

# COMP60411

## Modelling Data On The Web

Tim Morris & Uli Sattler

---

Week 1 Introduction, Data Models, Tables, and SQL

# Topic Overview

- What is a (core) **data model**? E.g.,
  - **Flat**: flat files
  - **Table** based: relational
  - **Tree** based: XML and a bit of JSON
  - **Graph** based: RDF
- **Trade offs** (esp. representational) between them
  - Discussing *pain points* & *sweet spots*, distinguishing
    - principled ones from
    - DM-based ones from
    - those caused by your usage of DM

# Course Goals:

## Knowledge & Understanding

- This **course unit** aims to give you a
  - good understanding of core concepts of data modelling
  - some familiarity with formalisms, APIs, and languages
    - for modelling data on the web
    - design/representation issues that arise

# Course Goals:

## Skills

- This **course unit** aims to give you the ability/skill to
  - compare different data modelling formalisms,
  - design or analyse a data management system,
    - does it make good use of the formalism's features?
    - does it fit its purpose?

# Course Structure

- Lectures
  - Active learning
- Lab
  - Make sure you understand the coursework!
- Readings
  - All readings available online
  - Core: the "Learning" eBook series
    - Learning SQL
    - Learning XML
    - Learning SPARQL

# Our Expectations

- Lectures:
  - active listening & participation
- Lab Mondays afternoon:
  - make sure you understand the coursework!
- Lab during week:
  - work on your coursework
  - make use of TAs: 14:00-15:00
- Coursework:
  - submit on time
- Read!

# Assessment

- Coursework (50%,  $\approx$ 200 marks)
  - Each week, a mixture
    1. MCQ quizzes ( $\approx$ 10 marks)
    2. Short essays ( $\approx$ 5 marks)
    3. A modelling assignment ( $\approx$ 10 marks)
    4. A programming assignment ( $\approx$ 15 marks)
  - Precise mark breakdown varies
- Exam (50%)
  - Taken online
  - Very like 1 & 2

# Materials & Blackboard

- All course materials are available online on the materials page
- We use **Blackboard** for
  - Coursework
  - Online forums
    - Subscribe to each forum
    - Ask questions there
    - Answer questions there
    - Share examples, test cases there
  - Exam



# Variant Circumstances

- Disability (Equality Act):
  - any condition which has a significant, adverse and long-term effect on a person's ability to carry out normal day-to-day activities.
  - **Disability Advisory and Support Service**
    - Exam & Study support & more
    - Great, helpful people
- **Counselling service**
- **SSO and Mitigating Circumstances** process

...feel free to ask us: we're *happy* to advise!

# Assistance & Help

- Early intervention is more effective
  - If you are having challenges of any sort
    - the sooner they are identified *and*
    - communicated to us
    - the more likely we can find a good resolution
- This is very true for mitigating circumstances
  - If something is interfering, document it!
  - Fill out the form *when* things are happening
  - There is a "too late" here!

...when in doubt, ask us and SSO for [MitCircs](#)

# Expected Conduct

- We expect of you (and ourselves) to
  - be fair minded
  - treat each other well & with respect
  - avoid **academic malpractice**
  - take responsibility for course duties
  - be engaged, curious, and active
- If you have a problem or issue
  - please raise it with us
  - if that doesn't help, contact your programme director

# Preliminaries



*We all have to start somewhere*

# Data Management (1)

- Almost every program must do some data management
  - If only config files!
  - Many are *information heavy*
    - and must deal with that information over time
- Database Management Systems (DBMSs)
  - Separate (or separable) component
  - Specialised for variables purposed
    - secondary storage, scaling, complexity, etc.

# Data Management:

## Lifetime

- Some data is (typically) **transient** or **ephemeral**
  - Position of the cursor on the screen
- Some data is (typically) **persistent**
  - Bank records, addresses, health data, library entries
  - Cursor position can be!
    - (If you are recording the screen...)

*We're focused on data that leans toward **persistent***

# Data Management:

## Structure

- Some data is (more or less) **informationally opaque**
  - e.g., images, video, text, audio
  - its information/content isn't (easily) available
    - You typically must do some *extraction*
  - this is called **unstructured data**
- Some data is **informationally transparent**
  - its information/content is programmatically explicit
  - this is called **(semi-)structured data**



# Out Of Scope

- There is lots of DM that's outside our scope
  1. Performance & Scaling: see [COMP62421](#)
  2. Concurrency
    - Thus *transactions*
      - (You should read up on ACIDity)
  3. Tuning, indeed most physical level stuff
  4. Cleansing
  5. Integration
    - Except for a tiny bit, around *merging*

These considerations *do* affect modelling!



# Data And The Web

- The Web is a collaborative information structure
  - Largely decentralised
  - Immense
  - Growing rapidly
  - Changing rapidly
- The Web produces new data challenges
  - Scale of data
  - Kind of data
  - Shape of data
  - Use of data

# Data on, from, behind the Web

- **On** the Web
  - data.gov, data.gov.uk, ...
- **From** the Web
  - Log files
- **Behind** the Web
  - Data(base) backed Websites
    - The filesystem is a kind of database
  - Content Management Systems
    - Wordpress
  - Sites as Database Front Ends
    - See Amazon

# What is a Data Model?

- Three Key Aspects
  1. Underlying Data Structure, "**Core Data Model**"
  2. Data Integrity
  3. Data Manipulation
  4. (Plus a fourth!) Data Sharing
    - More important on the Web \*

# "Data Model" is Ambiguous:

1. a complete data representation and manipulation approach (we do this!)
2. just the **core data model**
3. a particular data representation for a domain or application, also called the **domain model**
  - "Does your calendar data model include leap years?"

Generally, you can tell from context, (2) is rare.

# Kinds of Data

- Data can lend itself to different **shapes**
  - Array-like
  - Tree-like
  - Graph-like
  - Document-like
- Data can have different **volumes**
  - Small to "big" data
- Data can have different **velocities**
  - Static/offline to streaming
- Data can have different **use patterns**
  - Many readers/few writers or the reverse or other!

# Data Does Not Grow on Trees

- Data may lend itself to one shape
  - e.g., tree-shape or graph-shape
- but this does **not** mean that
  - we have to persist it in this form
  - we know exactly how to cast it in this form
  - ...consider **pain-points** and **sweet spots**
  - others share it in this form

# Polyglot Persistence

*...we are gearing up for a shift to **polyglot persistence** — where any decent sized enterprise will have a **variety of different data storage technologies for different kinds of data**. There will still be large amounts of it managed in relational stores, but increasingly we'll be **first asking how we want to manipulate the data** and only then figuring out what technology is the best bet for it.*

*— Martin Fowler*

# Polyglot Persistence (2)

*This **polyglot [e]ffect** will be apparent even within a single application. A complex enterprise application uses different kinds of data, and already usually integrates information from different sources. **Increasingly we'll see such applications manage their own data using different technologies depending on how the data is used.***

*— Martin Fowler*



# Poly -Glot/-System Persistence

- Even a **single** core data model can result in
  - multiple systems with different characteristics
  - multiple, overlapping, domain models
  - multiple, overlapping owners, versions, variants

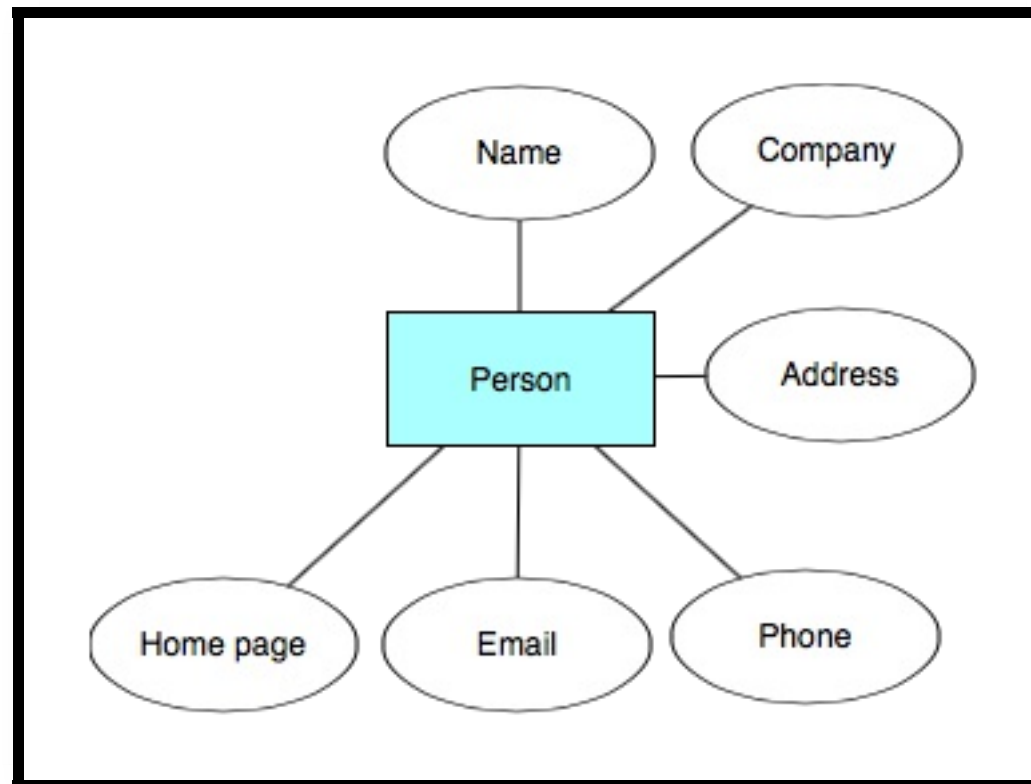
This is particularly true in on the Web!

# "Flat Files" – A Simple Model



# A Sample Domain

- We start with a classic example: The Address Book
  - People and information about them
  - Names and contact information
- We can do a first cut as a diagram



# For Example

- Bijan!
  - Name: Bijan Parsia
  - Company: University of Manchester
  - Email: [bijan.parsia@manchester.ac.uk](mailto:bijan.parsia@manchester.ac.uk)
  - ...
- Uli!
  - Name: Uli Sattler
  - Company: University of Manchester
  - Email: [uli.sattler@manchester.ac.uk](mailto:uli.sattler@manchester.ac.uk)

# Storing!



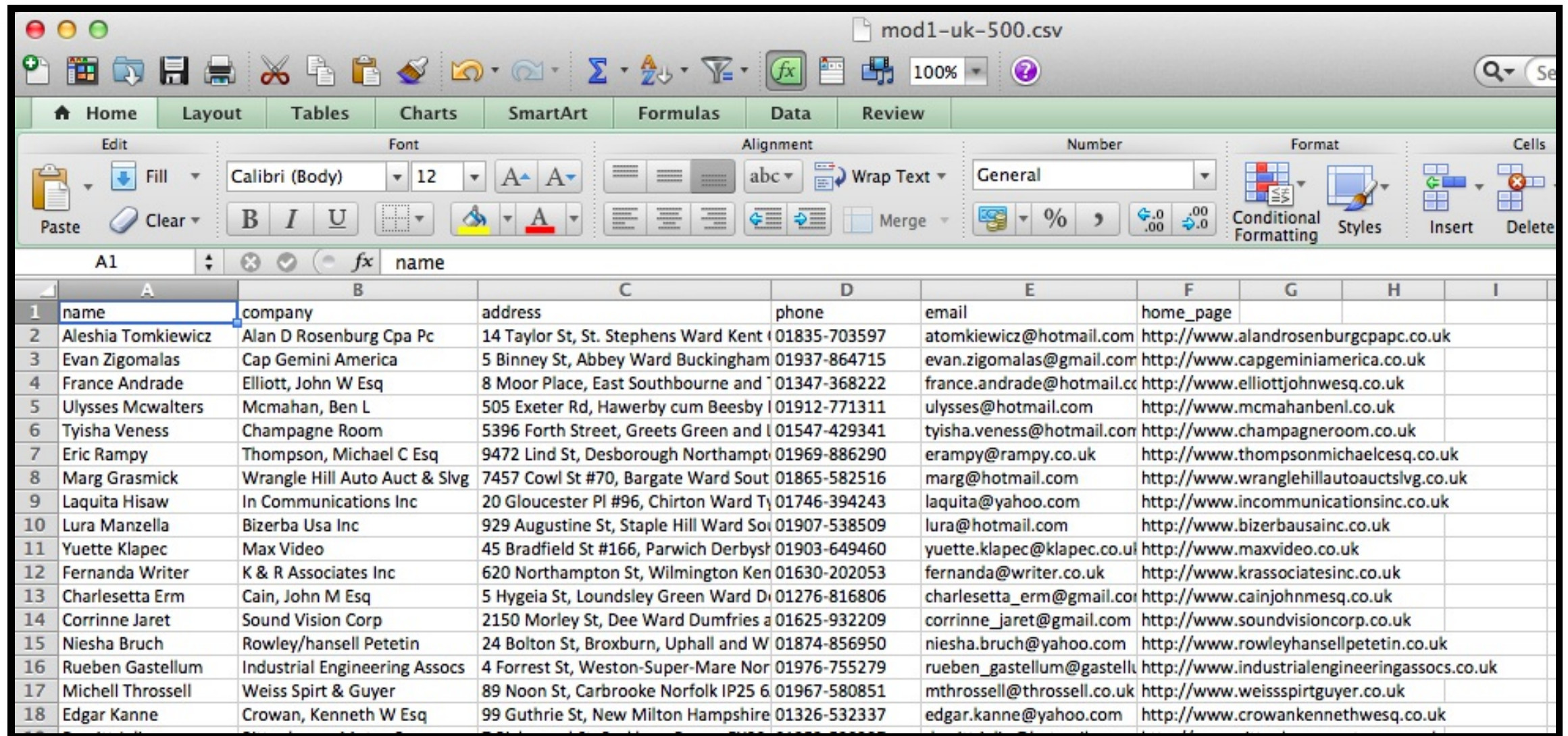
- Slides are not a good storage place for data
- We have an array like structure so...
  - How about a spreadsheet!
    - 1 entity/record/person per **row**
    - Each field/attribute is a **column**
- We have software that works well with this!

The screenshot shows a Microsoft Excel spreadsheet titled 'mod1-uk-500.csv'. The spreadsheet contains 18 rows of data. The columns are labeled 'name', 'company', 'address', 'phone', 'email', and 'home\_page'. The data includes names, companies, addresses, phone numbers, email addresses, and website URLs.

	A	B	C	D	E	F	G	H	I
1	name	company	address	phone	email	home_page			
2	Aleshia Tomkiewicz	Alan D Rosenberg Cpa Pc	14 Taylor St, St. Stephens Ward Kent	01835-703597	atomkiewicz@hotmail.com	http://www.alandrosenburgcpapc.co.uk			
3	Evan Zigomalas	Cap Gemini America	5 Binney St, Abbey Ward Buckingham	01937-864715	evan.zigomalas@gmail.com	http://www.capgeminiamerica.co.uk			
4	France Andrade	Elliott, John W Esq	8 Moor Place, East Southbourne and	01347-368222	france.andrade@hotmail.co	http://www.elliottjohnwesq.co.uk			
5	Ulysses Mcwalters	Mcmahan, Ben L	505 Exeter Rd, Hawerby cum Beesby	01912-771311	ulysses@hotmail.com	http://www.mcmahanbenl.co.uk			
6	Tyisha Veness	Champagne Room	5396 Forth Street, Greets Green and	01547-429341	tyisha.veness@hotmail.com	http://www.champagneroom.co.uk			
7	Eric Rampy	Thompson, Michael C Esq	9472 Lind St, Desborough Northampt	01969-886290	erampy@rampy.co.uk	http://www.thompsonmichaelcesq.co.uk			
8	Marg Grasmick	Wrangle Hill Auto Auct & Slvg	7457 Cowl St #70, Bargate Ward Sout	01865-582516	marg@hotmail.com	http://www.wranglehillautoauctslvg.co.uk			
9	Laquita Hisaw	In Communications Inc	20 Gloucester Pl #96, Chirton Ward T	01746-394243	laquita@yahoo.com	http://www.incommunicationsinc.co.uk			
10	Lura Manzella	Bizerba Usa Inc	929 Augustine St, Staple Hill Ward So	01907-538509	lura@hotmail.com	http://www.bizerbausainc.co.uk			
11	Yvette Klapec	Max Video	45 Bradfield St #166, Parwich Derbys	01903-649460	yvette.klapec@klapec.co.uk	http://www.maxvideo.co.uk			
12	Fernanda Writer	K & R Associates Inc	620 Northampton St, Wilmington Ken	01630-202053	fernanda@writer.co.uk	http://www.krassociatesinc.co.uk			
13	Charlesetta Erm	Cain, John M Esq	5 Hygeia St, Loundsley Green Ward D	01276-816806	charlesetta_erm@gmail.co	http://www.cainjohnmesq.co.uk			
14	Corrinne Jaret	Sound Vision Corp	2150 Morley St, Dee Ward Dumfries a	01625-932209	corrinne_jaret@gmail.com	http://www.soundvisioncorp.co.uk			
15	Niesha Bruch	Rowley/hansell Petetin	24 Bolton St, Broxburn, Uphall and W	01874-856950	niesha.bruch@yahoo.com	http://www.rowleyhansellpetetin.co.uk			
16	Rueben Gastellum	Industrial Engineering Assocs	4 Forrest St, Weston-Super-Mare Nor	01976-755279	rueben_gastellum@gastellu	http://www.industrialengineeringassocs.co.uk			
17	Michell Throssell	Weiss Spirt & Guyer	89 Noon St, Carbrooke Norfolk IP25 6	01967-580851	mthrossell@throssell.co.uk	http://www.weisspirtguyer.co.uk			
18	Edgar Kanne	Crowan, Kenneth W Esq	99 Guthrie St, New Milton Hampshire	01326-532337	edgar.kanne@yahoo.com	http://www.crowankennethwesq.co.uk			



# Interacting With The Data



The screenshot shows a Microsoft Excel spreadsheet with the following data:

	A	B	C	D	E	F	G	H	I
1	name	company	address	phone	email	home_page			
2	Aleshia Tomkiewicz	Alan D Rosenberg Cpa Pc	14 Taylor St, St. Stephens Ward Kent	01835-703597	atomkiewicz@hotmail.com	<a href="http://www.alandrosenburgcpapc.co.uk">http://www.alandrosenburgcpapc.co.uk</a>			
3	Evan Zigomalas	Cap Gemini America	5 Binney St, Abbey Ward Buckingham	01937-864715	evan.zigomalas@gmail.com	<a href="http://www.capgeminiamerica.co.uk">http://www.capgeminiamerica.co.uk</a>			
4	France Andrade	Elliott, John W Esq	8 Moor Place, East Southbourne and	01347-368222	france.andrade@hotmail.co	<a href="http://www.elliottjohnwesq.co.uk">http://www.elliottjohnwesq.co.uk</a>			
5	Ulysses Mcwalters	Mcmahan, Ben L	505 Exeter Rd, Hawerby cum Beesby	01912-771311	ulysses@hotmail.com	<a href="http://www.mcmahanbenl.co.uk">http://www.mcmahanbenl.co.uk</a>			
6	Tyisha Veness	Champagne Room	5396 Forth Street, Greets Green and	01547-429341	tyisha.veness@hotmail.com	<a href="http://www.champagneroom.co.uk">http://www.champagneroom.co.uk</a>			
7	Eric Rampy	Thompson, Michael C Esq	9472 Lind St, Desborough Northampt	01969-886290	erampy@rampy.co.uk	<a href="http://www.thompsonmichaelcesq.co.uk">http://www.thompsonmichaelcesq.co.uk</a>			
8	Marg Grasmick	Wrangle Hill Auto Auct & Slvg	7457 Cowl St #70, Bargate Ward Sout	01865-582516	marg@hotmail.com	<a href="http://www.wranglehillautoauctslvg.co.uk">http://www.wranglehillautoauctslvg.co.uk</a>			
9	Laquita Hisaw	In Communications Inc	20 Gloucester Pl #96, Chirton Ward Ty	01746-394243	laquita@yahoo.com	<a href="http://www.incommunicationsinc.co.uk">http://www.incommunicationsinc.co.uk</a>			
10	Lura Manzella	Bizerba Usa Inc	929 Augustine St, Staple Hill Ward Sov	01907-538509	lura@hotmail.com	<a href="http://www.bizerbausainc.co.uk">http://www.bizerbausainc.co.uk</a>			
11	Yvette Klapac	Max Video	45 Bradfield St #166, Parwich Derbys	01903-649460	yvette.klapac@klapac.co.uk	<a href="http://www.maxvideo.co.uk">http://www.maxvideo.co.uk</a>			
12	Fernanda Writer	K & R Associates Inc	620 Northampton St, Wilmington Ken	01630-202053	fernanda@writer.co.uk	<a href="http://www.krassociatesinc.co.uk">http://www.krassociatesinc.co.uk</a>			
13	Charlesetta Erm	Cain, John M Esq	5 Hygeia St, Loundsley Green Ward D	01276-816806	charlesetta_erm@gmail.co	<a href="http://www.cainjohnmesq.co.uk">http://www.cainjohnmesq.co.uk</a>			
14	Corrinne Jaret	Sound Vision Corp	2150 Morley St, Dee Ward Dumfries a	01625-932209	corrinne_jaret@gmail.com	<a href="http://www.soundvisioncorp.co.uk">http://www.soundvisioncorp.co.uk</a>			
15	Niesha Bruch	Rowley/hansell Petetin	24 Bolton St, Broxburn, Uphall and W	01874-856950	niesha.bruch@yahoo.com	<a href="http://www.rowleyhansellpetetin.co.uk">http://www.rowleyhansellpetetin.co.uk</a>			
16	Rueben Gastellum	Industrial Engineering Assocs	4 Forrest St, Weston-Super-Mare Nor	01976-755279	rueben_gastellum@gastellu	<a href="http://www.industrialengineeringassocs.co.uk">http://www.industrialengineeringassocs.co.uk</a>			
17	Michell Throssell	Weiss Spirt & Guyer	89 Noon St, Carbrooke Norfolk IP25 6	01967-580851	mthrossell@throssell.co.uk	<a href="http://www.weisspirtguyer.co.uk">http://www.weisspirtguyer.co.uk</a>			
18	Edgar Kanne	Crowan, Kenneth W Esq	99 Guthrie St, New Milton Hampshire	01326-532337	edgar.kanne@yahoo.com	<a href="http://www.crowankennethwesq.co.uk">http://www.crowankennethwesq.co.uk</a>			

To the demo!

# Pain points

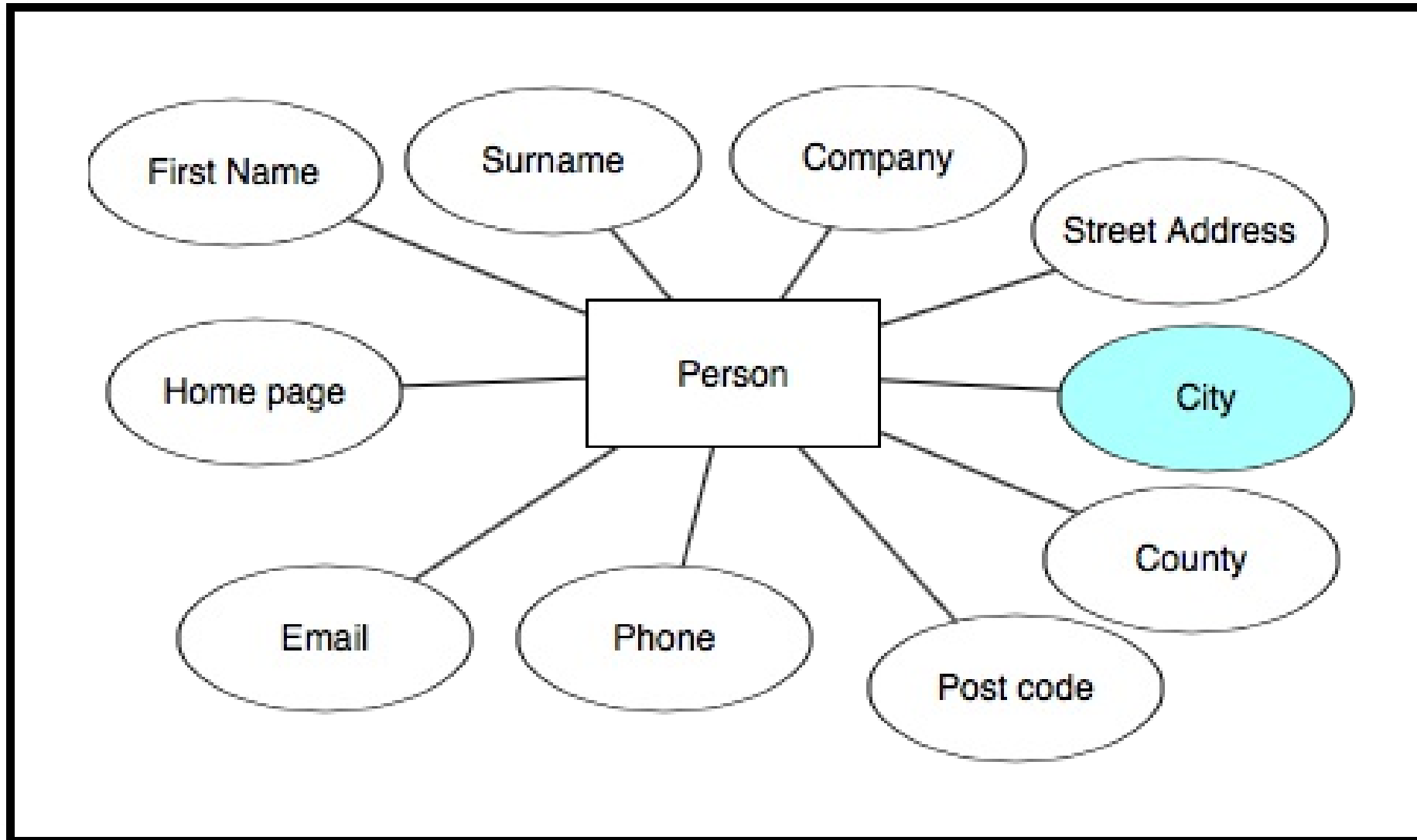
- Around "name"
  - Sorting is on **columns**
    - Cannot sort by **surname**
  - Filtering: can filter by **names** beginning with **Z**
    - Cannot filter by **surnames** beginning with **Z**
- Around "address"
  - Can't sort or filter by **postcode**
  - Can't sort or filter by **city**
  - Can't sort or filter by **county**

Problems with **spreadsheets** or **our format?**



# Format 2

- This should fix our pain points!



# Interacting!

The screenshot shows a Microsoft Excel spreadsheet with the following data:

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
1	first_name	surname	company_na	address	city	county	postal	phone	email	web				
2	Socorro	Abrahams	Martin Morr	93 Clyde Rd	Deepdale W	Lancashire	PR1 6TN	01311-56705	socorro_abr	<a href="http://www.martinmorrissey.co.uk">http://www.martinmorrissey.co.uk</a>				
3	Rusty	Adelsperger	Clarke, Jame	4313 Princes	Launceston	Cornwall	PL15 9QN	01467-17255	rusty.adelspe	<a href="http://www.clarkejameshesq.co.uk">http://www.clarkejameshesq.co.uk</a>				
4	Olen	Ailey	Shohet, Grac	9 Fielding St	Wombourne	Staffordshire	WV5 0BB	01654-86555	olen@gmail.	<a href="http://www.shohetgracecesq.co.uk">http://www.shohetgracecesq.co.uk</a>				
5	Letha	Akey	Jeanettes Dr	603 Pall Mall	Layton Ward	Lancashire	FY3 8ND	01694-42420	letha_akey@	<a href="http://www.jeanettesdraperyupholstery.co.uk">http://www.jeanettesdraperyupholstery.co.uk</a>				
6	Margret	Alcazar	Advantage T	5466 Sedley	Coatbridge S	North Lanark	ML5 4LJ	01626-59077	margret@alc	<a href="http://www.advantageitleescrowinc.co.uk">http://www.advantageitleescrowinc.co.uk</a>				
7	Nettie	Aldaco	Miller Searl	51 Freehold	Wheatley W	Yorkshire, So	DN2 4PP	01388-97491	nettie.aldacc	<a href="http://www.millersearlfitch.co.uk">http://www.millersearlfitch.co.uk</a>				
8	Latosha	Alexy	Laitinen, Ste	37 Langham	St. Georges	Surrey	KT13 0AZ	01704-50806	latosha@yah	<a href="http://www.laitinenstephenbesq.co.uk">http://www.laitinenstephenbesq.co.uk</a>				
9	Lemuel	Allis	Computer Se	8430 Shadwe	Great Barr w	West Midlan	WS5 4SU	01580-25286	lemuel_allis	<a href="http://www.computersecuritycnslnsinc.co.uk">http://www.computersecuritycnslnsinc.co.uk</a>				
10	Phillip	Aloi	Duffield, Mic	6 Cannock St	Scarcroft	West Yorkshi	LS14 3BW	01490-89817	paloi@hotm	<a href="http://www.duffieldmichaelc.co.uk">http://www.duffieldmichaelc.co.uk</a>				
11	Mira	Alpheaus	East County	51 St Anne S	Stratfield Mc	Berkshire	RG7 3RA	01241-27395	mira.alpheau	<a href="http://www.eastcountyprocess.co.uk">http://www.eastcountyprocess.co.uk</a>				
12	Ahmad	Alsaqri	Alliance Con	21 Pickwick S	Sutton cum	Derbyshire	S44 5DS	01567-55557	ahmad.alsaq	<a href="http://www.allianceconstructioncoinc.co.uk">http://www.allianceconstructioncoinc.co.uk</a>				
13	Kandis	Alsbury	Fast Cash	70 Rose Vale	Reydon	Suffolk	IP18 6PE	01797-83727	kalsbury@hc	<a href="http://www.fastcash.co.uk">http://www.fastcash.co.uk</a>				
14	Luther	Alsman	Crossroads T	227 Albert T	Belvedere W	Greater Lonc	DA17 6EF	01536-63925	luther@gma	<a href="http://www.crossroadstravelserviceinc.co.uk">http://www.crossroadstravelserviceinc.co.uk</a>				
15	Janella	Altobell	Shannon, Pa	3768 Hey Gr	Hartshill	Warwickshire	CV10 0TH	01746-50536	jaltobell@ho	<a href="http://www.shannonpaulvesq.co.uk">http://www.shannonpaulvesq.co.uk</a>				
16	France	Andrade	Elliott, John	8 Moor Place	East Southbc	Bournemouth	BH6 3BE	01347-36822	france.andra	<a href="http://www.elliottjohnwesq.co.uk">http://www.elliottjohnwesq.co.uk</a>				
17	Alyssa	Ansbro	Berg, Michael	85 Hero St	Stanhope	County Durh	DL13 2TZ	01632-88782	alyssa_ansbr	<a href="http://www.bergmichaeldesq.co.uk">http://www.bergmichaeldesq.co.uk</a>				
18	Narcisa	Araiza	Danka Busin	8783 High St	Milton	Cambridgesh	CB24 6ZR	01724-64476	naraiza@hot	<a href="http://www.dankabusinesssystem.co.uk">http://www.dankabusinesssystem.co.uk</a>				
19	Nada	Arey	Advanced En	22 Harewood	Acton Trusse	Staffordshire	ST17 0RU	01576-62782	nada@hotmail	<a href="http://www.advancedengineeringassocs.co.uk">http://www.advancedengineeringassocs.co.uk</a>				
20	Lashunda	Argiro	Kluza Associa	205 Forge St	Stainburn	North Yorksh	LS21 2LS	01422-72814	lashunda@y	<a href="http://www.kluzaassociates.co.uk">http://www.kluzaassociates.co.uk</a>				
21	Remedios	Arlinghaus	Miller, Marti	9 Duckenfiel	Aldbrough	E Riding of Y	HU11 4QA	01536-49875	remedios.arl	<a href="http://www.millermartinmesq.co.uk">http://www.millermartinmesq.co.uk</a>				
22	Ivan	Aronov	Center For P	94 Regent St	Over Kellet	Lancashire	LA6 1DB	01478-39225	ivan@gmail.	<a href="http://www.centerforpediatrics.co.uk">http://www.centerforpediatrics.co.uk</a>				

Demo encore!

# New Pain Points

- Variable numbers of the "same" attribute
  - Phone number
  - Email address
  - Web page
  - Inserting columns is painful
    - Lots of partial columns
    - Sheer number sucks
- Companies have addresses!
  - More than one!
  - And phone numbers, etc.

*More problems with our format*



# NOT A New Format

- **Not** a fix to our format:

	B	C	D	E	F	G	H	I	J	K	L	M
1	last_name	company_na	address	city	county	postal	phone1	phone2	email	web		
2	Tomkiewicz	Alan D Roser	14 Taylor St	St. Stephens	Kent	CT2 7PP	01835-70355	01944-36996	atomkiewicz	<a href="http://www.alandrosenburgcpapc.co.uk">http://www.alandrosenburgcpapc.co.uk</a>		
3	Zigomas	Cap Gemini /	5 Binney St	Abbey Ward	Buckingham	HP11 2AX	01937-86471	01714-73766	evan.zigoma	<a href="http://www.cappgeminiamerica.co.uk">http://www.cappgeminiamerica.co.uk</a>		

# Fixing The Format Again

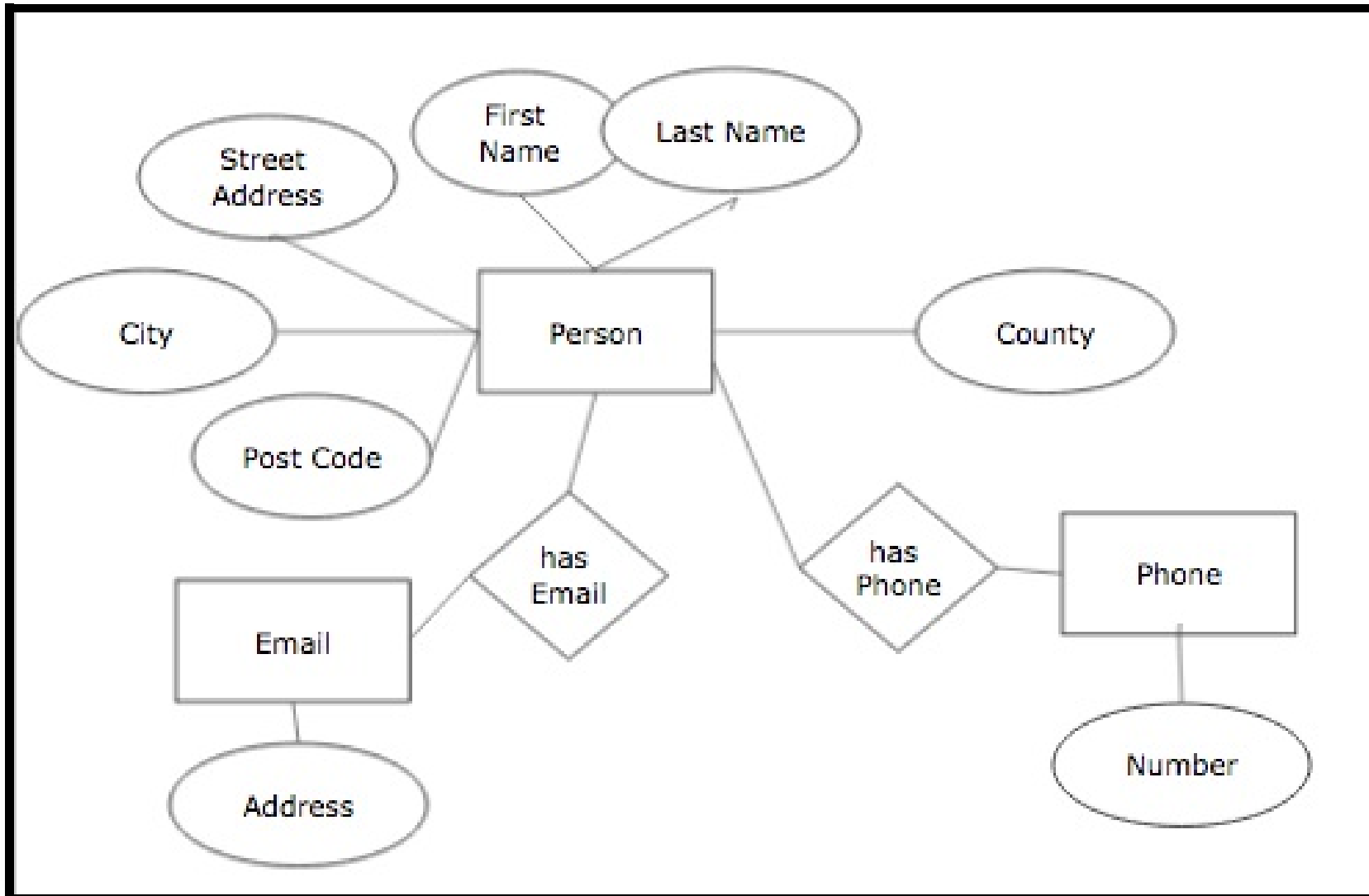
- We want adding a (similar) column to be easy!
  - Easy as adding a row!
  - Make a *new table* just for phone numbers
  - Index numbers with person rows

	A	B	C	D	E	F	G	H	I	J	K	L	M
1	first_name	surname	company_name	address	city	county	postal		Row	phone		Row	email
2	Socorro	Abrahams	Martin Morrison	93 Clyde Rd	Deepdale W	Lancashire	PR1 6TN		2	01311-567052		3	socorro_abrahams@abrahams.co.
3	Rusty	Adelsperger	Clarke, James	4313 Princes	Launceston	Cornwall	PL15 9QN		2	01311-182590		3	rusty.adelsperger@yahoo.com
4	Olen	Ailey	Shohet, Grace	9 Fielding St	Wombourne	Staffordshire	WV5 0BB		2	01234-865543		2	olen@gmail.com
5	Letha	Akey	Jeanettes Dr	603 Pall Mall	Layton Ward	Lancashire	FY3 8ND		3	01694-424205		1	letha_akey@akey.co.uk
6	Margret	Alcazar	Advantage Tr	5466 Sedley	Coatbridge S	North Lanark	ML5 4LJ		3	01626-590776		2	margret@alcazar.co.uk
7	Nettie	Aldaco	Miller Searl &	51 Freehold	Wheatley W	Yorkshire, So	DN2 4PP			01388-974910			nettie.aldaco@yahoo.com
8	Latosha	Alexy	Laitinen, Steve	37 Langham	St. Georges P	Surrey	KT13 0AZ			01704-508066			latosha@yahoo.com



# Format 3

- Now this should fix our pain points!



# Still Pain Points

- Sorting **destroys** the relationship
  - We used row numbers to connect
  - Sorting changes the row number!
- Hard to see the record
- No longer a simple flat file
  - CSV format makes assumptions

These are (mostly) **implementation** problems!



# Analyse Format Failure

- Did we
  - get the domain wrong (addresses)?
  - fit it wrong into our core DM (tables)?
  - pick the wrong core DM to model it in?
- Is our format
  - unworkable?
  - workable but requires a lot of application code?
  - reasonable with some workarounds?
- How much **technical debt** are we piling up?
- What's the **cost of switching**?

# Unsuitable Core Data Model

- If you are
  - always "fighting" the system
  - use lots of application code to hack things
  - live in an error rich environment
  - have increasing amounts of workaround support in your data

Your core data model might not be a good fit for your domain and application!

# The Rest Of The DBMS

- Even if your core DM isn't a good fit, you might
  - be stuck with the system
    - You paid good money for that Oracle database!
  - need features of the implementation
    - is there an XML database with transactions?
    - what's the support contract?
  - be stuck with the model (critical legacy apps)
- Just because the **model** is broken doesn't mean that the **system** is
  - Or is **broken enough** to justify a switch

# Flat File Programming



# Sharing Our Databases

- Spreadsheets?
  - **Proprietary-ish** (Excel, Google Doc, OpenOffice)
- Lingua franca: **CSV**
  - Comma (or Tab) Delimited Values
  - *Exactly* the (pure) flat file model
  - Format: text file
    - 1 record per line
    - First line can be special (column names)
    - Each column separated by a ","
      - We may need to quote cells (with commas)

# CSV Example

```
uk-500.csv
1 "first_name","last_name","company_name","address","city","county","postal","phone1","phone2","email","web"
2 "Aleshia","Tomkiewicz","Alan D Rosenburg Cpa Pc","14 Taylor St","St. Stephens Ward","Kent","CT2 7PP","0183"
3 "Evan","Zigomas","Cap Gemini America","5 Binney St","Abbey Ward","Buckinghamshire","HP11 2AX","01937-864"
4 "France","Andrade","Elliott, John W Esq","8 Moor Place","East Southbourne and Tuckton W","Bournemouth","BH"
5 "Ulysses","Mcwalters","Mcmahan, Ben L","505 Exeter Rd","Hawerby cum Beesby","Lincolnshire","DN36 5RP","019"
6 "Tyisha","Veness","Champagne Room","5396 Forth Street","Greets Green and Lyng Ward","West Midlands","B70 9"
7 "Eric","Rampy","Thompson, Michael C Esq","9472 Lind St","Desborough","Northamptonshire","NN14 2GH","01969-"
8 "Marg","Grasmick","Wrangle Hill Auto Auct & Slvg","7457 Cowl St #70","Bargate Ward","Southampton","S014 3"
9 "Laquita","Hisaw","In Communications Inc","20 Gloucester Pl #96","Chirton Ward","Tyne & Wear","NE29 7AD",""
10 "Lura","Manzella","Bizerba Usa Inc","929 Augustine St","Staple Hill Ward","South Gloucestershire","BS16 4"
11 "Yvette","Klapec","Max Video","45 Bradfield St #166","Parwich","Derbyshire","DE6 1QN","01903-649460","019"
12 "Fernanda","Writer","K & R Associates Inc","620 Northampton St","Wilmington","Kent","DA2 7PP","01630-2020"
13 "Charlesetta","Erm","Cain, John M Esq","5 Hygeia St","Loundsley Green Ward","Derbyshire","S40 4LY","01276-
```

# Programmatic Manipulation

- If we store our databases as CSV
  - We can load and parse them into structures
  - Manipulate our data from **our** programs
- E.g., using Python

```
import csv
with open("../Adresses/mod2-uk-500.csv") as csvfile:
    line_count = 0
    myreader = csv.reader(csvfile, delimiter=',', quotechar='t')
    for row in myreader:
        if line_count == 0:
            line_count += 1
        else:
            print(f'Candidate {line_count}: Firstname {row[0]} Lastname {row[1]} City {row[4]}')
            line_count += 1
print(f'Processed {line_count - 1} Candidates.')
```

# Solving Problems

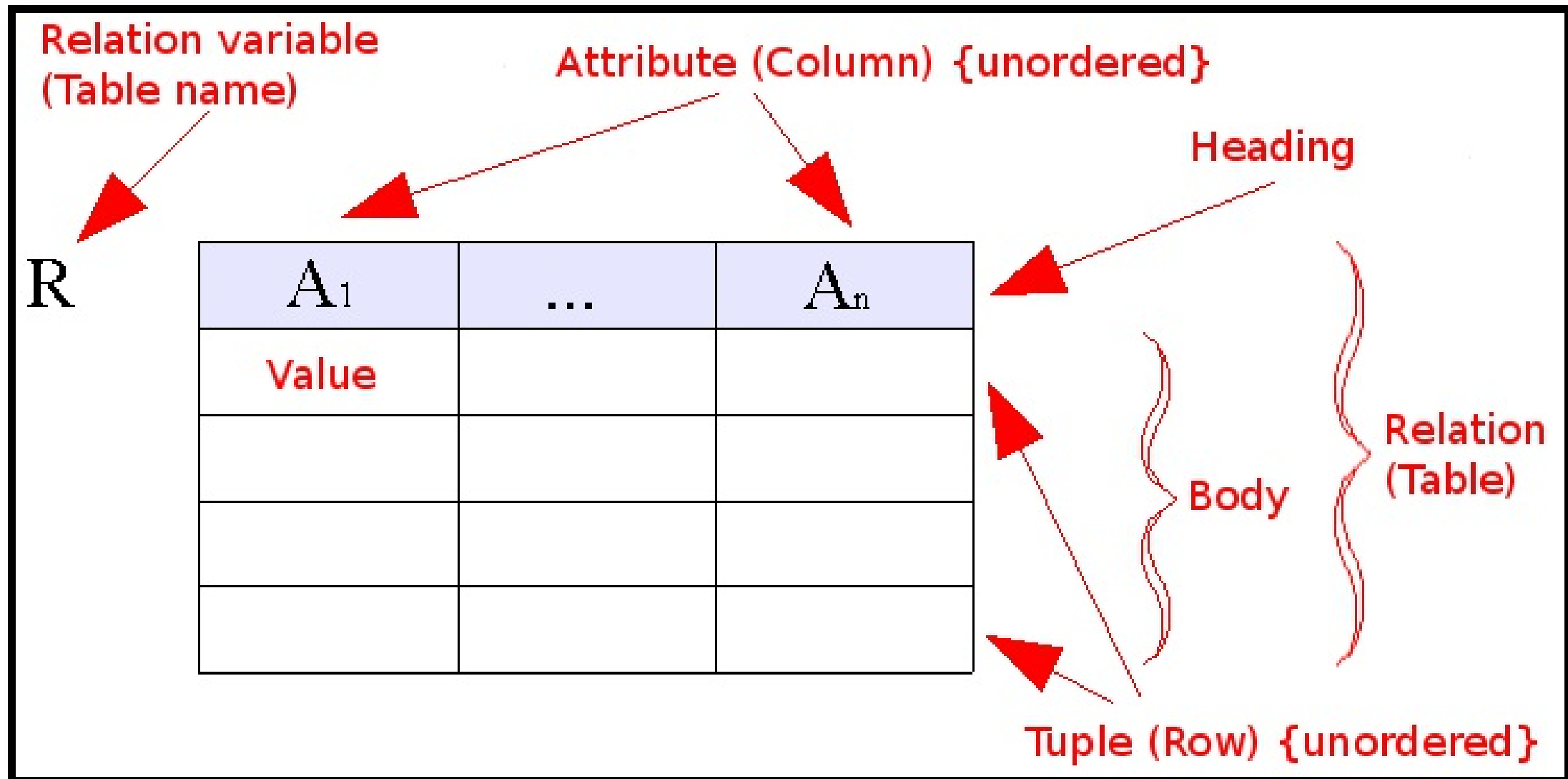
- This solves some problems!
  - Inserting/removing columns a "small matter of programming"
    - Or we could use multiple arrays with pointers
  - We can split/combine fields at will
    - Well, with a bit of programming
  - We can control sorting well enough
    - Use pointers to connect
- Lots of work!



# Against Bespoke Programming

- This is all at the wrong level
  - Flat files and flat file++ are ubiquitous
  - We shouldn't be coding complex functions
    - Over and over again!
- Even if we can program our way around problems
  - Doesn't eliminate the problems
  - Some solutions (pointers) effectively change the core model: no longer flat files!

# A Relational Model



# Tables

- A core DM where **table** (or **relation**) is the core data structure
  - A table is a **set of tuples**
  - A tuple is
    - an n-ary **sequence**
    - a **set** of key-value **pairs**
- Flat file had **one** table
  - We allow many!
  - **Named** tables
  - Aka **relations**

# Relations!

- (We use **table** and **relation** interchangeably)
- Relations are like First Order Logic (FOL) **predicates**
  - Relation name = Predicate name
  - Number of columns = Arity of predicate
    - Person(bijan, u\_o\_manchester, ...)
  - Predicate is **true** (or false!) of its arguments
    - Relation is "true" of tuples which occur in it
  - Predicates can have
    - **definitions** (intensional!)
    - **facts** (extensional!)

# Order and Identity

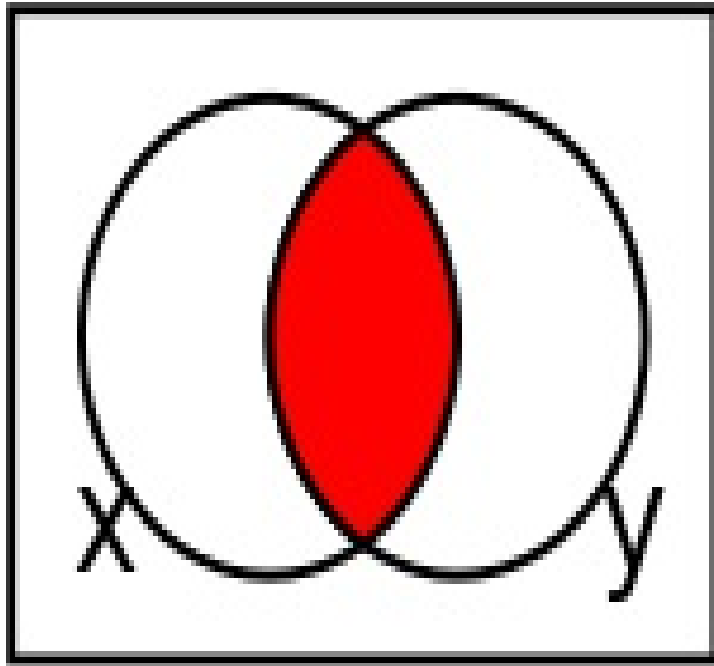
Records/Rows/Entities need **identity**

- In Excel, we had the **row label**
  - the order or position of a record was significant
- In our model, we need **distinguishing attributes**
  - we push identity *into* the data: a **key**
    - either a "naturally" unique set of attributes
    - or a made up one: an **ID**
- **Order** is always a property of the
  - data values
  - implementation

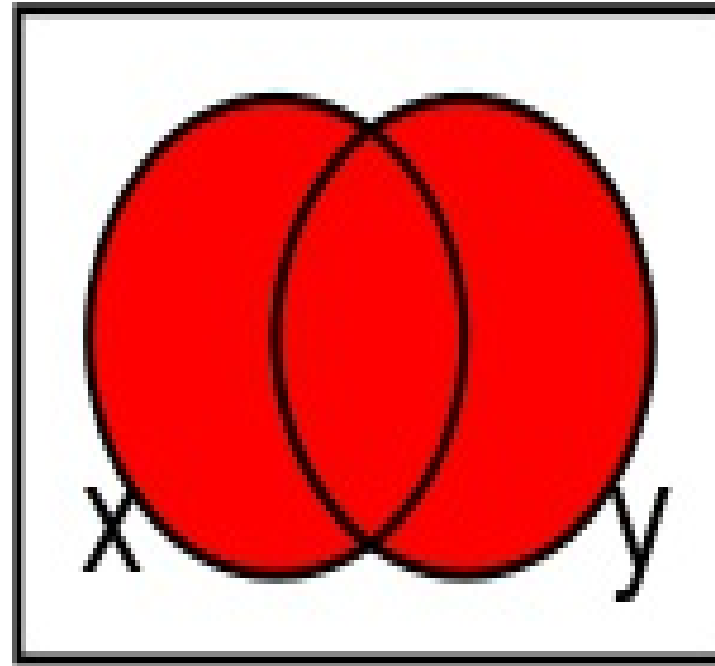
# Multiple Tables

- Actions on multiple tables:
  - **Splitting** at
    - design time: try to normalize your DB
    - run time: dropping bits
  - **Combining**
    - Take two tables and produce a new table
- The key to relational domain modelling
  - **Decompose** your problem into "base" tables
  - **Derive** new tables for specific needs

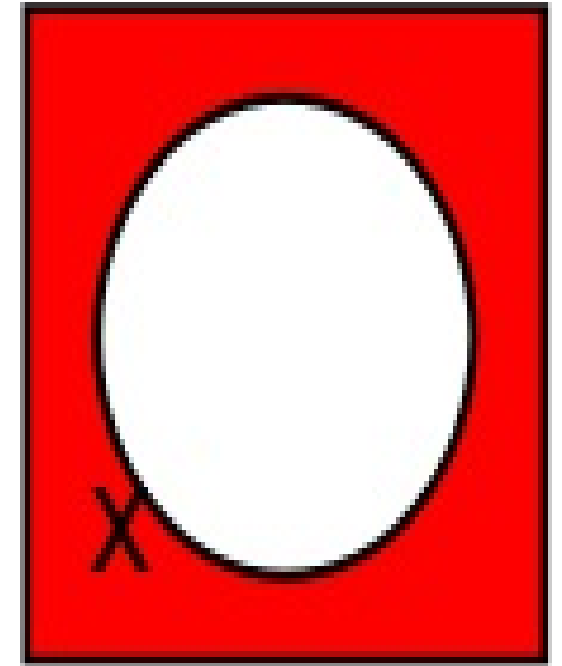
# A Relational Formalism



$$x \wedge y$$



$$x \vee y$$



$$\neg x$$

# What Is A Formalism?

- A formal system (or *formalism*):
  - **syntax**: what can we write?
  - **semantics**: what does our writing mean?
  - with precise (mathematical) definitions
  - designed to capture a coherent set of operations
  - ("syntax" is loose, e.g., we might just have a collection of operators)



# Key Goals Of A Formalism

1. to be **clear** about **what we mean**
  - In our spreadsheet is "1" a number, a string, either, both, something else?
2. to allow the determination of **key properties**
  - e.g., complexity of query answering
3. to **abstract** away from particular implementations
  - e.g., allow us to determine when wildly different implementations are *correct* thus can *interoperate*

# Formalism vs. Language

- Formalisms are often **abstract**
  - This can be an advantage!
  - Can be hard to use if **only** abstract
  - Concrete instances typically involve **compromise**
- We focus on concrete languages
  - Formalisms are the **theory**
  - Languages are the **practice**
  - **Other Quotes On Theory vs Practice**
    - Well, it may be all right in practice, but it will never work in theory.
    - In theory, there is no difference between theory and practice. But, in practice, there is.

# SQL: A Language For Tables

- Schema
  - **CREATE TABLE** *table\_name*
- Update
  - **INSERT INTO** *table\_name*
  - **DELETE FROM** *table\_name*
  - **UPDATE** *table\_name*
  - ...
- Query
  - **SELECT . . . FROM** *table\_name*

SQL operations (largely) are closed over tables

# An Infelicity

There is a lot of lingo with slight different meanings. Concepts get divided up in slightly different ways.

**Our talk**

**Common**

**Learning SQL p.10**

---

Core Data Model

---

Data Integrity

Data Definition

SQL schema statements "CREATE"

---

Data Manipulation

Query/Update  
Language

SQL Data statements

# A Sample SQL Program

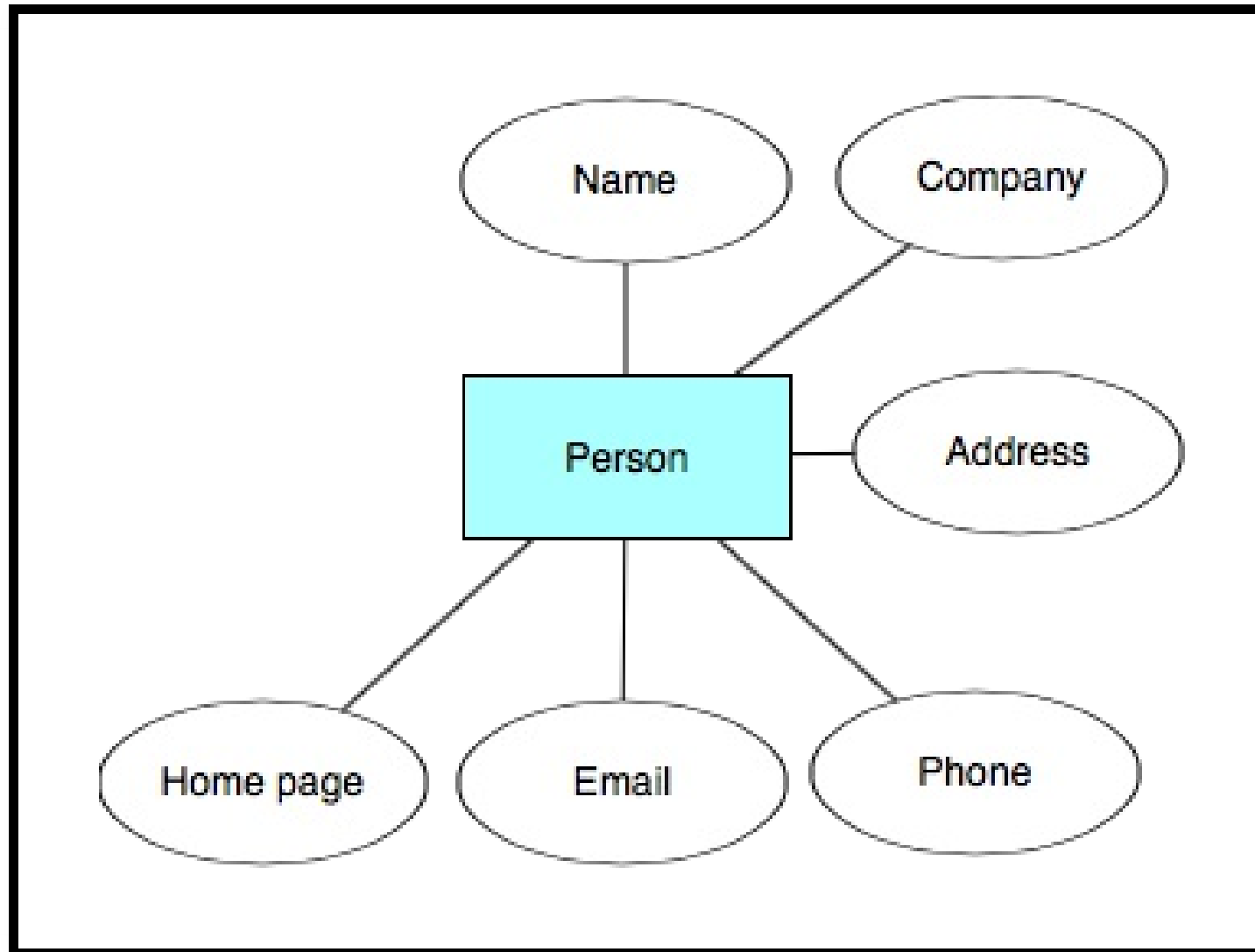
```
CREATE TABLE People (  
  name varchar(255),  
  company varchar(255),  
  address varchar(255),  
  phone varchar(255),  
  email varchar(255),  
  home_page varchar(255));  
  
INSERT INTO People  
  VALUES ('Aleshia Tomkiewicz', 'Alan D Rosenberg Cpa Pc',  
          '14 Taylor St, St. Stephens Ward, Kent CT2 7PP',  
          '01835-703597', 'atomkiewicz@hotmail.com',  
          'http://www.alandrosenburgcpapc.co.uk');  
  
SELECT name FROM People
```

- You must Define before Update before Query
  - I.e., **CREATE** before **INSERT** before **SELECT**

# Modelling With SQL

- SQL lets us express models at the **logical** to (some of the) **physical** level
  - Specifying indices is a bit physical
  - Knowledge about implementation may inform modelling choices
- SQL has no mechanisms for **conceptual** level

# Format 1 In SQL



# Format 1 In SQL

```
CREATE TABLE People (  
  name varchar(255),  
  company varchar(255),  
  address varchar(255),  
  phone varchar(255),  
  email varchar(255),  
  home_page varchar(255));
```

```
INSERT INTO People  
VALUES ('Aleshia Tomkiewicz', 'Alan D Rosenberg Cpa Pc',  
       '14 Taylor St, St. Stephens Ward, Kent CT2 7PP',  
       '01835-703597', 'atomkiewicz@hotmail.com',  
       'http://www.alandrosenburgcpapc.co.uk');
```

...

Can we do all that we did in the spreadsheet?



# SQL Manipulation of Format 1

- Count records in your People table:

```
SELECT COUNT(*) FROM People
```

- Search for items:

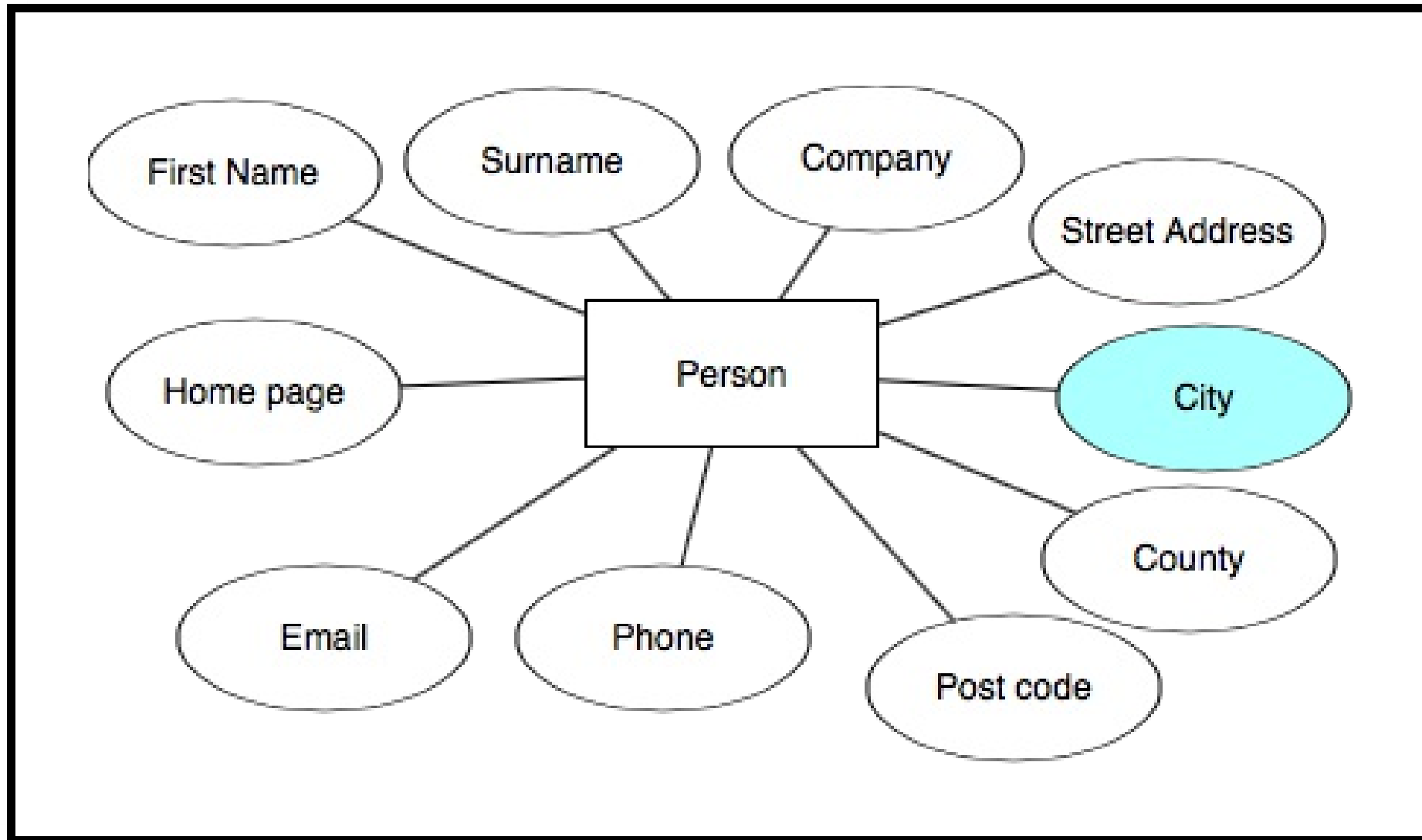
```
SELECT * FROM People  
WHERE name like 'Aleshia%'
```

```
SELECT * FROM People  
WHERE name like '%Tomkiewicz'
```

- Sort the table!

```
SELECT * FROM People  
ORDER BY name asc
```

# Format 2 In SQL



# Format 2 In SQL

```
CREATE TABLE People (  
  first_name varchar(255),  
  surname varchar(255),  
  company varchar(255),  
  street_address varchar(255),  
  city varchar(255),  
  county varchar(255),  
  post_code varchar(255),  
  phone varchar(255),  
  email varchar(255),  
  home_page varchar(255));  
  
INSERT INTO People  
  VALUES ('Aleshia', 'Tomkiewicz', 'Alan D Rosenberg Cpa Pc',  
          '14 Taylor St', 'St. Stephens Ward', 'Kent', 'CT2 7PP',  
          '01835-703597', 'atomkiewicz@hotmail.com',  
          'http://www.alandrosenburgcpapc.co.uk');
```

...

# SQL Manipulation of Format 2

- The old queries work, but we can improve them
  - Search for items:

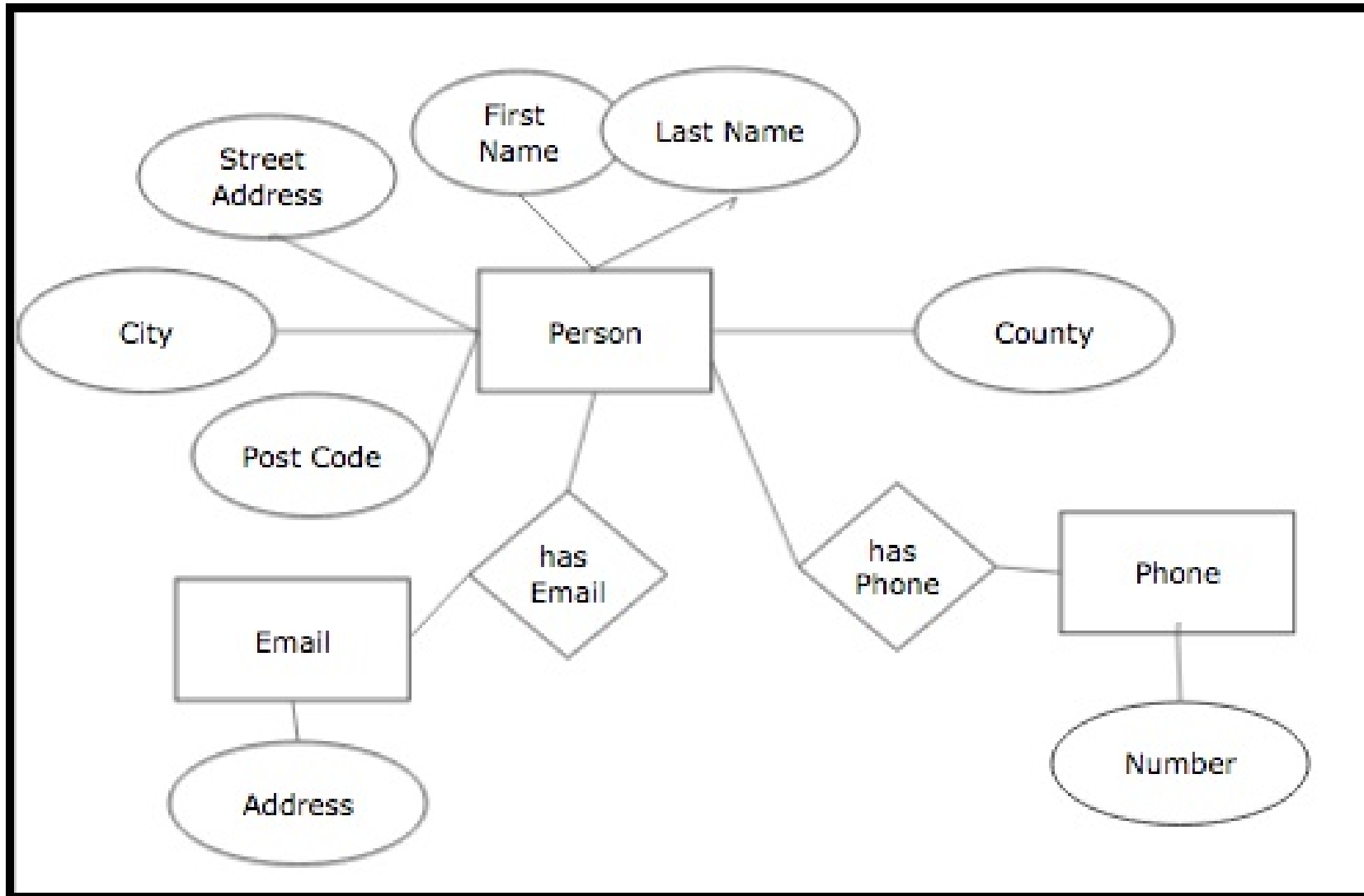
```
SELECT * FROM People
WHERE first_name = 'Aleshia'
```

```
SELECT * FROM People
WHERE surname = 'Tomkiewicz'
```

- We can recreate Format 1!

```
SELECT first_name || " " || surname as name,
street_address || ", " || city || ", " || county || " " || post_code as a
phone,
email,
home_page
FROM People
```

# Format 3 In SQL



# Format 3 In SQL

```
CREATE TABLE People (  
    person_id SMALLINT UNSIGNED,  
    first_name varchar(255),  
    surname varchar(255),  
    company varchar(255),  
    street_address varchar(255),  
    city varchar(255),  
    county varchar(255),  
    post_code varchar(255),  
    email varchar(255),  
    home_page varchar(255),  
    CONSTRAINT pk_person PRIMARY KEY (person_id));
```

```
CREATE TABLE Phone (  
    person_id varchar(255),  
    number varchar (255),  
    CONSTRAINT pk_phone_number PRIMARY KEY (number));
```

```
INSERT INTO People  
VALUES ('1', 'Aleshia', 'Tomkiewicz', 'Alan D Rosenberg Cpa Pc',  
    '14 Taylor St', 'St. Stephens Ward', 'Kent', 'CT2 7PP',  
    'atomkiewicz@hotmail.com',  
    'http://www.alandrosenburgcpa.co.uk');
```

# SQL Manipulation of Format 3

- Recreate Format 1 and Format 2: easy
- Find everyone with same phone number
- Can we have unassigned phone numbers?

# How did our formats do?

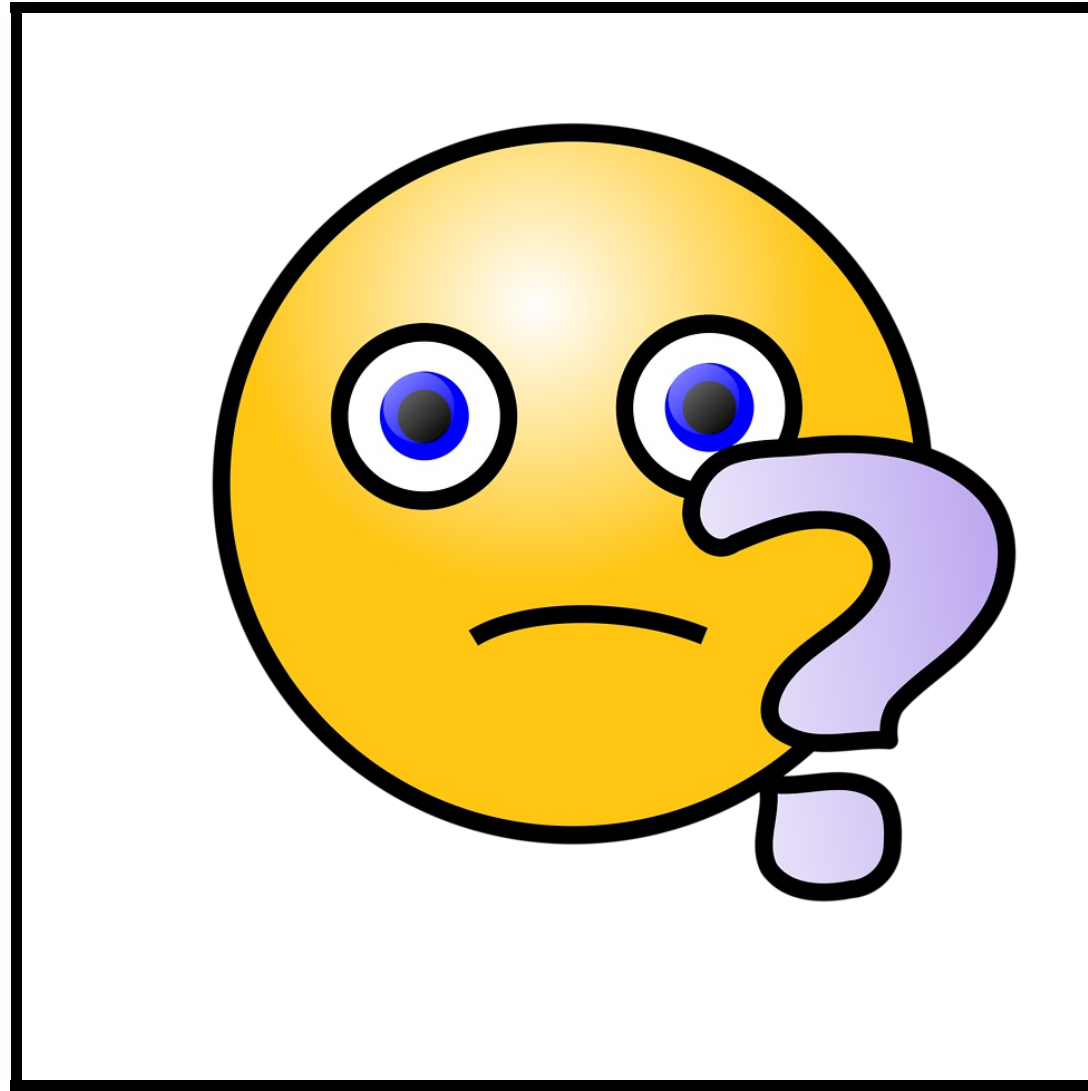
- Core DM/Data structure: Tables seem to work!
- SQL and Relational Model
  - We can do everything!
    - All queries in all models
    - Format 3 has 2 tables/requires joins
- Format 3
  - Neater inserting and deleting
    - Can have as many phones as you want!
  - Every other domain model can be derived
    - Just write the query!



# Expressive Power

- SQL is **expressive**
  - The core data model is rich
    - Composing and filtering tables does a lot!
    - Operators and functions helpful
      - Without concat(...), there'd be trouble!
  - The language is **powerful**
    - Reasonably **composable**
    - Lots of features
    - Extended & extensible in many implementations
      - Interop problems!

# Querying With SQL



# Schemas Vs. Queries

- **CREATE** statements
  - "create" *empty* tables
  - out of nothing at all
  - with certain constraints
  - with some expectation of permanence
- **SELECT** statements
  - "generate" *new* tables (possibly with data)
  - out of existing tables
  - according to some constraints
  - with no expectation of permanence

# Closed Over Tables

- SQL is (mostly) **closed** over tables
  - Most SQL constructs take & produce tables
  - Clear exception: Functions!
- Manipulation is manipulation of tables
  - Not rows, columns, or cells directly
  - Rows, columns, and cells are "degenerate tables"...

# Filtering

- Key operation **SELECT**: ignoring some parts
  - Basically "find"
  - Can filter rows or columns or both
  - Requires "testing" functions on values

# Filtering Columns

- aka "Projection", specified in SELECT clause
  - Keep all columns:

```
SELECT * FROM People
```

- Just a single column:

```
SELECT county FROM People
```

- Multiple columns:

```
SELECT name, county FROM People
```

- Rename columns:

```
SELECT street_address AS address FROM People
```

# Filtering rows

- Selecting specific tuples
- Specified in the WHERE clause of your query:
  - Equality:

```
SELECT * FROM People  
WHERE surname = "Smith"
```

- Range:

```
SELECT * FROM People  
WHERE heartrate > 95
```

- Compound criteria:

```
SELECT * FROM People  
WHERE heartrate > 95 AND county="Kent"
```

# Building Tables with Cross Join

- The fundamental operation is Cartesian product
  - $T_1 \times T_2$
  - for example *People x Phone*
- Makes a new row for **every** pair of rows from  $T_1$  &  $T_2$ 
  - What's the size of the result?
- Not really a user-oriented feature
  - "Incidentally" cross joins are dangerous!



# Building Tables With Inner Join

- An **inner join** is a join *filtered* on common columns
  - Useful for our phone records!

```
SELECT * FROM People, Phone
INNER JOIN ON People.person_id = Phone.person_id
```

- The above is special case, called "natural" join
  - can be written as follows:

```
SELECT * FROM People NATURAL JOIN Phone
```

# Building Tables with Outer Join

- An **outer join** is like an inner join but it returns also rows that do **not** have a match in the other table
  - *left outer* different from *right outer*

```
SELECT * FROM People, Phone  
RIGHT OUTER JOIN ON People.person_id = Phone.person_id
```

- will return also people who have no phone!

# Building And Filtering

- Once we've built a table we can filter things we need:

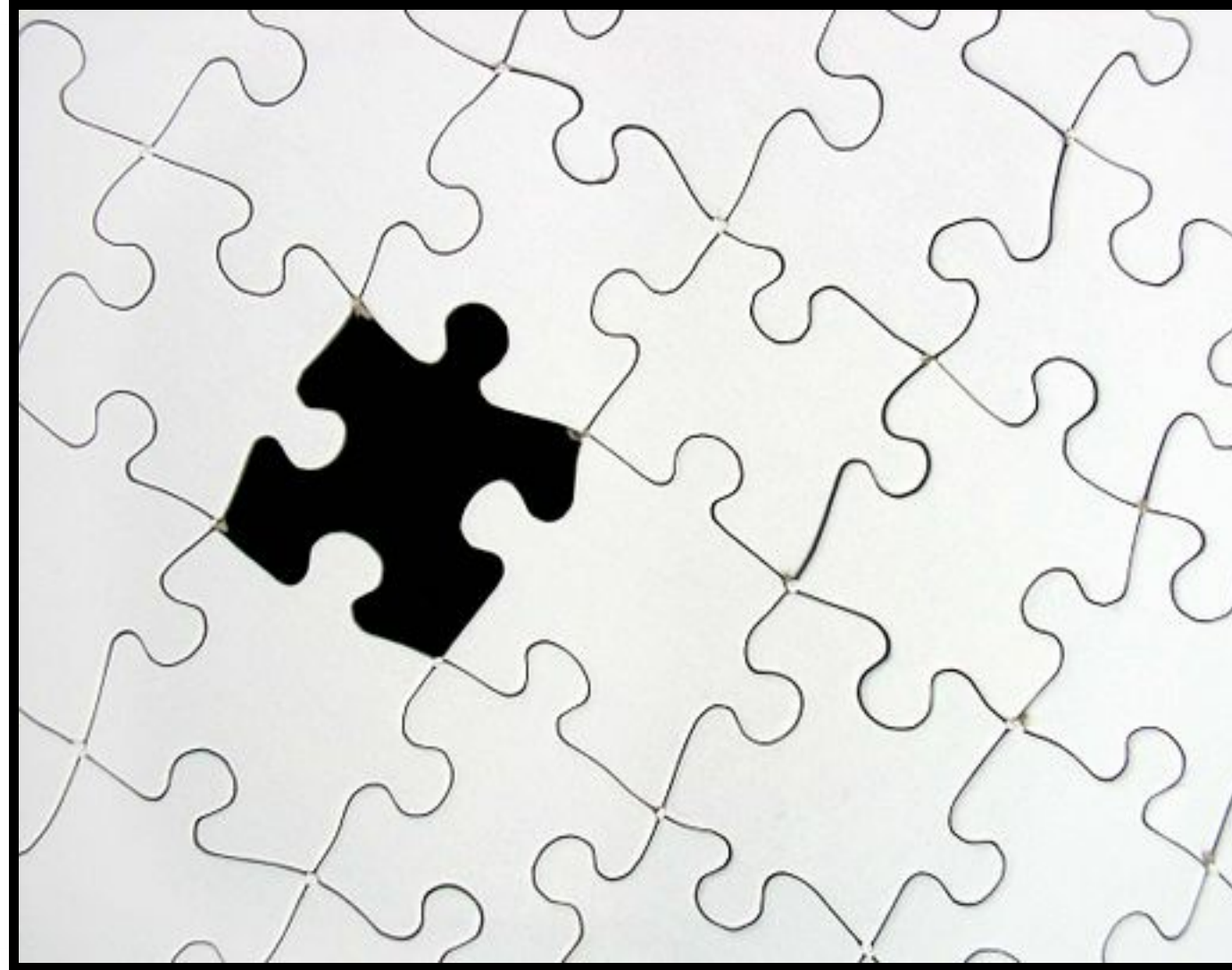
```
SELECT * FROM People, Phone  
RIGHT OUTER JOIN ON People.person_id = Phone.person_id  
WHERE People.surname = "Smith"
```

- ...you knew that already!?

# The Cost

- A **key issue** with joins
  - Worst case for their computation is a CROSS
  - Even if you don't **generate** the CROSS
    - You might have to **consider** all the pairs
    - (If you aren't careful)
- **Good optimisers** avoid both
  - Considering lots of matches (think indexes)
  - Generating large intermediate tables

# Incomplete Data



# Multiple Phone Columns

- Some people have **none or one**
- Or no **email** or **web page**

	A	B	C	D	E	F	G	H	I	J	K
1	first_name	last_name	company_name	address	city	county	postal	phone1	phone2	email	web
2	Aleshia	Tomkiewicz	Alan D Roser	14 Taylor St	St. Stephens	Kent	CT2 7PP	01835-703597	01944-36996	atomkiewicz@hotmail.com	
3	Evan	Zigomalas	Cap Gemini A	5 Binney St	Abbey Ward	Buckingham	HP11 2AX	01937-864715		evan.zigomalas@gmail.com	
4	France	Andrade	Elliott, John	8 Moor Place	East Southbc	Bournemouth	BH6 3BE	01347-368222	01935-82163	france.andrade	http://www.e
5	Ulysses	Mcwalters	Mcmahan, B	505 Exeter R	Hawerby cur	Lincolnshire	DN36 5RP	01912-771311		ulysses@hotmail	http://www.m
6	Tyisha	Veness	Champagne	5396 Forth S	Greets Greer	West Midlan	B70 9DT	01547-429341	01290-36724	tyisha.veness@hotmail.com	
7	Eric	Rampy	Thompson, M	9472 Lind St	Desborough	Northampton	NN14 2GH	01969-886290		erampy@ramp	http://www.th
8	Marg	Grasmick	Wrangle Hill	7457 Cowl St	Bargate War	Southampton	SO14 3TY	01865-582516		marg@hotmail.com	
9	Laquita	Hisaw	In Communic	20 Glouceste	Chirton Ward	Tyne & Wear	NE29 7AD	01746-394243			http://www.in
10	Lura	Manzella	Bizerba Usa I	929 Augustin	Staple Hill W	South Gloucc	BS16 4LL	01907-538509	01340-71395	lura@hotmail.com	
11	Yvette	Klapec	Max Video	45 Bradfield	Parwich	Derbyshire	DE6 1QN	01903-649460		yvette.klapec@	http://www.m
12	Fernanda	Writer	K & R Associ	620 Northarr	Wilmington	Kent	DA2 7PP	01630-202053		fernanda@writ	http://www.k
13	Charlesetta	Erm	Cain, John M	5 Hygeia St	Loundsley Gr	Derbyshire	S40 4LY	01276-816806	01517-624517		
14	Corrinne	Jaret	Sound Vision	2150 Morley	Dee Ward	Dumfries anc	DG8 7DE	01625-932209			http://www.sc
15	Niesha	Bruch	Rowley/hans	24 Bolton St	Broxburn, Up	West Lothiar	EH52 5TL	01874-856950	01342-79360	niesha.bruch@yahoo.com	
16	Rueben	Gastellum	Industrial Enj	4 Forrest St	Weston-Supe	North Somer	BS23 3HG	01976-755279		rueben_gastell	http://www.in
17	Michell	Throssell	Weiss Spirt &	89 Noon St	Carbrooke	Norfolk	IP25 6JQ	01967-580851		mthrossell@throssell.co.uk	
18	Edgar	Kanne	Crowan, Ken	99 Guthrie St	New Milton	Hampshire	BH25 5DF	01326-532337		edgar.kanne@yahoo.com	

# No Surname

- Even if we normalised that away
  - Some people don't have a surname!



# Null

- `null` is a distinguished value which can mean:
  - "Value not yet known"
  - "Not applicable to this entity"
  - "Value undefined"
  - check out [LSQL](#)
- Key property: Unequal to everything
  - `null = null` is **never** true
  - Match on not `null`, rather than `null`

Strange value!



# Outer Joins

- If you have no **nuLLs** in your base tables
  - you can't get them in tables derived by inner join
- However, the 2 phone column table **is** derivable
  - We use the **outer** join
  - Outer joins take a table T
    - for each row in T
      - extend it with the (projected) columns from another table
      - *If* there's a match, add the matched values
      - \*else, add **nuLLs**
- See Learning SQL **Chapter 10** for examples

# Null Proliferation

- **nuLL** never matches
  - So iterated outer joins proliferate **nuLL**s
    - As you get wider, you get sparser
    - If you are matching on a sparse attribute
- **nuLL**s pose challenge for relational theory
  - And somewhat for practice
  - Starts moving from the sweet spot

# SQL And The Web

A brief tour



# SQL Driven Websites

- Many websites are **backed by** a database
  - PHP makes it easy
  - Consider WordPress and other CMSs
- Lots of **unstructured** content
  - Stuff in blobs and text fields
- Key properties
  - Scaling
  - ACID: Atomicity, Consistency, Isolation, Durability
    - Transactions
  - Concurrent access

There is a **key historical text** that is still good reading,  
esp chps **11-12**



# CSV & SQL programs on the Web

- [UN Data repository](#)
- Other government repositories:
  - [data.gov](#)
  - [data.gov.uk](#)
- Scientific sites
  - [ClinicalTrials.gov](#) all about clinical trials!
  - [UniProt](#) all about proteins!
  - ...

# Google Query Viz Language

- A SQL like language
  - Used in Google Docs Spreadsheet
  - QUERY function takes queries as argument

# WebSQL

The WhatWG and W3C tried to standardize **WebSQL**

*This specification introduces a set of APIs to manipulate client-side databases using SQL.*

```
function prepareDatabase(ready, error) {  
  return openDatabase('documents', '1.0', 'Offline document storage', 5*1024*1024, function (db,  
    db.changeVersion('', '1.0', function (t) {  
      t.executeSql('CREATE TABLE docids (id, name)');  
    }, error);  
  });  
}
```

*Local database backed web apps*

- *For offline use*
- *Just increased capabilities*

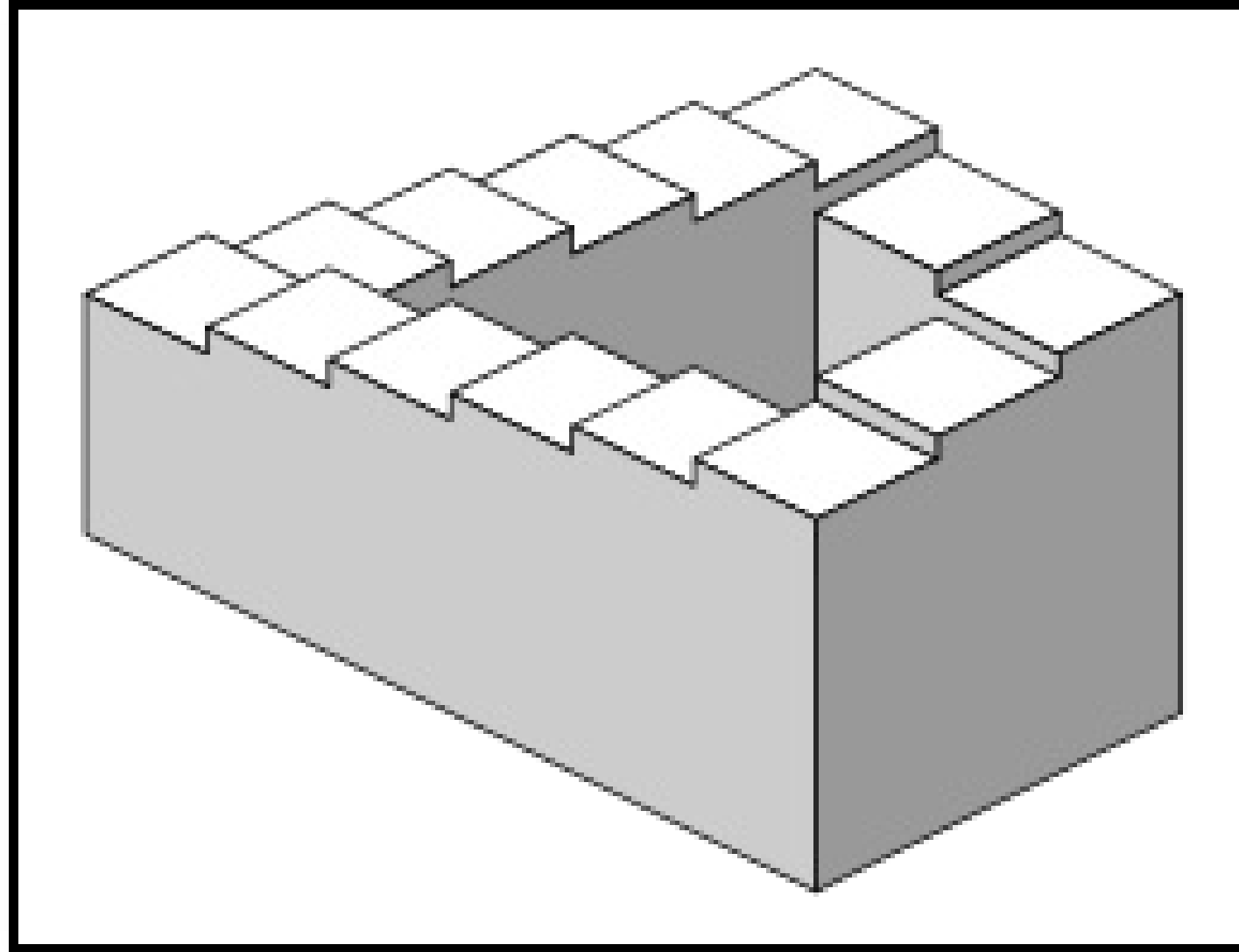




# What is this data?

- A recurring issue: what **is** in this shared document?
  - csv
  - table
  - JSON snippet
  - ...
- What does it mean?
- How to parse?
- How to share? So that it's good to use?
- **Self-Describing** and **Meaning** will be discussed at length

# Next Steps



# Reading

There is a **key historical text** that is still good reading,  
esp chps **11-12**

Any Questions So Far?

# Labs & Coursework

- Next, we go to the Labs
- You look in BB at Week 1 coursework:
  - Quiz Q1
  - Short Essay SE1
  - Small Modelling exercise M1
  - Some querying CW1
- Read, think, ask us!