLE example(  
    column1 INTEGER PRIMARY KEY  
    column2 VARCHAR(50),  
    column3 DATE NOT NULL,  
);
```

DATA TYPES

- A data type is a constraint on the kind of data a column can have.
- Having strong types helps you collect accurate and valid data.
- Example types are:
 - Integer
 - Float
 - Char
 - Varchar
 - Text
 - Date
 - Time

SO HOW DOES THIS RELATE TO MVC?

WHAT IS A MODEL?

- Models are, in essence, a simplified description of a real world object. A database table is a simple model.
- In object-oriented code, models are objects. The columns correspond to properties. Here's an example constructor in JavaScript:

```
function Employee(name, age, billingRate, hours) {  
    this.name      = name;  
    this.age       = age;  
    this.billingRate = billingRate;  
    this.hours     = hours;  
}
```

WHAT IS A MODEL?

- Models can also have behavior. In our example, you could have a function (method) that calculates the amount owed per employee. It would be Billing Rate times Hours. In JavaScript it looks like:

```
Employee.prototype.amountOwed = function (billingRate, hours) {  
    return billingRate * hours  
}
```

MODELING YOUR DATA WITH SQL

- Create tables to model your objects

```
CREATE TABLE employees(  
    id INTEGER PRIMARY KEY  
    name VARCHAR(50)  
    age INTEGER,  
    billingRate FLOAT,  
    hours INTEGER  
);
```

WHAT CAN WE DO WITH THIS?

CRUD

Create, Read, Update, and Destroy



CREATE

CRUD



CREATING A TABLE

- Name the table and list the desired columns and data types.

```
CREATE TABLE articles(  
    id INTEGER PRIMARY KEY  
    title VARCHAR(50)  
    author VARCHAR(50),  
    markdown TEXT,  
    publishedOn DATETIME  
);
```

ACTION: CREATE TABLE

.....

RESULT: CREATED TABLE

.....

RESULT: CREATED TABLE

.....

id	title	author	markdown	publishedOn

INSERTING RECORDS

- Use INSERT INTO
- Name the columns you wish to affect and list the values for the record.

```
INSERT INTO articles (title, author, markdown, publishedOn)  
VALUES ('Bacon Ipsum', 'Kevin Bacon', '# hickory smoked',  
'2015-12-25');
```

ACTION: INSERT RECORD

idtitleauthor			markdown	publishedOn

RESULT: TABLE WITH NEW RECORD

idtitleauthor			markdown	publishedOn

RESULT: TABLE WITH NEW RECORD

idtitleauthor			markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22

MORE RECORDS!

id	title	author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
2	Six Degrees	Keven Bacron	# I worked with...	2013-12-13
3	Cat Ipsum	Meow Meow	# chasing cute...	2013-07-18
4	Cajun Isum	Zatarans	# boudin cajun...	2012-06-05
5	Sagan Ipsum	Carl Sagan	# distant epochs...	2014-08-01
6	Hipsters Ipsum	Macklemore	# Freegan helvetica...	2015-12-02
7	Pirate Ipsum	Wesley	# prow scuttle	2015-06-08

HOW TO GET JUST WANT WE WANT?



READ

.....

CRUD



QUERYING DATA

- Use the SELECT clause with optional constraints to build rich queries.

```
SELECT title, author, publishedOn  
FROM articles  
WHERE publishedOn BETWEEN '2013-01-01' AND '2013-12-31'  
ORDER BY publishedOn DESC;
```


ACTION: SELECT QUERY

id	title	author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
2	Six Degrees	Keven Bacron	# I worked with...	2013-12-13
3	Cat Ipsum	Meow Meow	# chasing cute...	2013-07-18
4	Cajun Isum	Zatarans	# boudin cajun...	2012-06-05
5	Sagan Ipsum	Carl Sagan	# distant epochs...	2014-08-01
6	Hipsters Ipsum	Macklemore	# Freegan helvetica...	2015-12-02
7	Pirate Ipsum	Wesley	# prow scuttle	2015-06-08

RESULT: QUERY

title	author	publishedOn
Bacon Ipsum	Kevin Bacon	2013-04-22
Cat Ipsum	Meow Meow	2013-07-18
Six Degrees	Keven Bacron	2013-12-13

RESULT: QUERY

title	author	publishedOn
Bacon Ipsum	Kevin Bacon	2013-04-22
Cat Ipsum	Meow Meow	2013-07-18
Six Degrees	Keven Bacron	2013-12-13

BAD DATA: WHAT TO DO?

title	author	publishedOn
Bacon Ipsum	Kevin Bacon	2013-04-22
Cat Ipsum	Meow Meow	2013-07-18
Six Degrees	Keven Bacron	2013-12-13



HOW CAN WE CHANGE THIS?



UPDATE

.....

CRUD



UPDATING RECORDS

- Use the UPDATE clause to alter an existing record.

```
UPDATE articles  
SET author = 'Kevin Bacon'  
WHERE author = 'Keven Bacron'
```


ACTION: UPDATE RECORD

id	title	author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
2	Six Degrees	Keven Bacron	# I worked with...	2013-12-13
3	Cat Ipsum	Meow Meow	# chasing cute...	2013-07-18

RESULT: UPDATED TABLE

id	title	author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
2	Six Degrees	Keven Bacron	# I worked with...	2013-12-13
3	Cat Ipsum	Meow Meow	# chasing cute...	2013-07-18

RESULT: UPDATED TABLE

idtitle		author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
2	Six Degrees	Kevin Bacon	# I worked with...	2013-12-13
3	Cat Ipsum	Meow Meow	# chasing cute...	2013-07-18

ALWAYS USE A CONDITION WITH UPDATE!

- You must remember to use a condition when using update to avoid affecting ALL records.

```
UPDATE articles
```

```
SET author = 'Kevin Bacon'
```

```
WHERE author = 'Keven Bacron'
```



ACTION: UPDATE RECORDS

.....

id	title	author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
2	Six Degrees	Keven Bacron	# I worked with...	2013-12-13
3	Cat Ipsum	Meow Meow	# chasing cute...	2013-07-18
4	Cajun Isum	Zatarans	# boudin cajun...	2012-06-05
5	Sagan Ipsum	Carl Sagan	# distant epochs...	2014-08-01
6	Hipsters Ipsum	Macklemore	# Freegan helvetica...	2015-12-02
7	Pirate Ipsum	Wesley	# prow scuttle	2015-06-08

RESULT:

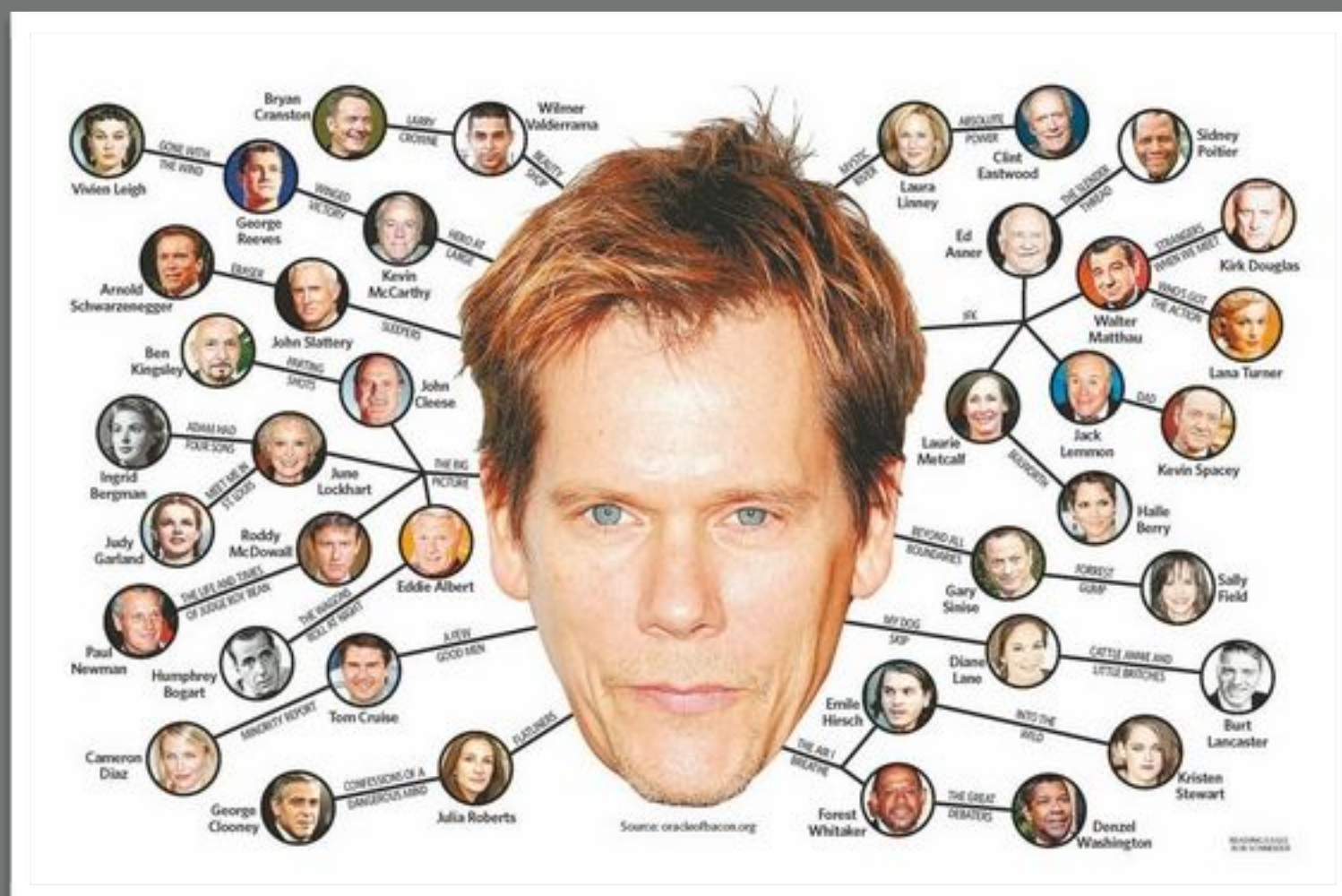
id	title	author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
2	Six Degrees	Keven Bacron	# I worked with...	2013-12-13
3	Cat Ipsum	Meow Meow	# chasing cute...	2013-07-18
4	Cajun Isum	Zatarans	# boudin cajun...	2012-06-05
5	Sagan Ipsum	Carl Sagan	# distant epochs...	2014-08-01
6	Hipsters Ipsum	Macklemore	# Freegan helvetica...	2015-12-02
7	Pirate Ipsum	Wesley	# prow scuttle	2015-06-08

RESULT: KEVIN BACON

id	title	author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
2	Six Degrees	Kevin Bacon	# I worked with...	2013-12-13
3	Cat Ipsum	Kevin Bacon	# chasing cute...	2013-07-18
4	Cajun Isum	Kevin Bacon	# boudin cajun...	2012-06-05
5	Sagan Ipsum	Kevin Bacon	# distant epochs...	2014-08-01
6	Hipsters Ipsum	Kevin Bacon	# Freegan helvetica...	2015-12-02
7	Pirate Ipsum	Kevin Bacon	# prow scuttle	2015-06-08

RESULT: KEVIN BACON

id	title	author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
2	Six Degrees	Kevin Bacon	# I worked with...	2013-12-13
3	Cat Ipsum	Kevin Bacon	# chasing cute...	2013-07-18
4	Cajun Ipsum	Kevin Bacon	# boudin cajun...	2012-06-05
5	Sagan Ipsum	Kevin Bacon	# distant epochs...	2014-08-01
6	Hipsters Ipsum	Kevin Bacon	# Freegan helvetica...	2015-12-02
7	Pirate Ipsum	Kevin Bacon	# prow scuttle	2015-06-08



WHAT IF WE WANT TO REMOVE RECORDS?



DESTROY!

CRUD



DELETING RECORDS

- Use the DELETE FROM clause to remove an existing record.

```
DELETE FROM articles
```

```
WHERE author = 'Keven Bacron'
```

ACTION: DELETE

id	title	author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
2	Six Degrees	Keven Bacron	# I worked with...	2013-12-13
3	Cat Ipsum	Meow Meow	# chasing cute...	2013-07-18

RESULT: UPDATED TABLE (EXCERPT)

.....

id	title	author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
2	Six Degrees	Keven Bacron	# I worked with...	2013-12-13
3	Cat Ipsum	Meow Meow	# chasing cute...	2013-07-18

RESULT: UPDATED TABLE (EXCERPT)

.....

id	title	author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
2	Six Degrees	Kevin Bacon	# I worked with...	2013-12-13
3	Cat Ipsum	Meow Meow	# chasing cute...	2013-07-18

RESULT: UPDATED TABLE (EXCERPT)

.....

idtitle		author	markdown	publishedOn
1	Bacon Ipsum	Kevin Bacon	# Hickory Smoked...	2013-04-22
3	Cat Ipsum	Meow Meow	# chasing cute...	2013-07-18
4	Cajun Isum	Zatarans	# boudin cajun...	2012-06-05

COOL. HOW CAN WE USE THIS ALL?



SQL IN BROWSER

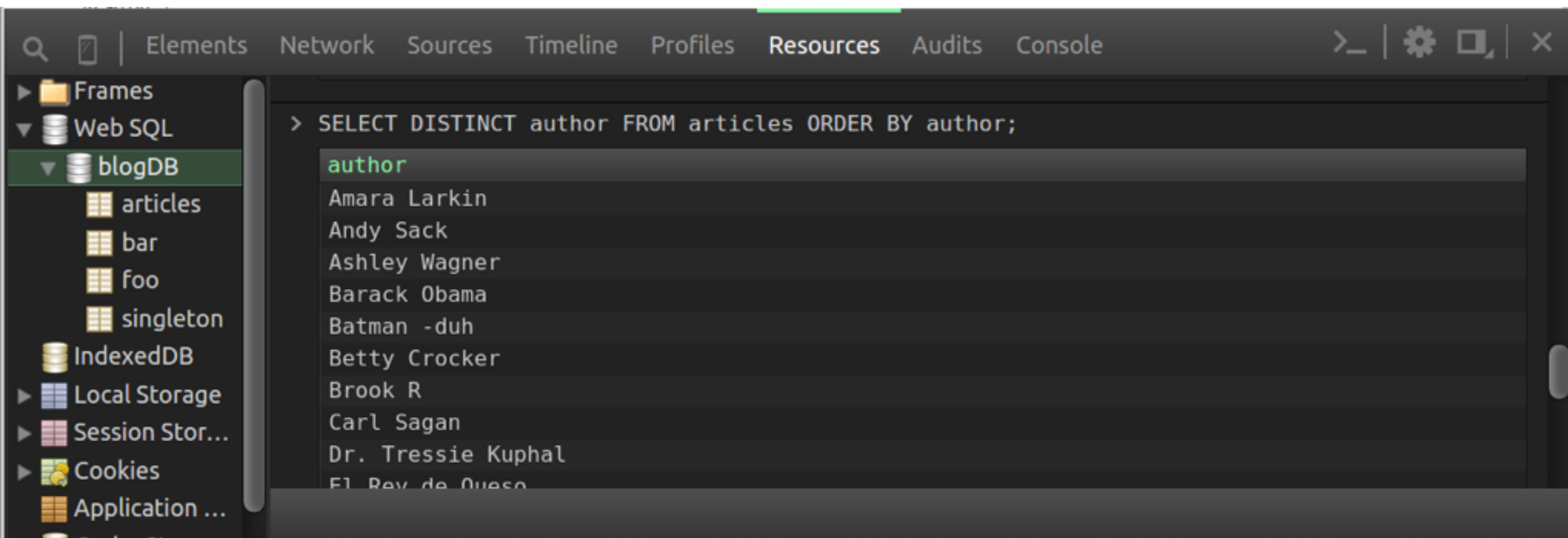
WEBSQL DATABASE

- Provides in-browser SQL functionality
- Based on SQLite 3.1.19
- Best supported by Chrome



CHROME DEV TOOLS WEB SQL

.....



WEBDB API: (REFERENCE, FYI ONLY)

- Provides a simple JavaScript interface to Web SQL.
- Uses the html5sql.js library to allow *sequential* processing of SQL statements.

```
webDB.init();  
webDB.execute();
```

WEBDB API: INIT METHOD (REFERENCE, FYI ONLY)

- Creates a connection to a Web SQL database.
- Provides useful defaults.
- Sets up success and error logging for SQL operations.

```
webDB.init();
```

WEBDB API: EXECUTE METHOD (REFERENCE, FYI ONLY)

- Allows you to execute SQL statements sequentially.
- You can specify a success callback function.

```
webDB.execute (sql, callback);
```


WEBDB API: EXECUTE METHOD (REFERENCE, FYI ONLY)

- SQL statements can be passed in many forms*.

```
webDB.execute (sql, callback);
```

* See <http://html5sql.com/guide.html> for full info.

WEBDB API: EXECUTE METHOD (REFERENCE, FYI ONLY)

- A single SQL string.

```
webDB.execute (  
    'SELECT * FROM articles;',  
    callback  
);
```

WEBDB API: EXECUTE METHOD (REFERENCE, FYI ONLY)

- A single SQL string.
- An array of SQL strings.

```
webDB.execute (  
  [  
    'DELETE FROM articles WHERE id = 10;',  
    'SELECT * FROM articles;'   
  ],  
  callback  
);
```

WEBDB API: EXECUTE METHOD (REFERENCE, FYI ONLY)

- A single SQL string.
- An array of SQL strings.
- An array of SQL objects which can *safely* take dynamic data.

```
webDB.execute (  
  [  
    {  
      sql: 'SELECT * FROM articles WHERE id = ?;',  
      data: [id]  
    }  
  ],  
  callback  
);
```


WEBDB API: EXECUTE METHOD (REFERENCE, FYI ONLY)

- The success callback will be passed a *resultsArray* argument.
- For example, to select and log all records:

```
webDB.execute (  
    "SELECT * FROM articles;",  
    function(resultsArray) {  
        console.log(resultsArray);  
    }  
);
```

GREAT. NOW SQL AND JS CAN BE FRIENDS.

RECAP



RECAP

- Databases allow powerful interaction with data
- SQL -- language for relational databases
- CRUD -- Create, Read, Update, Destroy
- Web SQL -- Access to SQL in browser



REFERENCES AND SOURCES

- A Primer on SQL: <https://leanpub.com/aprimeronsql/read>
- Introducing Web SQL: <http://html5doctor.com/introducing-web-sql-databases/>
- HTML5SQL.js Library: <http://html5sql.com/>