CENTER FOR TEXTUAL STUDIES AND
DIGITAL HUMANITIES

DIGH 402 - Introduction to Digital Humanities Design and Programming

Spring Semester 2014

Week 1

Course Schedule

Wednesday 2.30pm - 5pm

14 classes

No class on Wednesday 5th March 2014

Final taught class is scheduled for Wednesday 23rd April 2014

Final Assessment due on Monday 28th April 2014 by 9.30pm

Goals of the course - Part 1

- best practices in educational software...
- explore existing platforms for e-learning
- learn to effectively communicate your content with words and visuals
- multimedia usage for learning purposes and effective balance of media
- design examples and practice exercises for online learning
- test collaborative learning and networked skills
- evaluate simulations and games for instructional usage

.....

Goals of the course - Part 2

Plus

- practical development of online educational resource
- learn use and implementation of LAMP (XAMPP/WAMP) stack
- XHTML, HTML5 development

If time

- modify online resource to mobile framework and publication (HTML5…)
    - responsive web design

Course Assessment and Evaluation

Ongoing assessment

- design project (50%)
- ongoing weekly assessment/reading (25%)
- class discussions (15%)
- class presentation/demonstration (10%)

Course Assessment and Evaluation

Ongoing assessment - design project (50%)

- development, publication, and demonstration of an online educational resource

Basic

- online educational resource for any age or target audience…
- free choice of content and material
    - organisation and logic of content will also be assessed (eg: flow, presentation, taxonomy etc…)

Additional

- coding is important ie: it needs to at least work!
- documentation of code
- repository publication, developer and user instructions…
- user testing (unit and UI testing required)

...

Course Assessment and Evaluation

Ongoing assessment - ongoing weekly assessment/reading (25%)

- class questions and tasks
- occasional weekly exercises
- class contributions
- course understanding and class performance
- weekly reading material to help inform design and e-Learning concepts, best practices, research, and testing

and

- ongoing development of final project code and modules

Course Assessment and Evaluation

Ongoing assessment - class discussions (15%)

- weekly discussion topic
- hosted on Google Groups
- moderated by myself
- private group for class members and CTSDH faculty only
- weekly discussion topic posted each Thursday after Wednesday's class
- topic available until following Wednesday

Course Assessment and Evaluation

Ongoing assessment - class presentation/demonstration (10%)

- scheduled for the end of the semester
- practical demonstration of online educational resource
- teach the class

...

Course Assessment and Evaluation

Course Website will be available by next week's class

Online Educational Resource

- LAMP (XAMPP/WAMP) stack
    - Linux, Apache, MySQL, PHP
    - use Raspberry Pi, XAMPP/WAMP, or LAMP (CTSDH Linux laptops etc…)
- PHP, XHTML, CSS, Javascript, MySQL
- documentation
- testing
- content structure
- online publication
- repository publication

...

Online Educational Resource

Already

- PHP, HTML, CSS basics
- XML and TEI
- semantic (taxonomy etc) organisation
- metadata
- digitisation practices

and now

- Databases (and then SQL in particular…)

Databases - Intro

- storing organised information, and subsequently knowing how to retrieve it again
- managing databases can get complicated very quickly
- data is often not only critical but data retrieval may also be time-sensitive
- database management involves designing and programming ways to store and retrieve data

A few basics

- a big bucket to dump information
- storing information and retrieving information
- three main types of database designs
    - free-form
    - flat-file
    - relational

Databases - free-form

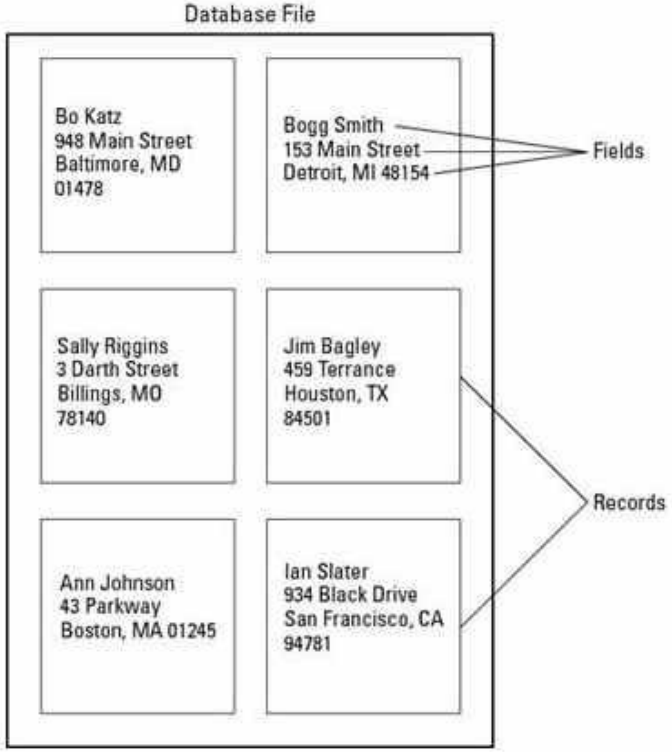
- designed to make it easy to store and retrieve information, albeit limited
- acts like a notepad or post-it board
- freedom to store dissimilar information in one place
- finding what you want can be problematic
- to find information you need to know at least part of the data you want to find

Disadvantages include:

- they're clumsy for retrieving information
- they can't sort or filter information

Databases - flat-file

- biggest difference between free-form and flat-file is structure in flat-file
- flat-file forces you to add information by first defining the structure of your data
- then you can add the data itself
- each chunk of data, eg: a first name, is known as a 'field'
- a group of fields is a 'record'

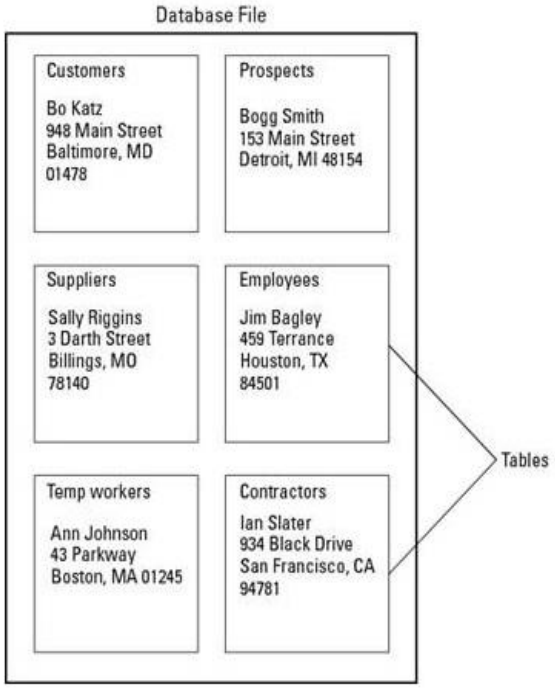


Databases - flat-file

- impose structure on the type of information you can store to make retrieval easier
- you need to design the structure of your database carefully
- size relative to field is also important for the type of information you can store
- information larger than storage size, eg: 10 characters, will simply be cut off
- definition of fields is also particularly important
- separation of fields is often useful to enable better sort and search options
- makes flat-file databases easier to search and sort information

Databases - relational

- suitable if you need to store large amounts of data
- majority of current database programs are relational
- you must define number and size of fields relative to type of information required
- unlike flat-file databases, relational databases can further organise data into tables
- organisation of information as tables with further sub-division in fields



Databases - relational

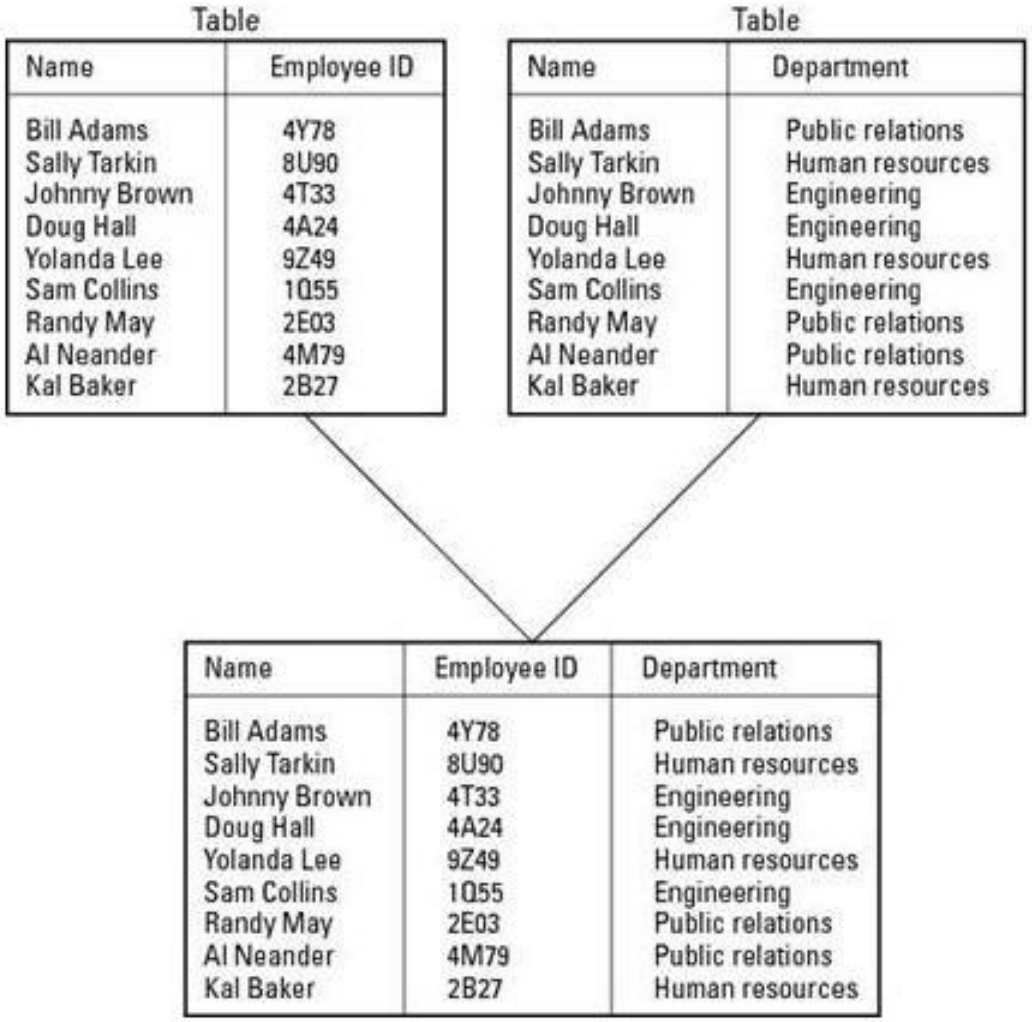
- dividing data into tables with a table grouping the minimum amount of data
- column in a table represents a single field or 'attribute'
- row in a table represents a single record or 'tuple'
- tables can be linked together
- link between separate tables automatically keeps that information updated and accurate in all other linked tables
- by linking or relating tables together you can combine data in different ways
- relating tables together allows you to create 'virtual' databases
- Primary/Unique key and Foreign key

Databases - relational

- create tables to contain required data
    - eg: content, content type, taxonomy, project, user....
- create lookup tables to cross-reference tables
- query lookup table to get information from multiple tables
- sort and organise lookup table to get different query results

and on, and on...

Databases - relational



Table

Name	Employee ID
Bill Adams	4Y78
Sally Tarkin	8U90
Johnny Brown	4T33
Doug Hall	4A24
Yolanda Lee	9Z49
Sam Collins	1Q55
Randy May	2E03
Al Neander	4M79
Kal Baker	2B27

Table

Name	Department
Bill Adams	Public relations
Sally Tarkin	Human resources
Johnny Brown	Engineering
Doug Hall	Engineering
Yolanda Lee	Human resources
Sam Collins	Engineering
Randy May	Public relations
Al Neander	Public relations
Kal Baker	Human resources

Name	Employee ID	Department
Bill Adams	4Y78	Public relations
Sally Tarkin	8U90	Human resources
Johnny Brown	4T33	Engineering
Doug Hall	4A24	Engineering
Yolanda Lee	9Z49	Human resources
Sam Collins	1Q55	Engineering
Randy May	2E03	Public relations
Al Neander	4M79	Public relations
Kal Baker	2B27	Human resources

Databases - manipulating data

- write commands for modifying and manipulating the information
- three basic commands for manipulating data
    - select
    - project
    - join
- select command retrieves a single row or multiple rows from a table
- project command retrieves the entire column from a table
- project acts like a filter
- join command combines separate tables together to create a virtual table