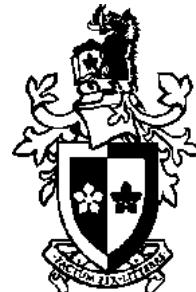




SWINBURNE
UNIVERSITY OF
TECHNOLOGY

COS20031
Computing Technology Design Project

Week 05
Refining the model





Database Development Lifecycle

1. Planning
2. Requirement gathering
3. Conceptual design

4. Logical design

5. Physical design
6. Construction
7. Implementation and rollout
8. Ongoing support



Outline

1. Normalising your data
2. Logical design considerations
3. Project update



(A)Normalisation



Normalising the data

- Identify key fields
- 1st normal form
- 2nd normal form
- 3rd normal form
- Denormalisation



Identify key fields

- Primary Key (PK) field: uniquely identify field values
 - thus also the rows (records) in the table
- Properties:
 - unique values
 - not null
 - specified when a row is created
 - values are very *rarely* changed
- PK Types:
 - natural key
 - surrogate (artificial) key

Invoice

Invoice Number	Customer Name	Date	Product Name	Quantity
121	Delish Food	3/12/2014	First Cold Press	2
122	Bread Express	3/12/2014	Extra Virgin	1
123	Bread Express	3/12/2014	First Cold Press	1



Identify key fields (2)

- Surrogate key:
 - no natural key exist
 - create a surrogate key, e.g. EmployeeID
 - values can be automatically generated number (e.g. auto-increment)
- Composite key:
 - e.g. Latitude & Longitude

Employees

First Name	Last Name	Phone Number	Address	...
Lilah	Douglas	(969) 116-3050	201 Buckaroo Road	...
Karyn	Reese	(517) 316-7182	548 Hominy Drive	...
Chester	Levine	(422) 736-7379	583 Roosevelt Lane	...

EmployeeID



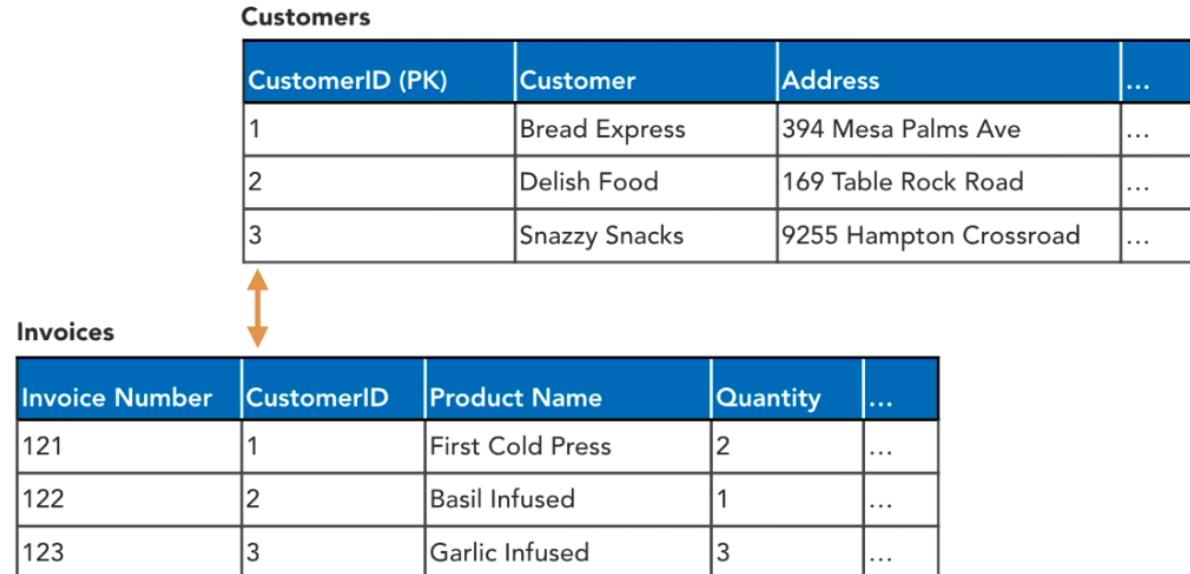
Wells

Latitude (PK +)	Longitude (PK +)	Depth
45	120	5.6
45	121	4.7
48	120	9.2



Foreign Key

- Foreign Key (FK): realises the relationship between two tables
 - references the PK of the related table
 - e.g. Invoice.CustomerID is FK that references Customers.CustomerID





Normalisation

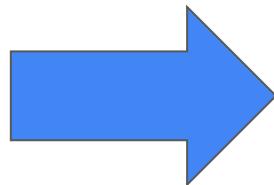
- **Normalisation:** a series of rules to ensure a quality relational data model
- Rules are called normal forms
- Must satisfy one form after another
 - like passing through a series of check-points



3 Normal forms (by Codd)

1NF

“the key”

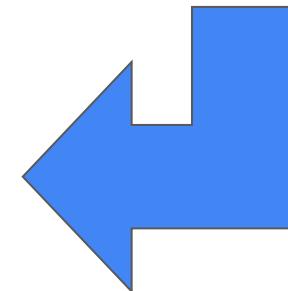


2NF

“the whole key”

3NF

“nothing but the key”





1st Normal Form (1NF): “the key”

- **1NF:** every non-key field must provide a fact about the key
- Break multivalued fields into separate fields or tables
 - use separate tables if the number of values is not predetermined
- Example: Tours.
 - How do we transform it into 1NF?

Tours

TourID	Tour Name	Tour Keywords	...
1	Big Sur Retreat	Hiking, National Parks, Big Sur	...
2	In the Steps of John Muir	Hiking, National Parks, Yosemite, John Muir, Camping	...
3	The Death Valley Survivor's Trek	Hiking, Desert, Mojave, Death Valley	...
4	The Mt. Whitney Climbers Tour	Hiking, Climbing, Backpacking, Mt. Whitney	...



Example: Tours (1)

- Is this ok?

Tours

TourID	Tour Name	Keyword1	Keyword2	Keyword3	Keyword4	Keyword 5
1	Big Sur Retreat	Hiking	National Park	Big Sur		
2	In the Steps of John Muir	Hiking	National Park	Yosemite	John Muir	Camping
3	The Death Valley Survivor's Trek	Hiking	Desert	Camping	Mojave	Death Valley
4	The Mt. Whitney Climbers Tour	Hiking	Climbing	Backpacking	Mt. Whitney	



Example: Tours (2)

- Solution for 1NF:

Tours

TourID	Tour Name
1	Big Sur Retreat
2	In the Steps of John Muir
3	The Death Valley Survivor's Trek
4	The Mt. Whitney Climbers Tour

TourKeywords

TourID	Keyword
1	Hiking
1	National Park
1	Big Sur
2	Hiking
2	National Park
2	Yosemite
2	John Muir
2	Camping
...	...



Example: Transactions (1)

- Transform this into 1NF:

Invoices

Invoice Number	Customer	Address	Product Name	Quantity
116	Bread Express	394 Mesa Palms Ave	First Cold Press	2
			Basil Infused	3
117	Snazzy Snacks	9255 Hampton Crossroad	Basil Infused	4
			Mission Oil	2
			Extra Virgin	3



Example: Transactions (2)

- Solution for 1NF:

Invoices

Invoice Number (PK)	Customer	Address
116	Bread Express	394 Mesa Palms Avenue
117	Snazzy Snacks	9255 Hampton Crossroad

Line Items

Invoice Number (PK+)	LineItem Number (PK+)	Product Name	Quantity
116	1	First Cold Press	2
116	2	Basil Infused	3
117	1	Basil Infused	4
117	2	Mission Oil	2
117	3	Extra Virgin	3



2nd Normal Form (2NF): “the whole key”

- **2NF:** non-key fields are determined by the whole PK (all the fields in the PK)
 - esp. if PK is a composite key => all component fields are required to determine other fields
- Move fields not uniquely identified by PK to another (referenced) table
- Example: LineItems: has a composite key (Invoice Number, Product Name)
 - is it in 2NF?

LineItems

Invoice Number (PK+)	Product Name (PK+)	Quantity	Manufacturer	...
116	First Cold Press	2	Two Trees	...
116	Basil Infused	3	Mary's Dipping Oils	...
117	Basil Infused	4	Mary's Dipping Oils	...
117	Mission Oil	2	Two Trees	...
117	Extra Virgin	3	Two Trees	...



Example: LineItems (1)

- Answer: No
 - CK uniquely identifies Quantity but not Manufacturer (Why?)
- How do we transform it to 2NF?
 - move Manufacturer to another table (named Products)
 - copy ProductName to this table and make it PK
 - make LineItems.ProductName reference Products.ProductName

LineItems

Example: LineItems (2)

- Solution for 2NF:

Invoice Number (PK+)	Product Name (PK+)	Quantity	...
116	First Cold Press	2	...
116	Basil Infused	3	...
117	Basil Infused	4	...
117	Mission Oil	2	...
117	Extra Virgin	3	...

Products 1..1 0..N

Product Name (PK+)	Manufacturer	...
First Cold Press	Two Trees	...
Basil Infused	Mary's Dipping Oils	...
Mission Oil	Two Trees	...
Extra Virgin	Two Trees	...



3rd Normal Form (3NF): “nothing but the key”

- 3NF: non-key fields are only dependent on the key field
 - all non-key fields are independent from any other non-key field
 - move dependent non-key field(s) to another table
- Example: Tours: is it in 3NF?

Tours

TourID (PK)	Tour Name	Difficulty	Difficulty Rating	...
1	Big Sur Retreat	Medium	3	...
2	In the Steps of John Muir	Easy	1	...
3	The Death Valley Survivor's Trek	Difficult	5	...
4	The Mt. Whitney Climbers Tour	Difficult	5	...



Example: Tours (1)

- Answer: No
 - Difficulty depends on Difficulty Rating
- How do we transform it to 3NF?
 - if it is a calculating field then can simply remove
 - otherwise, move either column to another table (called TourDifficulty)
 - leave the “Difficulty Rating” in Tours (as it takes less storage space)
 - create a 1:N relationship from the new table to the existing table



Example: Tours (2)

- Solution for 3NF:

Tours

TourID (PK)	Tour Name	Difficulty Rating	...
1	Big Sur Retreat	3	...
2	In the Steps of John Muir	1	...
3	The Death Valley Survivor's Trek	5	...
4	The Mt. Whitney Climbers Tour	5	...

0..N

1..1

TourDifficulty

Difficulty Rating	Difficulty Description	...
1	Easy	...
3	Medium	...
5	Difficult	...



Example: Employees (calculating fields)

- Remove “Initials” field as can be calculated in program logic

Employees

EmployeeID (PK)	First Name	Middle Name	Last Name	...
121	Lilah	April	Douglas	...
122	Karyn	Calista	Reese	...
123	Chester	Wallace	Levine	...

=LEFT(Middle Name, 1)



Denormalisation

- Denormalise a table means to reverse the normalisation rule(s) on the table
- Why denormalise a table?
 - performance
 - track historical values



Reason (1): Performance

- Column “Difficulty Description” may take time to look up for each Tour
- Could add this column to Tours table, even if this violates 3NF

Tours

TourID (PK)	Tour Name	Difficulty Rating	...
1	Big Sur Retreat		3 ...
2	In the Steps of John Muir		1 ...
3	The Death Valley Survivor's Trek		5 ...
4	The Mt. Whitney Climbers Tour		5 ...

0..N

1..1

TourDifficulty

Difficulty Rating	Difficulty Description	...
1	Easy	...
3	Medium	...
5	Difficult	...



Reason (2): Historical data

- Invoices.Address is copied from Customers
 - a kind of historical data
- Customers.Address: still stores current address

Invoices

Invoice Number (PK)	Customer ID	Address	...
121	2	169 Table Rock Road	...
122	1	394 Mesa Palms Ave	...
123	1	394 Mesa Palms Ave	...

0..N

Customers

CustomerID (PK)	Customer	Address	...
1	Bread Express	394 Mesa Palms Ave	...
2	Delish Food	169 Table Rock Road	...
3	Snazzy Snacks	9255 Hampton Crossroad	...

1..1



(B) Logical Design Considerations



Logical design considerations

- Follow a naming convention
- Data types and precision
- Create junction tables
- Apply integrity constraints
- Enforce referential integrity



Follow a naming convention

- **Table design worksheet:** records design information about a table
- **Use naming convention to:**
 - document structure of the db
 - help understand design decisions
 - aid addition of new features

Table Design Worksheet

Table Name:

Primary Key	Foreign Key	Field Name	Data Type	Precision/ Length	Constraints	Required	Indexed



Naming convention

- Be consistent throughout the design
 - FirstName or First_Name or ...
- Avoid special characters
 - avoid: spaces in names
 - use alphanumeric characters only
 - don't use brackets or parenthesis
 - don't use special characters: !@#\$%^&*+=?
- Avoid reserved words
 - words used by the DBMS, e.g. SELECT, CREATE, ...
 - DATE, TIME, NAME, CHARACTER, GROUP, ...



Data types and precision

- Consult the DBMS documentation to know data types and precision
- Text data
 - alphanumeric characters and symbols
 - small strings or large blocks
 - variable or fixed length (e.g. varchar(50))
 - Long text fields:
 - CLOB, varchar(max), ...
 - can also be used for numeric data that do not require computation (e.g. phone number, zip code)
- Numeric data: whole or fractional number values
 - numeric, decimal, integer, bigint, smallint
- Approximate numbers: very large or very small values
 - float, real, single precision, double precision



Additional data types

- Date and time
- Time zone
- Boolean: Yes or No
- Additional vendor specific types
- Attachments, hyperlinks, geographic, XML and more

Table design worksheet



Table Design Worksheet

Table Name:

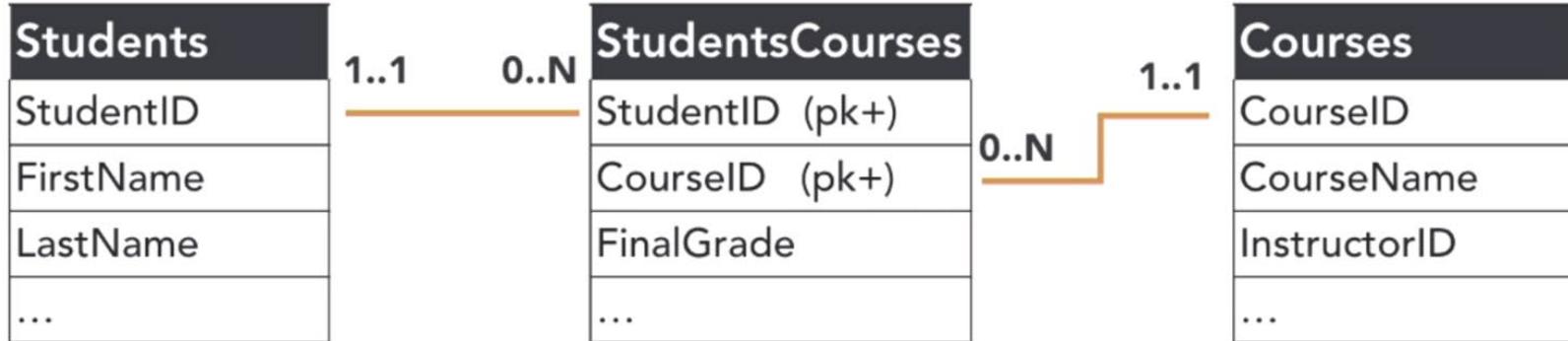
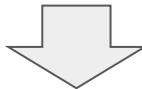


Create junction tables

- Junction table: intermediate or linking table that resolves the M-N relationship
- Effectively breaking the M-N relationship into two 1-N relationships
 - cardinality (1:1, 0:N) is applied to each 1:N relationship
- Has a composite PK referencing the PKs of the existing tables
- May contain additional attributes, pertaining to the M-N relationship



Example: (Students, Courses)





Example: (Students, Courses) example data

Students

StudentID(PK)	FirstName	LastName	...
700	Cassidy	Lamb	...
701	Sylvester	Gardner	...
702	Shellie	Hobbs	...

1..1

StudentsCourses

StudentID (PK +)	CourseID (PK+)	FinalGrade	...
701	CHEM101	A	...
701	PHYS101	B	...
702	PHYS101	A	...

0..N

Courses

CourseID (PK)	CourseName	InstructorID	...
CHEM101	Chemistry	54	...
PHYS101	Physics	13	...
STAT200	Statistics	95	...

1..1



Apply integrity constraints

- Integrity constraint is a rule that helps ensure valid data in the database
- Control data input
 - through a related table (e.g. look up table) to specify acceptable values
- Check constraint: based on the business rules
 - check the table structure
 - limit acceptable values
 - e.g. number range, date range
- NULL constraint:
 - specify if a field's values must not or can be NULL



Enforce referential integrity

- Referential integrity is a type of integrity constraint that ensures data integrity of the FK value
 - prevents orphaned records (a FK value does not refer to any PK value)
- Example: which of these is an orphaned record?

Employees

EmployeeID (PK)	FirstName	LastName	Department	...
101	Jordan	Hinton	Sales	...
102	Lilah	Douglas	Shipping	...
103	Karyn	Reese	Facilities	...
104	James	Darren	Maintenance	...

0..N



Departments

DepName(PK)
Customer Service
Facilities
Human Resources
Marketing
Product Development
Sales
Shipping



(C) Project update



Project update

- Use the **table design worksheet** to record design information for each table
- Pay particular attention to the followings:
 - normal forms (1NF, 2NF, 3NF)
 - denormalisation (**only if** necessary)
 - naming convention
 - junction table
 - integrity constraints
- Team discussion:
 - revise the data model for compliance with the new information
- Project plan:
 - adjust the Jira backlog to capture and allocate tasks.



Tutorial & Workshop

See Canvas.