ITMD 523 *Advanced Topics in Data Management* HW 1

Student Name **Henry Post** Section **422**

Instructor ***Luke Papademas*** Due Date **Sep 07**

Part	1	2	3	4	**TOTAL**	**Score**
Maximum Points	25 points	25 points	25 points	25 points	**100** points	

Textbook Reading Assignment Thoroughly read chapter(s) 1 - 3 in the textbook as well as the course lecture notes for Week(s) 1 - 3 .

Part 1 Concepts, Topics, Glossary Terms - Database Systems

Comment and expound, in detail, on each of the following questions. Use examples to support your comparisons and indicate when and / or where the individual concepts would apply.

(1) (Data, Information, Knowledge, Wisdom)

Data, information, knowledge and wisdom often provide for an important distinction. Provide five examples of categories of data that may be collected from a college bookstore chain. Provide five examples of information that could be gathered from your data.

What if we reverse the roles of data and information? Can information lead to data? Can knowledge lead to information? How does wisdom come into play? Explain your answers!

Data Points:

1. The most-bought books and who buys them (students or kids or parents?)
2. What times of the year different genres get bought (think back-to-school v.s. vacation times)
3. Least-bought books (too expensive? Boring? Should we stock them at all?)
4. Prices of books over time. (Can we make more money by slightly raising prices of books? Will that deter people?)
5. Most-stolen books (are they related to any other data points?)

Information Points:

1. The opinions of customers on their favorite books
2. The employees' favorite times to work, and why
3. The opinions of college students on book prices (read: bad!)
4. The customers' favorite places to read books
5. The customers' suggestions of how to improve the bookstore.

Data is numbers with context. Data can be useful depending on specific contexts, or useless in different contexts. Data could be times of the day, frequencies, etc. Without some related attribute to link the data (what time you get up), data without context is quite useless because you do not know how it was recorded or what it represents.

ITMD 523 *Advanced Topics in Data Management* HW 1

Student Name **Henry Post** Section **422**

Knowledge can be described as facts that one knows. You cannot store knowledge as data points in a traditional sense. If I know that “green light means go when crossing”, I need to have an understanding of “green”, what “go” means, and that I am talking about traffic etiquette. This is somewhat similar to qualitative data but it requires a deeper understanding of context.

Information is similar to knowledge but it is less of a ‘fact’ and more of an observation about a particular person, entity, datum, or process. It could be described as qualitative data. For example, “cheese tastes good” is information. That statement cannot be encoded as a number without some intermediate transformation of ‘good’ and ‘cheese’ into numbers.

Wisdom is experience. If you have been a database administrator for many, many years, you will be disposed to creating schemas in a particular way or normalizing your tables fully before you create them. You can apply knowledge (facts), and store data, but having both of those will not give you practical experience that wisdom provides.

(2) (Structured Data versus Unstructured Data)

When we examine data received into an enterprise we often separate the data into two categories - structured data and unstructured data.

For example, in a customer service application we can differentiate incoming data as being structured (customer feedback and time to resolve customer inquiries) or unstructured (images and illustrations) .

Provide some examples of transforming unstructured data into structured data.

Can structured data be transformed into unstructured data? Explain your answer!

You can transform unstructured data into structured data if you can map a quantitative field into a qualitative field. For example, if you wish to turn drawn shapes (images) into a list of enumerated shapes (SQUARE, CIRCLE, OVAL), you can do that with computer vision or a human operator.

However, information must be lost for this to happen. The aberrations that each hand-drawn shape had will be ‘flattened’ and only its general impression (“SQUARE”) will remain.

Alternatively, text can be an example of unstructured data. Customer feedback is a great example of this. Without a quantitative “good/bad” option, a human operator may be required to discern whether or not a customer has had a positive experience. Again, this transformation of qualitative data into quantitative data will cause information loss.

ITMD 523 *Advanced Topics in Data Management* HW 1

Student Name **Henry Post** Section **422**

Structured data can also be turned into unstructured data. Certain data (such as a list of X,Y coordinates) depends on other fields. If a table has a list of gas stations, if you separate the two fields, they are much less useful.

(3) (Structural Dependence versus Structural Independence)

Another important distinction in the realm of database management is that between structural dependence and structural independence.

Structural dependence arises when a change in the structure of a file requires the modification of all programs that refer to that file. Conversely, structural independence occurs when changes to the file structure can be performed without affecting the application program's ability to access the data.

From a management perspective, comment on how managers should understand the distinction between these two structural categories. Provide examples to support your claims.

ITMD 523 *Advanced Topics in Data Management* HW 1

Student Name **Henry Post** Section **422**

Managers can understand structural dependence through simple examples. They have likely worked with word/sheet processing tools before, so likening this to updating Excel spreadsheets may help them.

Suppose there is a table that contains records of “Tax professionals” and “Clients”. This table will link one tax professional to one or more clients. Let’s call it “TaxProClients”.

TaxProClients COMPOSITE KEY (TaxProID, ClientID)	
TaxProID (FK int)	ClientID (FK int)
1	1
1	2
2	3

The client’s name, address, phone number, etc. will NOT be stored in this “TaxProClients” table for reasons that will be explained.

Now, suppose a tax professional changes their home address. The “TaxPros” table will look like this:

TaxPros PRIMARY KEY (TaxProID)		
TaxProID (PK int)	Name (string)	Address (string)
1	Mary Jane	123 Example Place
2	Jack Johnson	345 Cool Ave

If we had stored the tax pro’s address as a column in the “TaxProClients” table, we would have had to update Mary Jane’s address twice. It is better to store information in a way that de-duplicates it than it is to duplicate it. It also makes for simpler, shorter tables that read more logically.

ITMD 523 *Advanced Topics in Data Management* HW 1

Student Name **Henry Post** Section **422**

(4) (Data Consistency versus Data Inconsistency)

Examine Web sites that offer data for financial markets, stocks and bonds, etc., such as: **https://finance.yahoo.com**

What are some issues of data inconsistency that can occur when examining stock data for your favorite publicly traded companies?

Data inconsistency could be when duplicated data (either from your own database or two different sources) is different. It could be difference in real value, difference in format, length, etc.

All of these things can be issues if you are trying to make assumptions based off of inconsistent data or if you are trying to determine which piece of data to use.

If, in the case of stocks, you make assumptions based off of the apparent consistency (22.59 dollars versus 22.74 dollars), you can lose large amounts of money depending on the difference in data.

You also must consider the source and quality of the financial data, because that may affect the consistency of the data that you are receiving. If you are doing high-speed trading, some people prefer to be right next to the New York stock exchange.

(5) (Database Design versus Database Structure)

Consider designing an employee feedback application for a corporate firm that wishes to train their employees on a new system and then assess their knowledge of the subject. All employees will be required to take the employee feedback application training and afterwards take the application quiz.

Referring to the firm's attempts to design their quiz application, comment on some issues that could occur between the back - end database development team and the front - end design team.

One way to determine how to model your data is to first identify what needs to be recorded either by creating 'objects' that encapsulate the information or just by writing it down.

The data being stored will be:

- Training data (the content)
  - Text
  - Videos
- Quiz questions and answers
- Employee scores

ITMD 523 *Advanced Topics in Data Management* HW 1

Student Name **Henry Post** Section **422**

Some issues between the database and front-end teams could be:

- How do we store media like images or videos?
- How do we create a way to create training modules from a series of separate quiz questions?
- How do we store user responses and evaluate their responses in a pass-fail manner?

Part 2 DBMS Concepts - Database Systems

(1) Although the database system yields considerable advantages over previous data management approaches, database systems do impose significant costs.

What are the potential costs of implementing a database system?

- Security
  - How do you store private information?
  - How securely do you store passwords (For the love of god, hash and salt them!)
  - What is the potential cost of a database breach?
- Infrastructure
  - Self-hosted
    - How many databases do you need?
    - Is there replication?
  - Cloud
    - How much does it cost per month?
    - Is it scalable?
- Maintenance
  - How many database/system admins do you need?
  - Scalability of having many database admins?
  - How many 'fires' need to be put out regularly? Why?

(2) List the five types of users identified in a database system and describe their individual interactions with database systems. Use the course textbook(s) to support your descriptions of the types of users.

1. End users
   - These are the people who use your website. They generally do not have any technical information about your backend. However, attackers may be able to gain a good amount of knowledge about your system, esp. insider threats.
2. Programmers
   - These people will have knowledge about your database system. They will create code that creates, reads from, updates, and deletes records from the system.

Student Name **Henry Post** Section **422**

3. System administrators
    - These people will be the ones who may set up, debug, and maintain the system that runs the service and database throughout its life. They will have a good knowledge about how the database runs and its performance metrics, but not necessarily about the database schema.
4. Database administrators
    - These people will set up, debug, and maintain the database specifically. They will have a good knowledge of permissions, security, and performance with regards to the database.
5. Database designers
    - These people are responsible for planning the schema of the database before it is created, and sometimes long before any servers are actually created.

Part 3 Data Modeling Concepts - Advanced Topics in Data Management

(1) Consider these types of local businesses, and their individual system needs, that could be near or in your area of residence.

- Inez's Imports: inventory management system
- Madison Marketing and Merchandise: sales ordering system
- Vermont Venues: concert ticket reservation system

Choose one of these business and comment on how that particular business would benefit by using a database system for their particular needs? What data tables do you think would be important for their database? What analytics can be performed with the database?

ITMD 523 *Advanced Topics in Data Management* HW 1

Student Name **Henry Post** Section **422**

- Inez's Imports: Inventory mgmt system
  - Benefits:
    - Can see stock at a glance
    - Can run analysis on most profitable stock
    - Can streamline stock-keeping into digital records, avoid slow manual stock-keeping.
    - Can create detailed per-day and per-month breakdowns of profits and losses
    - Can determine which vendors of goods to keep versus drop based on profit margins

  - Tables:
    - **Item**: *A single stock item from one or many different vendors.*
      - **ItemID** int PK
      - ItemPrice int FK

    - **ItemPrice**: *The price of a single item from a specific vendor at a specific date.*
      - **ItemPriceID** int PK
      - **VendorID** int FK
      - **ItemID** int FK
      - Price float
      - Date date

    - **ItemTransaction**: *A transaction of one or more items.*

      This table is used to compute the current stock of an item instead of manipulating a field as it can be rolled back.
      - **ItemTransactionID** int PK
      - **ItemID** int FK
      - TransactionDate date
      - ItemAmount int
      - **BuyerPersonID** int FK
      - **SellerPersonID** int FK

    - **VendorPerson** (COMPOSITE {**VendorID,PersonID**}): *A person who represents a vendor.*
      - **VendorID** int FK
      - **PersonID** int FK
      - Role string

    - **Vendor**: *A vendor that sells items.*
      - **VendorID** int PK
      - VendorName string

    - **Person**: *A person. Does not need to be an employee or*

Student Name **Henry Post** Section **422**

customer. Intentionally generic.

- **PersonID** int PK
- Firstname string
- Middlename string
- Lastname string
- DOB date

This current schema allows you to perform quite a few useful queries with low extra processing because of how it was designed. You can:

- Compare the efficiency of vendors based on profit margins
- See a per-person breakdown of purchases, money made, money lost
- See per-day, per-month, etc. breakdown of profits versus costs
- See who stole those 20 biscuits because of a detailed transaction table
- See what the most popular items are (also per-day and per-month!)
- See if a provider is changing their prices!

(2) The general tasks that are required to form an RDBMS solution include:

- preliminary consultancy
- requirements analysis
- system specification
- database design
- programming procedures
- testing
- implementation
- training
- continual maintenance

Consider being given the task of designing and implementing a database system that would track business credit memos, which are issued by the seller of goods or services to the buyer, reducing the amount that the buyer owes to the seller due perhaps to a change of terms for a prior invoice. A sample credit memo follows.

Choose one of the above tasks and comment on what occurs during the achievement of the task as far as the contribution of the task to the completion of the database project.

ITMD 523 *Advanced Topics in Data Management* HW 1

Student Name **Henry Post** Section **422**

Credit Memo ***Elston Environmental Services*** **(EES)**

JOB **[Job description]** **TO**

Customer ID [ABC12345]
DATE:
CREDIT NO. [100]
[NAME]
[COMPANY NAME]
[STREET ADDRESS]
[CITY, ST ZIP CODE]
[PHONE]

Quantity	Item #	Description	Unit Price	Line Total
			Subtotal	
			Sales Tax	
			Total	

Testing!

Testing is critical to confirming that:

- The schema works
- The schema can handle different data
- The database can handle a simulated load
- The code changes are able to be tested

Testing is something that will need to occur throughout the life of your application, and should occur early on in your application. Testing can validate code and produce production-like loads that ensure your application can perform well.

ITMD 523 *Advanced Topics in Data Management* HW 1

Student Name **Henry Post** Section **422**

Part 4 Data Design Concepts - Database Systems

(1) (File Structures)

Consider the File Structure for the XYZ Company Project Management data given below. Then, respond to each of the following questions related to the structure.

ProjectCode	ProjectManager	MgrTelephone	MgrLocation	ProjBidPrice
31-205A	Sami A. Allen	773-555-1216	2900 S. Federal St. Chicago, IL 60616	$46,000
37-403B	Cecily D. Worth	773-555-1217	27 Beckley Rd. Battle Creek, MI 49015	$1,342,000
33-906T	Daisy B. Burns	773-555-8821	2543 W. Foster Ave. Chicago, IL 60625	$847,320
29-107D	Alice M. Zane	773-555-1219	7202 Harrison Ave. Rockford, IL 61112	$1,449,000
21-929A	Dean P. Pence	773-555-2222	6302 N. Northwest Hwy. Chicago, IL 60631	$903,117
41-386C	Cecily D. Worth	773-555-1217	27 Beckley Rd. Battle Creek, MI 49015	$1,805,000
26-903C	Sami A. Allen	773-555-1216	123 Lane St. Chicago, IL 60616	$78,081
29-227A	Cecily D. Worth	773-555-1217	27 Beckley Rd. Battle Creek, MI 49015	$2,550,273

(a) If the XYZ Company wishes to display a listing of the Project Codes alphabetically by the right - most character, what problem(s), if any, would you encounter?

Would it make any sense to solve this problem by altering the file structure?

There is a lack of ***information*** about how data is being encoded in the ProjectCode column.

It is unclear if the first two characters (DIGIT DIGIT) are a separate field, and also unclear what the last character (CHARACTER) represents.

If this field does not fully encapsulate a self-contained piece of information, then splitting it will lose information.

If a field contains multiple pieces of information, then a transformation must be applied to extract a single piece of information. This is generally awkward to do in SQL and may depend on unsafe factors to rely on, such as the length of the string.

If, for some reason, another field is later appended to this string, and we must extract the last character, then it no longer refers to the field that we

Student Name **Henry Post** Section **422**

used to extract and we must modify our code.

We should normalize our tables as much as possible, and this includes splitting up compound fields into atomic, well-defined fields.

(b) What problem would you encounter if you desire to produce a listing by state? How would you solve this problem by altering the file structure?

There is no singular field for listing by state, and no well-defined structure for the state field. This means that there is no standard way to extract the state from a compound field.

Again, we must normalize our table by splitting up data as much as possible.

(c) If you wanted to produce a listing of the file contents by last name, area code, city, state or zip code, how would you alter the file structure?

The first and last name must be split up.

The area code, city, state, and zip codes also must all receive their own columns.

Ideally, there should be a “Person” table that stores a person’s name and “PersonID”.

There should be an “Address” table that stores addresses, as atomic as possible.

There should be an “PersonAddress” table that only stores a composite key of a PersonID and an AddressID to allow a person to have multiple addresses.

(d) What data redundancies do you detect? How could those redundancies lead to anomalies?

- Project Code may have multiple IDs in it. This may make it hard to save/retrieve specific IDs from it.
- Project Manager’s name, telephone number, and address should be stored elsewhere because they are not inherently related to a Project. They can be put into a “Person” table.
- The project manager’s name should be split into a first name, last

name, and middle name(s).

(e) The ProjBidPrice column, in the given table, appears to have a wide range of values. An XYZ Company database clerk suggests splitting the Project Management data into two file structures, as this may make it easier to search for values in this column. What issues, if any, could be resolved by proceeding with the clerk's suggestions? What issues, if any, could be resolved by proceeding with the clerk's suggestions?

The ProjBidPrice column could be split into a log of prices in the past. This should happen because it might be useful to record past prices and compare them.

This fixes issues that have to do with multiple managers setting different prices that would result in data duplication under this current schema.

If we wanted to look up past prices or allow multiple managers to bid on projects, we must implement a different structure.

However, this would cause queries to require more JOIN statements and become more complicated.

(2) (Database File Structures: Data Redundancy)

Consider the File Structure for the ABC Company Project Management data given below. Then, respond to each of the following questions related to the structure.

ProjectNum	ProjectName	EmpNum	EmpName	JobCode	Job_Chg_Hour	Proj_Hours	Emp_Phone
1001	Thunder	121	Daisy B. Burns	AB	80.00	14.2	773-555-1216
1001	Thunder	217	Alice M. Zane	CD	65.00	15.7	773-555-1216
1002	Chicago	821	Dean P. Pence	BD	95.00	14.4	773-555-1231
1002	Chicago	219	Cecily D. Worth	CD	80.00	17.2	773-555-1217
1003	Archimedes	222	Sami A. Allen	EH	65.00	24.9	773-555-8821
1003	Archimedes	121	Cecily D. Worth	AB	95.00	37.8	773-555-1219
1003	Archimedes	516	Denny T. Li	HW	80.00	22.2	773-555-2222
1004	Emerald	355	Danny T. Li	UG	65.00	19.7	773-555-

ITMD 523 *Advanced Topics in Data Management* HW 1

Student Name **Henry Post** Section **422**

							2222
1005	Diamond	217	Alice M. Zane	CD	80.00	19.5	773-555-1216
1005	Diamond	222	Sami A. Allen	BD	95.00	22.3	773-555-8821

(a) List and discuss two additional fields that could or should appear on the above table structure.

Project manager: It is not clear who is managing projects in this table, and a project manager must be able to be contacted if a project has an issue that requires escalation.

Project cost table FK: A project generally has funds, personnel, and equipment that get allocated to it. To be able to properly manage a project requires knowledge of resources that it has access to.

(b) Identify and discuss the serious data redundancy problems exhibited by the file structure shown in the given data sheet.

There are multiple fields, specifically EmpName EmpNum, and Emp_Phone, that should be moved to an “Employee” table.

This would allow the “Project” table to only store information that only has to do with a project. It is true that an employee can work on a project, but that information should be stored in a table whose only purpose is to document that relationship, such as a “EmployeeProject” table.

After all of the EmpName and EmpNum, etc. fields are moved out of this table, the redundancy of having multiple ProjectName and ProjectNum fields can be eliminated.

The “EmployeeProject” table could then store the hourly rate the employee is getting paid to work on that project. This is better because that information (an employee’s rate for working on a project) is being stored in a table that is more specific to it.