MONASH University

MONASH
INFORMATION
TECHNOLOGY

# Week 5 - Normalisation

## FIT2094 - FIT3171 Databases

## **Clayton Campus S2 2019.**

GROUP OF EIGHT AUSTRALIA

# Recap

- Week 1: unit intro etc
- Week 2: Relational Model
  - Codd, PK, FK, Algebra...
- Week 3: Conceptual Model
  - ANSI/SPARC vs DBDLC
  - ERD, Chen, Attrib types, Entities
  - Case study
- Week 4: Logical Model
  - Terminology change! :)
  - ERD → Logical Relations
  - 1:1, 1:M, M:N
  - Unary, Binary, Ternary
  - Case study continues...


Img src: me.me

# Overview

- **Hour 1**
  - –Data in reports/user views
  - –Revisiting Functional Dependencies (Total/Full/Partial/Transitive)
  - –Textbook case study adapted by Lindsay
  - –UNF
  - –1NF

**… then COFFEE BREAK!**

- **Hour 2**
  - –2NF
  - –3NF
  - –Our case study: Monash Software
  - –Conclusion

# Data Normalisation

- Relations should be normalised in order to avoid anomalies which may occur when inserting, updating and deleting data - operates at the **LOGICAL** level.
- Normalisation is a ***systematic series of steps*** for progressively refining the data model.
- A formal approach to analysing relations based on their primary key (or candidate keys) and functional dependencies.
- Used:
  - as a design technique "bottom up design", and
  - as a way of validating table structures produced via "top down design" (ER modelling)

**FIGURE 6.1** Tabular representation of the report format

* against EMP_NAME indicates the project leader.
Adapted from Lindsay Smith (2019), and credits to Coronel, Rob, Morris' Textbook.

**Table name: RPT_FORMAT** **Database name: Ch06_ConstructCo**

| PROJ_NUM | PROJ_NAME | EMP_NUM | EMP_NAME | JOB_CLASS | CHG_HOUR | HOURS |
|---|---|---|---|---|---|---|
| 15 | Evergreen | 103 | June E. Arbough | Elect. Engineer | 84.50 | 23.80 |
| 15 | Evergreen | 101 | John G. News | Database Designer | 105.00 | 19.40 |
| 15 | Evergreen | 105 | Alice K. Johnson * | Database Designer | 105.00 | 35.70 |
| 15 | Evergreen | 106 | William Smithfield | Programmer | 35.75 | 12.60 |
| 15 | Evergreen | 102 | David H. Senior | Systems Analyst | 96.75 | 23.80 |
| 18 | Amber Wave | 114 | Annelise Jones | Applications Designer | 48.10 | 24.60 |
| 18 | Amber Wave | 118 | James J. Frommer | General Support | 18.36 | 45.30 |
| 18 | Amber Wave | 104 | Anne K. Ramoras * | Systems Analyst | 96.75 | 32.40 |
| 18 | Amber Wave | 112 | Darlene M. Smithson | DSS Analyst | 45.95 | 44.00 |
| 22 | Rolling Tide | 105 | Alice K. Johnson | Database Designer | 105.00 | 64.70 |
| 22 | Rolling Tide | 104 | Anne K. Ramoras | Systems Analyst | 96.75 | 48.40 |
| 22 | Rolling Tide | 113 | Delbert K. Joenbrood * | Applications Designer | 48.10 | 23.60 |
| 22 | Rolling Tide | 111 | Geoff B. Wabash | Clerical Support | 26.87 | 22.00 |
| 22 | Rolling Tide | 106 | William Smithfield | Programmer | 35.75 | 12.80 |
| 25 | Starflight | 107 | Maria D. Alonzo | Programmer | 35.75 | 24.60 |
| 25 | Starflight | 115 | Travis B. Bawangi | Systems Analyst | 96.75 | 45.80 |
| 25 | Starflight | 101 | John G. News * | Database Designer | 105.00 | 56.30 |
| 25 | Starflight | 114 | Annelise Jones | Applications Designer | 48.10 | 33.10 |
| 25 | Starflight | 108 | Ralph B. Washington | Systems Analyst | 96.75 | 23.60 |
| 25 | Starflight | 118 | James J. Frommer | General Support | 18.36 | 30.50 |
| 25 | Starflight | 112 | Darlene M. Smithson | DSS Analyst | 45.95 | 41.40 |



[Clayton Q&A]: **What could possibly go wrong? :-)**

# Problems with sample data

- JOB_CLASS invites **entry errors**
  - different phrasings: Elec. Eng. vs Elect. Engineer vs E.E.
  - also “Programer”
- Table has **redundant data**
  - Details of a charge per hour are repeated for every occurrence of job class
  - Every time an employee is assigned to a project emp_name repeated

| PROJ_NUM | PROJ_NAME | EMP_NUM | EMP_NAME | JOB_CLASS | CHG_HOUR | HOURS |
|---|---|---|---|---|---|---|
| 15 | Evergreen | 103 | June E. Arbough | Elect. Engineer | 84.50 | 23.8 |
| 15 | Evergreen | 101 | John G. News | Database Designer | 105.00 | 19.4 |
| 15 | Evergreen | 105 | Alice K. Johnson * | Database Designer | 105.00 | 35.7 |
| 15 | Evergreen | 106 | William Smithfield | Programmer | 35.75 | 12.6 |
| 15 | Evergreen | 102 | David H. Senior | Systems Analyst | 96.75 | 23.8 |
| 18 | Amber Wave | 114 | Annelise Jones | Applications Designer | 48.10 | 24.6 |
| 18 | Amber Wave | 118 | James J. Frommer | General Support | 18.36 | 45.3 |
| 18 | Amber Wave | 104 | Anne K. Ramoras * | Systems Analyst | 96.75 | 32.4 |
| 18 | Amber Wave | 112 | Darlene M. Smithson | DSS Analyst | 45.95 | 44.0 |
| 22 | Rolling Tide | 105 | Alice K. Johnson | Database Designer | 105.00 | 64.7 |
| 22 | Rolling Tide | 104 | Anne K. Ramoras | Systems Analyst | 96.75 | 48.4 |
| 22 | Rolling Tide | 113 | Delbert K. Joenbrood * | Applications Designer | 48.10 | 23.6 |
| 22 | Rolling Tide | 111 | Geoff B. Wabash | Clerical Support | 26.87 | 22.0 |
| 22 | Rolling Tide | 106 | William Smithfield | Programmer | 35.75 | 12.8 |
| 25 | Starflight | 107 | Maria D. Alonzo | Programmer | 35.75 | 24.6 |
| 25 | Starflight | 115 | Travis B. Bawangi | Systems Analyst | 96.75 | 45.8 |
| 25 | Starflight | 101 | John G. News * | Database Designer | 105.00 | 56.3 |
| 25 | Starflight | 114 | Annelise Jones | Applications Designer | 48.10 | 33.1 |
| 25 | Starflight | 108 | Ralph B. Washington | Systems Analyst | 96.75 | 23.6 |
| 25 | Starflight | 118 | James J. Frommer | General Support | 18.36 | 30.5 |
| 25 | Starflight | 112 | Darlene M. Smithson | DSS Analyst | 45.95 | 41.4 |

# Problems with sample data

- Relations that contain redundant information may potentially suffer from several update anomalies
- Types of update anomalies include:
  - **Insert Anomaly**
    - Insert a new employee only if they are assigned to a project… (but what if a new employee has no project?)
  - **Delete Anomaly**
    - Delete the last employee assigned to a project?
    - Delete the last employee of a particular job class?
  - **Modification (or update) Anomaly**
    - Update a job class hourly rate - need to update multiple rows

# The Normalisation Process Goals

- Creating valid relations, i.e. each relation meets the properties of the relational model. In particular:
  - Entity integrity
  - Referential integrity
  - No many-to-many relationship
  - Each cell contains a single value (is atomic).
- In practical terms:
  - Each table represents a single subject
  - No data item will be unnecessarily stored in more than one table.
  - The relationship between tables can be established (pair of PK and FK is identified).
  - Each table is void of insert, update and delete anomalies.



[Clayton Q&A]: **Terminology shift → recall**
**Entity (Conceptual, Lec 3) → Relation (Logical, Lec 4) → Table**

# Representing a form as a relation

- This process follows a **standard** approach:
  - arrive at a name for the form which indicates what it represents (its subject)
  - determine if any attribute is multivalued (repeating) **for a given entity instance of the forms subject**
    - if an attribute (or set of attributes) appears multiple times then the group of related attributes need to be shown enclosed in brackets to indicate there are multiple sets of these values for each instance
- Looking at our SAMPLE DATA
  - **Name: EMPLOYEE_PROJECT_ASSIGNMENT**
    - **simplify name to ASSIGNMENT for lecture**
  - **ASSIGNMENT (proj_num, emp_num, emp_name, job_class, chg_hour, assign_hours)**
  - i.e. the form consists of repeating rows (instances) of assignment data

# Representing a form as a relation?

**CUSTOMER ORDER**

**Order Number:** 61384 **Order Date:** 12/3/2018

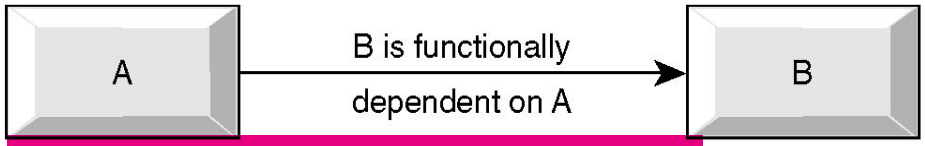
**Customer Number:** 1273
**Customer Name:** Computer Training Centre
**Customer Address:** 123 Excellent St
Monash, Vic, 3000

| PART NUMBER | DESCRIPTION | QTY ORDERED | LINE PRICE |
|---|---|---|---|
| M128 | Bookcase | 4 | 800 |
| B381 | TV Cabinet | 2 | 600 |
| R210 | Round Table | 3 | 1500 |

**ORDER** ( orderno, orderdate, custnumb, custname, custaddress (partno, partdesc, qtyordered, lineprice))

The 'inner bracketed' (partno, partdesc, qtyordered, lineprice) - is a multivalued set of attributes – called a repeating group in normalisation terminology
**You will be tested on this soon...**

# Functional Dependency Revisited




- An attribute B is FUNCTIONALLY DEPENDENT on another attribute A, if a value of A determines a single value of B at any one time.
  - A ➔ B
  - EMP# ➔ EMP_NAME
  - CUSTNUMB ➔ CUSTNAME
  - ORDER-NUMBER ➔ ORDER-DATE
    - ORDER-NUMBER - independent variable, also known as the DETERMINANT
    - ORDER-DATE - dependent variable
- TOTAL DEPENDENCY
  - attribute A determines B AND attribute B determines A
    - EMPLOYEE-NUMBER ➔ TAX-FILE-NUMBER
    - TAX-FILE-NUMBER ➔ EMPLOYEE-NUMBER

# Functional Dependency - Full/Partial

- For a ***composite*** PRIMARY KEY, it is possible to have FULL or PARTIAL dependency.
- FULL DEPENDENCY
  - occurs when an attribute is always dependent on all attributes in the composite PK
  - ORDER-NUMBER, PART-NUMBER ➔ QTY-ORDERED
- Lack of full dependency for multiple attribute key = PARTIAL DEPENDENCY
  - ORDER-NUMBER, PART-NUMBER
    ➔ QTY-ORDERED, PART-DESCRIPTION
  - here although qty-ordered is ***fully dependent*** on order-number and part-number, *only* part-number is required to determine part-description
  - part-description is said to be ***partially dependent*** on order-number and part-number

# [Marc] [Clayton] Audience Q&A

**No Flux this slide.**

**Revision using a REAL WORLD database - contents edited for clarity.**
**A Netflix movie rating table is presented below.**
**1. What's an example of Functional Dependency? (Can it be inverted?)**
**2. What's an example of a Total Dependency? (Is there one at all?)**

| title | rating | ratingdescription | release_year | user_rating_score | user_rating_size |
|---|---|---|---|---|---|
| Lucky Number Slevin | R | 100 | 2006 | | 82 |
| Grey's Anatomy | TV-14 | 90 | 2016 | 98 | 80 |
| Prison Break | TV-14 | 90 | 2008 | 98 | 80 |
| How I Met Your Mother | TV-PG | 70 | 2014 | 94 | 80 |
| Supernatural | TV-14 | 90 | 2016 | 95 | 80 |
| Breaking Bad | TV-MA | 110 | 2013 | 97 | 80 |
| The Vampire Diaries | TV-14 | 90 | 2017 | 91 | 80 |
| The Walking Dead | TV-MA | 110 | 2015 | 98 | 80 |
| Pretty Little Liars | TV-14 | 90 | 2016 | 96 | 80 |



https://github.com/datacamp/community-groupby/tree/master/data/chasewillden-netflix-shows

# [Marc] [Clayton] Audience Q&A

**No Flux this slide.**

**Revision using a REAL WORLD receipt - contents edited for clarity.**

**This is a receipt from Lincraft, (contents redacted for brevity).**

**1. Can you spot the full dependency?**

**2. Can you spot the partial dependency?**


Img src: ausphotography.net.au

# Functional Dependency - Transitive

- TRANSITIVE DEPENDENCY
  - occurs when Y depends on X, and Z depends on Y - thus Z also depends on X ie. X ➔ Y ➔ Z
  - **and** Y is not a candidate key (or part of a candidate key)
  - ORDER-NUMB ➔ CUSTOMER-NUMB ➔ CUSTOMER-NAME
- Dependencies are depicted with the help of a **Dependency Diagram**.
- Normalisation converts a relation into relations of progressively smaller number of attributes and tuples until an **optimum level of decomposition is reached - little or no data redundancy exists.**
- **The output from normalisation is a set of relations** that meet all conditions set in the relational model principles.

## Case Study: from UNF to 3NF

Lindsay's **'ASSIGNMENT'** textbook case study will be woven with the theory throughout the lecture...

Img src: @samuelzeller at Unsplash

# ASSIGNMENT case study: Sample Data (courtesy of Lindsay)

| PROJ_NUM | PROJ_NAME | EMP_NUM | EMP_NAME | JOB_CLASS | CHG_HOUR | HOURS |
|---|---|---|---|---|---|---|
| 15 | Evergreen | 103 | June E. Arbough | Elect. Engineer | 84.50 | 23.80 |
| 15 | Evergreen | 101 | John G. News | Database Designer | 105.00 | 19.40 |
| 15 | Evergreen | 105 | Alice K. Johnson * | Database Designer | 105.00 | 35.70 |
| 15 | Evergreen | 106 | William Smithfield | Programmer | 35.75 | 12.60 |
| 15 | Evergreen | 102 | David H. Senior | Systems Analyst | 96.75 | 23.80 |
| 18 | Amber Wave | 114 | Annelise Jones | Applications Designer | 48.10 | 24.60 |
| 18 | Amber Wave | 118 | James J. Frommer | General Support | 18.36 | 45.30 |
| 18 | Amber Wave | 104 | Anne K. Ramoras * | Systems Analyst | 96.75 | 32.40 |
| 18 | Amber Wave | 112 | Darlene M. Smithson | DSS Analyst | 45.95 | 44.00 |
| 22 | Rolling Tide | 105 | Alice K. Johnson | Database Designer | 105.00 | 64.70 |
| 22 | Rolling Tide | 104 | Anne K. Ramoras | Systems Analyst | 96.75 | 48.40 |
| 22 | Rolling Tide | 113 | Delbert K. Joenbrood * | Applications Designer | 48.10 | 23.60 |
| 22 | Rolling Tide | 111 | Geoff B. Wabash | Clerical Support | 26.87 | 22.00 |
| 22 | Rolling Tide | 106 | William Smithfield | Programmer | 35.75 | 12.80 |
| 25 | Starflight | 107 | Maria D. Alonzo | Programmer | 35.75 | 24.60 |
| 25 | Starflight | 115 | Travis B. Bawangi | Systems Analyst | 96.75 | 45.80 |
| 25 | Starflight | 101 | John G. News * | Database Designer | 105.00 | 56.30 |
| 25 | Starflight | 114 | Annelise Jones | Applications Designer | 48.10 | 33.10 |
| 25 | Starflight | 108 | Ralph B. Washington | Systems Analyst | 96.75 | 23.60 |
| 25 | Starflight | 118 | James J. Frommer | General Support | 18.36 | 30.50 |
| 25 | Starflight | 112 | Darlene M. Smithson | DSS Analyst | 45.95 | 41.40 |

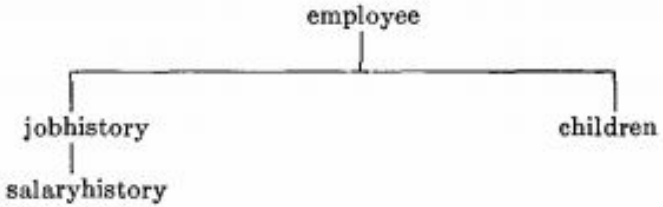
* against EMP_NAME indicates the project leader

# Unnormalised Form (UNF)

- The UNF representation of a relation is the representation which you have mapped from your inspection of the form
  - it is a **single** named representation (name is not pluralised)
  - no PK etc have as yet been identified

- **ASSIGNMENT** (proj_num, emp_num, emp_name, job_class, chg_hour, assign_hours)
- **ORDER** (orderno, orderdate, custnumb, custname, custaddress (partno, partdesc, qtyordered, lineprice))

Can ASSIGNMENT and/or ORDER be called a relation?
If not, why not?




employee (*man#*, name, birthdate, jobhistory, children)
jobhistory (*jobdate*, title, salaryhistory)
salaryhistory (*salarydate*, salary)
children (*childname*, birthyear)

FIG. 3(a). Unnormalized set

employee′ (*man#*, name, birthdate)
jobhistory′ (*man#*, *jobdate*, title)
salaryhistory′ (*man#*, *jobdate*, *salarydate*, salary)
children′ (*man#*, *childname*, birthyear)

FIG. 3(b). Normalized set

Normalization proceeds as follows. Starting with the relation at the top of the tree, take its primary key and expand each of the immediately subordinate relations by inserting this primary key domain or domain combination. The primary key of each expanded relation consists of the primary key before expansion augmented by the primary key copied down from the parent relation. Now, strike out from the parent relation all nonsimple domains, remove the top node of the tree, and repeat the same sequence of operations on each remaining subtree.

If normalization as described above is to be applicable, the unnormalized collection of relations must satisfy the following conditions:

(1) The graph of interrelationships of the nonsimple domains is a collection of trees.

(2) No primary key has a component domain which is nonsimple.

**Codd (1970).** A relational model of data for large shared data banks. Comms. of the ACM. Vol 13:6.

Monash Library access:
https://monash.hosted.exlibrisgroup.com/primo-explore/fulldisplay?docid=TN_acm362685&context=PC&vid=MONUI&search_scope=au_everything&tab=default_tab&lang=en_US

# First Normal Form (1NF) CRITERIA

- FIRST NORMAL FORM (part of formal definition of a relation)
- A RELATION IS IN FIRST NORMAL FORM (1NF) IF:
  - **a unique primary key** has been identified for each tuple/row.
  - it is a valid relation
    - Entity integrity (no part of PK is null)
    - Single value for each cell (**atomic**)
    - **No repeating group, no multivalued attribute.**
  - all attributes are functionally dependent on all or part of the primary key

# UNF or 1NF?

**Are there repeating group(s), if any, in the example below?**

| Position | Title | Artist | Last Week Pos | Time In Chart | Highest Pos |
|---|---|---|---|---|---|
| 3 | Shallow | Lady Gaga, Bradley Cooper | 2 | 24 | 1 |
| 4 | Good Lord | Birds of Tokyo | 8 | 3 | 4 |
| 5 | Dancing With A Stranger | Sam Smith, Normani | 4 | 10 | 3 |
| 6 | 7 Minutes | Dean Lewis | 10 | 9 | 6 |

Is this example in UNF or 1NF?



Source: as of 27 March 2019.
https://www.ariacharts.com.au/charts/digital-tracks-chart

## Q1. Case study:
## Is there any repeating group for the ASSIGNMENT relation?

A. Yes (what is it)?

B. No

| PROJ_NUM | PROJ_NAME | EMP_NUM | EMP_NAME | JOB_CLASS | CHG_HOUR | HOURS |
|---|---|---|---|---|---|---|
| 15 | Evergreen | 103 | June E. Arbough | Elect. Engineer | 84.50 | 23.80 |
| 15 | Evergreen | 101 | John G. News | Database Designer | 105.00 | 19.40 |
| 15 | Evergreen | 105 | Alice K. Johnson * | Database Designer | 105.00 | 35.70 |
| 15 | Evergreen | 106 | William Smithfield | Programmer | 35.75 | 12.60 |
| 15 | Evergreen | 102 | David H. Senior | Systems Analyst | 96.75 | 23.80 |
| 18 | Amber Wave | 114 | Annelise Jones | Applications Designer | 48.10 | 24.60 |
| 18 | Amber Wave | 118 | James J. Frommer | General Support | 18.36 | 45.30 |
| 18 | Amber Wave | 104 | Anne K. Ramoras * | Systems Analyst | 96.75 | 32.40 |
| 18 | Amber Wave | 112 | Darlene M. Smithson | DSS Analyst | 45.95 | 44.00 |
| 22 | Rolling Tide | 105 | Alice K. Johnson | Database Designer | 105.00 | 64.70 |
| 22 | Rolling Tide | 104 | Anne K. Ramoras | Systems Analyst | 96.75 | 48.40 |
| 22 | Rolling Tide | 113 | Delbert K. Joenbrood * | Applications Designer | 48.10 | 23.60 |
| 22 | Rolling Tide | 111 | Geoff B. Wabash | Clerical Support | 26.87 | 22.00 |
| 22 | Rolling Tide | 106 | William Smithfield | Programmer | 35.75 | 12.80 |
| 25 | Starflight | 107 | Maria D. Alonzo | Programmer | 35.75 | 24.60 |
| 25 | Starflight | 115 | Travis B. Bawangi | Systems Analyst | 96.75 | 45.80 |
| 25 | Starflight | 101 | John G. News * | Database Designer | 105.00 | 56.30 |
| 25 | Starflight | 114 | Annelise Jones | Applications Designer | 48.10 | 33.10 |
| 25 | Starflight | 108 | Ralph B. Washington | Systems Analyst | 96.75 | 23.60 |
| 25 | Starflight | 118 | James J. Frommer | General Support | 18.36 | 30.50 |
| 25 | Starflight | 112 | Darlene M. Smithson | DSS Analyst | 45.95 | 41.40 |

# UNF to 1NF transformation...

- Move from UNF to 1NF by:
  1. identify a **unique identifier** for the repeating group.
  2. ***remove the repeating group*** along with [a copy of] the **PK** of the main relation.
  3. The PK of the new relation resulting from the removal of repeating group will *normally* have a composite PK made up of the PK of the main relation and the unique identifier chosen in Step 1. above, but this ***must be checked***.

**Q2. Case study: Assume that we have started with ASSIGNMENT in UNF …**

**ASSIGNMENT(proj_num, emp_num, proj_name, emp_name, job_class, chg_hrs, assign_hours).**
**What would be the 1NF of this UNF relation?**

A. Two relations
- PROJECT (proj_num, proj_name) and
- ASSIGNMENT (proj_num, emp_num, emp_name, job_class, chg_hour, assign_hours)

B. ASSIGNMENT (proj_num, emp_num, proj_name, emp_name, job_class, chg_hour, assign_hours)

C. PROJECT (proj_num, proj_name)

D. ASSIGNMENT (proj_num, emp_num, job_class, proj_name, emp_name, chg_hour, assign_hours)

## 4.2 Conversion to Second Normal Form

Now, we observe that in the relation $PJ_1$ there are some irregularities in the dependence of attributes upon the primary key (P#,J#). Informally, we observe that QC is an attribute of the entire key, while JD, JM#, are attributes of the J# component of the primary key. These irregularities give rise to the following anomalies:

1) unless fictitious part numbers are introduced, data concerning a new project cannot be recorded until the project uses some parts (an insertion anomaly);

2) if only one kind of part remains in use by a project, deletion of data concerning that part causes deletion of the last remaining information on that project, while previous deletions did not have this consequence (a deletion anomaly);

3) if a change is made to the value of an attribute of a project (e.g., the manager's serial number JM#), the number of copies of this information to be updated in the data model depends on the number of parts in use by that project at the instant the update is performed (an update anomaly).

Codd's "Further Normalisation" ideas, as summarised in:
**Codd (1971). Normalized data base structure: a brief tutorial. Proceedings of SIGFIDET '71 Workshop on Data Description, Access and Control.**

**Codd (1971).** "Further Normalization of the Data Base Relational Model." (Presented at Courant Computer Science Symposia Series 6, "Data Base Systems," New York City, May 24th-25th, 1971.)

# 2NF CRITERIA

- A RELATION IS IN 2NF IF -
  - all nonkey attributes are functionally dependent on the **entire** primary key (simplified definition)
    - i.e. **no partial dependencies exist**
    - NB: all nonkey attributes are functionally dependent on **any candidate key** (general definition)
  - **for this unit we will only use the simplified definition i.e. look for partial dependencies based on the primary key**

## Q3. Case study:
## Which of the following attributes has a partial dependency in the relation ASSIGNMENT?

**ASSIGNMENT(proj_num, emp_num, proj_name, emp_name, job_class, chg_hrs, assign_hours).**

*Hint: dependent on EITHER number in the composite PK, not the whole!*

A. proj_name
B. emp_name
C. job_class
D. chg_hrs
E. assign_hours
F. More than one option is correct.

| PROJ_NUM | PROJ_NAME | EMP_NUM | EMP_NAME | JOB_CLASS | CHG_HOUR | HOURS |
|---|---|---|---|---|---|---|
| 15 | Evergreen | 103 | June E. Arbough | Elect. Engineer | 84.50 | 23.80 |
| 15 | Evergreen | 101 | John G. News | Database Designer | 105.00 | 19.40 |
| 15 | Evergreen | 105 | Alice K. Johnson * | Database Designer | 105.00 | 35.70 |
| 15 | Evergreen | 106 | William Smithfield | Programmer | 35.75 | 12.60 |
| 15 | Evergreen | 102 | David H. Senior | Systems Analyst | 96.75 | 23.80 |
| 18 | Amber Wave | 114 | Annelise Jones | Applications Designer | 48.10 | 24.60 |
| 18 | Amber Wave | 118 | James J. Frommer | General Support | 18.36 | 45.30 |
| 18 | Amber Wave | 104 | Anne K. Ramoras * | Systems Analyst | 96.75 | 32.40 |
| 18 | Amber Wave | 112 | Darlene M. Smithson | DSS Analyst | 45.95 | 44.00 |
| 22 | Rolling Tide | 105 | Alice K. Johnson | Database Designer | 105.00 | 64.70 |
| 22 | Rolling Tide | 104 | Anne K. Ramoras | Systems Analyst | 96.75 | 48.40 |
| 22 | Rolling Tide | 113 | Delbert K. Joenbrood * | Applications Designer | 48.10 | 23.60 |
| 22 | Rolling Tide | 111 | Geoff B. Wabash | Clerical Support | 26.87 | 22.00 |
| 22 | Rolling Tide | 106 | William Smithfield | Programmer | 35.75 | 12.80 |
| 25 | Starflight | 107 | Maria D. Alonzo | Programmer | 35.75 | 24.60 |
| 25 | Starflight | 115 | Travis B. Bawangi | Systems Analyst | 96.75 | 45.80 |
| 25 | Starflight | 101 | John G. News * | Database Designer | 105.00 | 56.30 |
| 25 | Starflight | 114 | Annelise Jones | Applications Designer | 48.10 | 33.10 |
| 25 | Starflight | 108 | Ralph B. Washington | Systems Analyst | 96.75 | 23.60 |
| 25 | Starflight | 118 | James J. Frommer | General Support | 18.36 | 30.50 |
| 25 | Starflight | 112 | Darlene M. Smithson | DSS Analyst | 45.95 | 41.40 |

## Q4. Case study:
## Which of the following attributes has a transitive dependency in the relation ASSIGNMENT?

**ASSIGNMENT(proj_num, emp_num, proj_name, emp_name, job_class, chg_hrs, assign_hours).**

*Hint: Movie_Title → Director, Studio, Year etc; Director → Director_Nationality; hence Star Wars → America.*

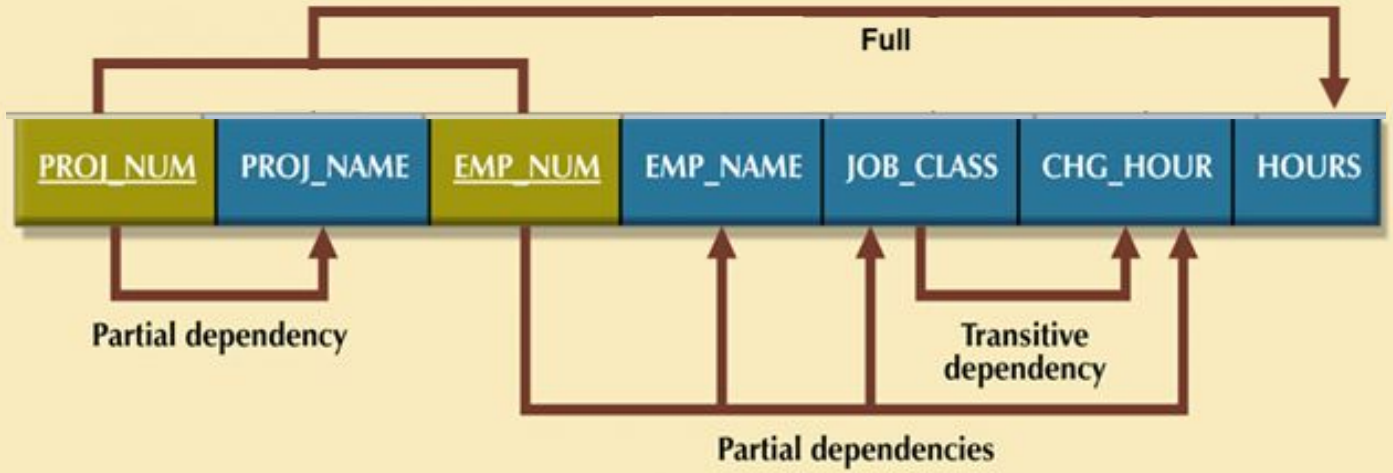
A. proj_name
B. emp_name
C. job_class
D. chg_hrs
E. assign_hours
F. More than one option is correct.

**EMPLOYEE_PROJECT_ASSIGNMENT - REPRESENTATION 2**

| PROJ_NUM | PROJ_NAME | EMP_NUM | EMP_NAME | JOB_CLASS | CHG_HOUR | HOURS |
|---|---|---|---|---|---|---|
| 15 | Evergreen | 103 | June E. Arbough | Elect. Engineer | 84.50 | 23.80 |
| 15 | Evergreen | 101 | John G. News | Database Designer | 105.00 | 19.40 |
| 15 | Evergreen | 105 | Alice K. Johnson * | Database Designer | 105.00 | 35.70 |
| 15 | Evergreen | 106 | William Smithfield | Programmer | 35.75 | 12.60 |
| 15 | Evergreen | 102 | David H. Senior | Systems Analyst | 96.75 | 23.80 |
| 18 | Amber Wave | 114 | Annelise Jones | Applications Designer | 48.10 | 24.60 |
| 18 | Amber Wave | 118 | James J. Frommer | General Support | 18.36 | 45.30 |
| 18 | Amber Wave | 104 | Anne K. Ramoras * | Systems Analyst | 96.75 | 32.40 |
| 18 | Amber Wave | 112 | Darlene M. Smithson | DSS Analyst | 45.95 | 44.00 |
| 22 | Rolling Tide | 105 | Alice K. Johnson | Database Designer | 105.00 | 64.70 |
| 22 | Rolling Tide | 104 | Anne K. Ramoras | Systems Analyst | 96.75 | 48.40 |
| 22 | Rolling Tide | 113 | Delbert K. Joenbrood * | Applications Designer | 48.10 | 23.60 |
| 22 | Rolling Tide | 111 | Geoff B. Wabash | Clerical Support | 26.87 | 22.00 |
| 22 | Rolling Tide | 106 | William Smithfield | Programmer | 35.75 | 12.80 |
| 25 | Starflight | 107 | Maria D. Alonzo | Programmer | 35.75 | 24.60 |
| 25 | Starflight | 115 | Travis B. Bawangi | Systems Analyst | 96.75 | 45.80 |
| 25 | Starflight | 101 | John G. News * | Database Designer | 105.00 | 56.30 |
| 25 | Starflight | 114 | Annelise Jones | Applications Designer | 48.10 | 33.10 |
| 25 | Starflight | 108 | Ralph B. Washington | Systems Analyst | 96.75 | 23.60 |
| 25 | Starflight | 118 | James J. Frommer | General Support | 18.36 | 30.50 |
| 25 | Starflight | 112 | Darlene M. Smithson | DSS Analyst | 45.95 | 41.40 |

# Case Study: 1NF dependencies

FIGURE 6.3 First normal form (1NF) dependency diagram




1NF (PROJ_NUM, EMP_NUM, PROJ_NAME, EMP_NAME, JOB_CLASS, CHG_HOURS, HOURS)

PARTIAL DEPENDENCIES:
(PROJ_NUM → PROJ_NAME)
(EMP_NUM → EMP_NAME, JOB_CLASS, CHG_HOUR)

TRANSITIVE DEPENDENCY:
(JOB CLASS → CHG_HOUR)

**FULL**

**proj_no, emp_no -> hours**

# 1NF to 2NF

- ASSIGNMENT(proj_num, emp_num, proj_name, emp_name, job_class, chg_hrs, assign_hours)

- Move from 1NF to 2NF by removing partial dependencies
  - 2NF: ASSIGNMENT (proj_num, emp_num, assign_hours)
  - 2NF: PROJECT (proj_num, proj_name)
  - 2NF: EMPLOYEE (emp_num, emp_name, job_class, chg_hour)


# Coffee break - see you in 10 minutes.

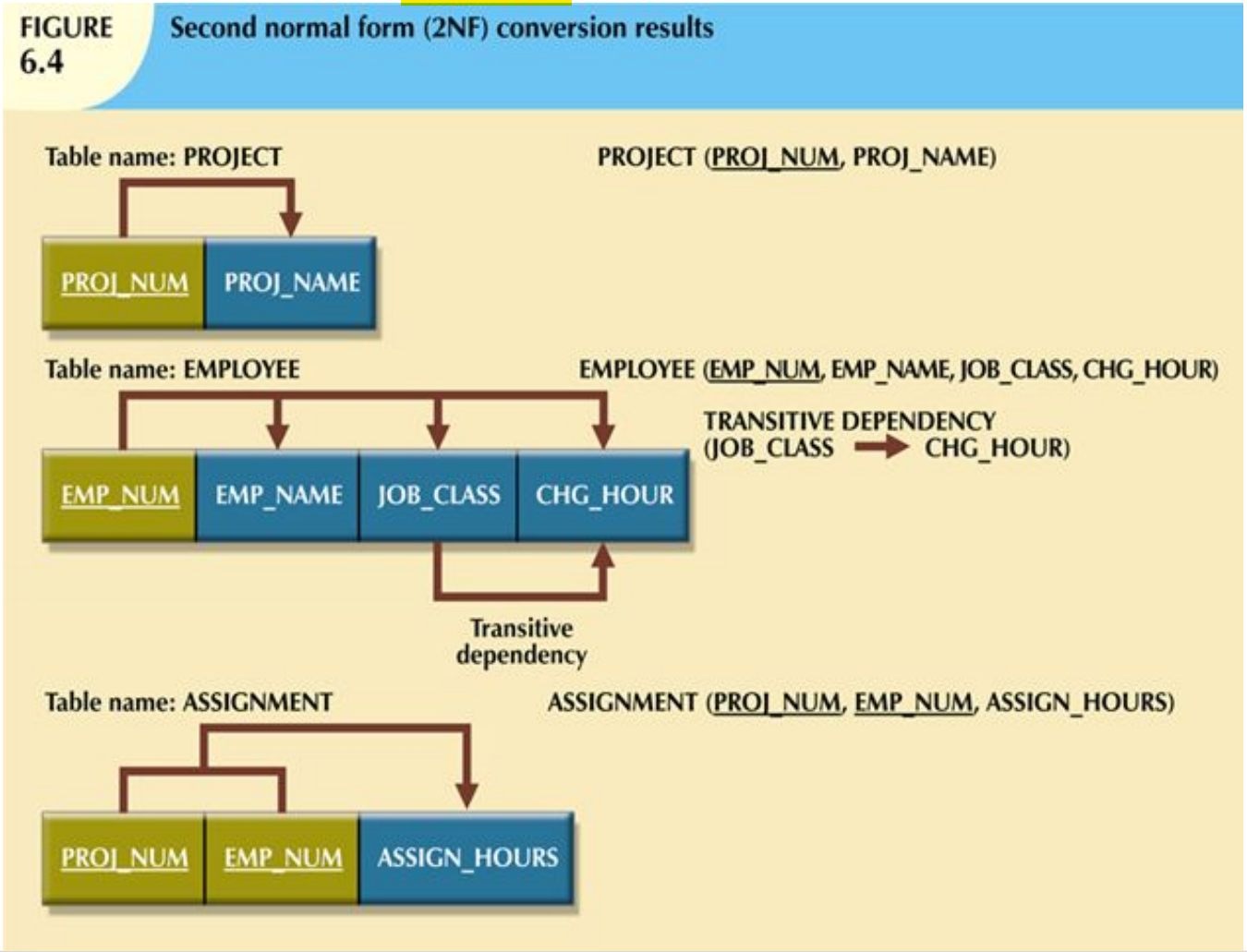
Img src: Jazmin Quaynor @jazminantoinette at Unsplash

## Q5. Case study:
## Where is the location of the FK for the relations below?

**EMPLOYEE (emp_num, emp_name, job_class, chg_hour)**
**ASSIGNMENT (proj_num, emp_num, assign_hours)**
**PROJECT (proj_num, proj_name)**

A. EMPLOYEE
B. ASSIGNMENT
C. PROJECT
D. More than one answer is correct

# Case Study: 2NF



**Q6. Case study: What type of relationship is the relationship between:**

**EMPLOYEE and ASSIGNMENT**

**and**

**PROJECT and ASSIGNMENT**

**EMPLOYEE (emp_num, emp_name, job_class, chg_hour)**
**ASSIGNMENT (proj_num, emp_num, assign_hours)**
**PROJECT (proj_num, proj_name)**

A. non-identifying, non-identifying
B. identifying, identifying
C. identifying, non-identifying
D. non-identifying, identifying



Hint: **Student** identifies Enrolment when **PK of Student** (Student ID) is within the PK of Enrolment (**Student ID** + Unit)

4.3 Conversion to Third Normal Form

To illustrate the final step of normalization, we consider the EMPLOYEE relation of Fig. 6. A sample snapshot of this relation is exhibited in Fig. 11 below.

EMPLOYEE

| E | (E#, | EMPLOYEE NUMBER | JC, | JOBCODE | D#, | DEPT NUMBER | DM#, | DEPT MGR NUMBER | CONTRACT TYPE CT) |
|---|---|---|---|---|---|---|---|---|---|
| | 1 | | a | | x | | 11 | | g |
| | 2 | | c | | x | | 11 | | g |
| | 3 | | a | | y | | 12 | | n |
| | 4 | | b | | x | | 11 | | g |
| | 5 | | b | | y | | 12 | | n |
| | 6 | | c | | y | | 12 | | n |
| | 7 | | a | | z | | 13 | | n |
| | 8 | | c | | z | | 13 | | n |

Fig. 11: A Relation not in Third Normal Form

Although this relation does not possess the kind of irregularity discussed in section 4.2, some of its attributes are transitively dependent on others, and this gives rise to anomalies similar to those discussed above. Take the attribute CT for example. This is dependent on D# which, in turn, is dependent on E#.

Once again, the irregularities and their associated anomalies (see [7]) are removable by splitting the EMPLOYEE relation into the two relations $E_3$, $D_3$ shown in Fig. 12.

| $E_3$(E#, | JC, | D#) |
|---|---|---|
| 1 | a | x |
| 2 | c | x |
| 3 | a | y |
| 4 | b | x |
| 5 | b | y |
| 6 | c | y |
| 7 | a | z |
| 8 | c | z |

| $D_3$(D#, | M#, | CT) |
|---|---|---|
| x | 11 | g |
| y | 12 | n |
| z | 13 | n |

Fig. 12: Relations in Third Normal Form

Codd's "Further Normalisation" ideas, as summarised in:
**Codd (1971). Normalized data base structure: a brief tutorial. Proceedings of SIGFIDET '71 Workshop on Data Description, Access and Control.**

**Codd (1971).** "Further Normalization of the Data Base Relational Model." (Presented at Courant Computer Science Symposia Series 6, "Data Base Systems," New York City, May 24th-25th, 1971.)

# 3NF CRITERIA

- A RELATION IS IN 3NF IF -
  - all transitive dependencies have been removed
  - check for ***non key attribute dependent on another non key attribute***

- Move from 2NF to 3NF by **removing transitive dependencies**

# Case study: 2NF to 3NF

- PROJECT and ASSIGN already in 3NF
  - **3NF PROJECT (proj_num, proj_name)**
  - **3NF ASSIGNMENT (proj_num, emp_num, assign_hours)**
- 2NF EMPLOYEE (emp_num, emp_name, job_class, chg_hour)
  - It has transitive dependency, job_class → chg_hour.
    - Remember: job_class is NONKEY!
    - Remove the attributes with transitive dependency into a new relation.
    - The determinant [job_class] will be an attribute in both the original and new relations (it will become the PK and FK relationship)
    - Assign the determinant to be the PK of the new relation.

# Case study: 2NF to 3NF

- After the removal of transitive dependency in EMPLOYEE, we have:
  - **3NF EMPLOYEE (emp_num, emp_name, job_class)**
  - **3NF JOB (job_class, chg_hour)**

# Case study: Relations in 3NF

FIGURE 6.5 Third normal form (3NF) conversion results

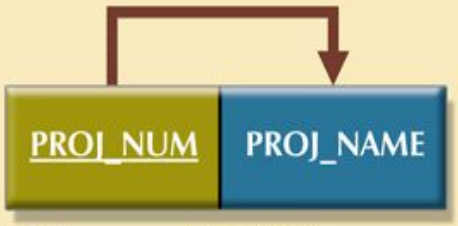



Table name: PROJECT

PROJECT (PROJ_NUM, PROJ_NAME)

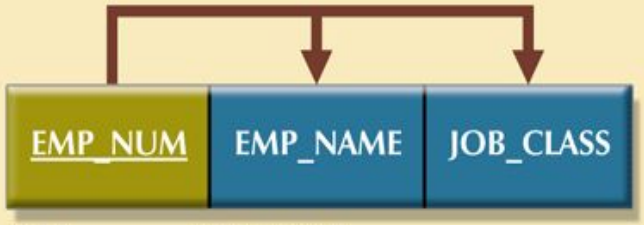



Table name: EMPLOYEE

EMPLOYEE (EMP_NUM, EMP_NAME, JOB_CLASS)

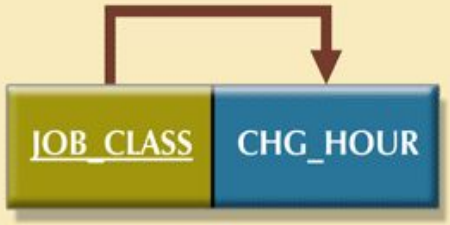



Table name: JOB

JOB (JOB_CLASS, CHG_HOUR)

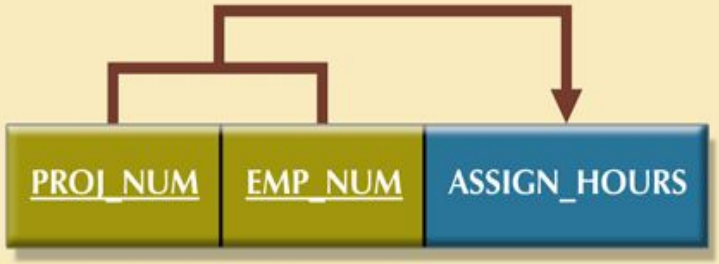



Table name: ASSIGNMENT

ASSIGNMENT (PROJ_NUM, EMP_NUM, ASSIGN_HOURS)

**Q7. Case study:**
**Where is the location of the FK for the relations below?**

**EMPLOYEE (emp_num, emp_name, job_class)**
**JOB (job_class, chg_hour)**

A. EMPLOYEE
B. JOB
C. Both EMPLOYEE and JOB

**Q8. What type of relationship is the relationship between the JOB and EMPLOYEE?**

**EMPLOYEE (emp_num, emp_name, job_class)**
**JOB (job_class, chg_hour)**

A. non-identifying
B. identifying
C. Cannot be determined

# Case study recap:
# Entire Process UNF to 3NF (Case Study)

- UNF
  - ASSIGNMENT (proj_num,emp_num, proj_name, emp_name, job_class, chg_hour, assign_hours)
- 1NF – remove repeating group...
  - ASSIGNMENT (proj_num, emp_num, proj_name, emp_name, job_class, chg_hour, assign_hours) ⇒ 1NF is only: identify PK, **no repeating group.**
- 2NF – remove partial dependencies
  - ASSIGNMENT (proj_num, emp_num, assign_hours)
  - PROJECT (proj_num, proj_name)
  - EMPLOYEE (emp_num, emp_name, job_class, chg_hour)
- 3NF – remove transitive dependencies
  - ASSIGNMENT (proj_num, emp_num, assign_hours)
  - PROJECT (proj_num, proj_name)
  - EMPLOYEE (emp_num, emp_name, job_class)
  - JOB (job_class, chg_hour)

# Case Study: Back to our Monash Software Case (since Week 3)

Img src: @samuelzeller at Unsplash

# Our Monash Software EMPLOYEE form

- List all attributes found on the form, maintain consistency with previously used attribute names if exist:
  - emp_no, emp_fname, emp_lname, emp_dob, emp_street_no, emp_street, emp_town, emp_pcode, phone_type, phone_no, degree_name, degree_institution, degree_year, fmemb_no, fmemb_name, fmemb_dob, skill_name
- Determine if any attribute is multivalued for a given entity instance (Hint: one employee can have 0 or more…?)
  - phone_type, phone_no, degree_name, degree_institution, degree_year, fmemb_no, fmemb_name, fmemb_dob, skill_name

# Our Monash Software EMPLOYEE form

- Group multivalued attributes that are related and **place in brackets**

EMPLOYEE (emp_no, emp_fname, emp_lname, emp_dob, emp_street_no, emp_street, emp_town, emp_pcode, (phone_type, phone_no), (degree_name, degree_institution, degree_year), (fmemb_no, fmemb_name, fmemb_dob), (skill_name))

- This is our beginning UNF
  - PK of main relation EMPLOYEE is emp_no
  - Four repeating groups to remove

# Monash Software EMPLOYEE form continued

EMPLOYEE (emp_no, emp_fname, emp_lname, emp_dob, emp_street_no, emp_street, emp_town, emp_pcode, (phone_type, phone_no), (degree_name, degree_institution, degree_year), (fmemb_no, fmemb_name, fmemb_dob), (skill_name))

**1NF**

EMPLOYEE (emp_no, emp_fname, emp_lname, emp_dob, emp_street_no, emp_street, emp_town, emp_pcode)

EMP_PHONE (emp_no, phone_no, phone_type)

EMP_QUALIFICATION (emp_no, degree_name, degree_institution, degree_year)

FAMILY_MEMBER (emp_no, fmemb_no, fmemb_name, fmemb_dob)

EMPLOYEE_SKILL (emp_no, skill_name)

# Monash Software EMPLOYEE form continued

**2NF**

There are no partial dependencies, relations are in 2NF

(NB: family member name/DOB depends on ENTIRE composite PK - there can be many families with Member 01;
degree institution/year depends on ENTIRE composite PK - there can be many BITS (2015), unique to BOTH employee and their institution.)

EMPLOYEE (emp_no, emp_fname, emp_lname, emp_dob, emp_street_no, emp_street, emp_town, emp_pcode)

EMP_PHONE (emp_no, phone_no, phone_type)

EMP_QUALIFICATION (emp_no, degree_name, degree_institution, degree_year)

FAMILY_MEMBER (emp_no, fmemb_no, fmemb_name, fmemb_dob)

EMPLOYEE_SKILL (emp_no, skill_name)

# Monash Software EMPLOYEE form continued

**3NF**

There are no transitive dependencies, relations are in 3NF

(NB: all nonkey degree attribs depend on PK; all nonkey family member attribs depend on PK).

EMPLOYEE (emp_no, emp_fname, emp_lname, emp_dob, emp_street_no, emp_street, emp_town, emp_pcode)

EMP_PHONE (emp_no, phone_no, phone_type)

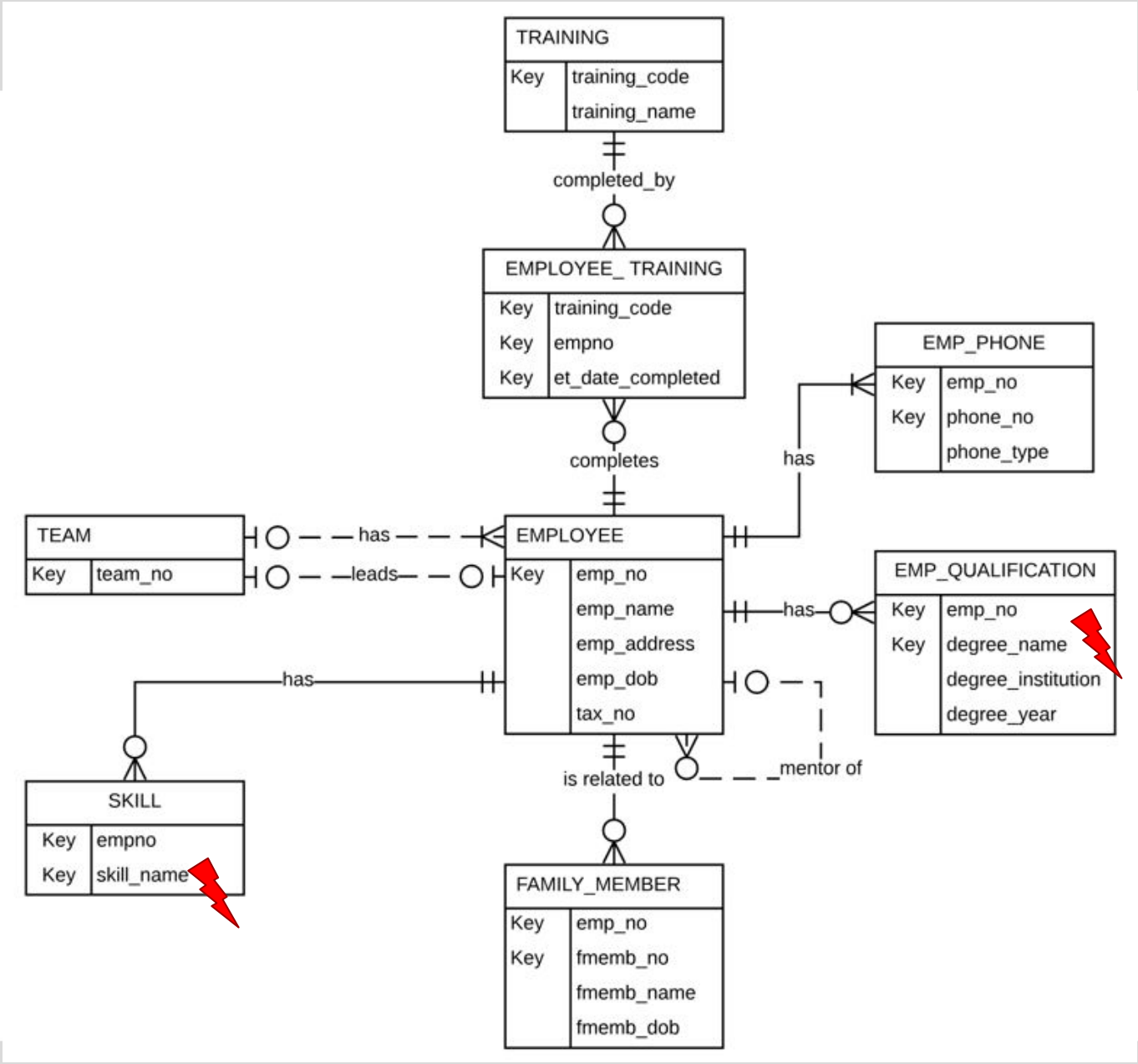
EMP_QUALIFICATION (emp_no, degree_name, degree_institution, degree_year)

FAMILY_MEMBER (emp_no, fmemb_no, fmemb_name, fmemb_dob)

EMPLOYEE_SKILL (emp_no, skill_name)

# Consolidate 3NF relations and Logical Model

- Check to see whether a new relation has been discovered from the normalisation
- Check to see whether a new attribute has been discovered from the normalisation
- Check whether FKs are correctly identified
- Update Logical Model
  - Reflect changes back to Conceptual Model


TRAINING
Key
training_code
training_name
completed_by
EMPLOYEE_ TRAINING
Key
training_code
Key
empno
Key
et_date_completed
completes
EMP_PHONE
Key
emp_no
Key
phone_no
phone_type
has
TEAM
Key
team_no
has
leads
EMPLOYEE
Key
emp_no
emp_name
emp_address
emp_dob
tax_no
has
EMP_QUALIFICATION
Key
emp_no
Key
degree_name
degree_institution
degree_year
has
has
SKILL
Key
empno
Key
skill_name
is related to
mentor of
FAMILY_MEMBER
Key
emp_no
Key
fmemb_no
fmemb_name
fmemb_dob

# Monash Software EMPLOYEE form continued

**As we are operating at the logical level**, surrogate keys may be added **AFTER** normalisation has been completed to improve design:

EMPLOYEE (emp_no, emp_fname, emp_lname, emp_dob, emp_street_no, emp_street, emp_town, emp_pcode)

EMP_PHONE (emp_no, phone_no, phone_type)

EMP_DEGREE (emp_no, *degree_code*, degree_year)

DEGREE (degree_code, degree_name, *institution_code*)

INSTITUTION (institution_code, institution_name)

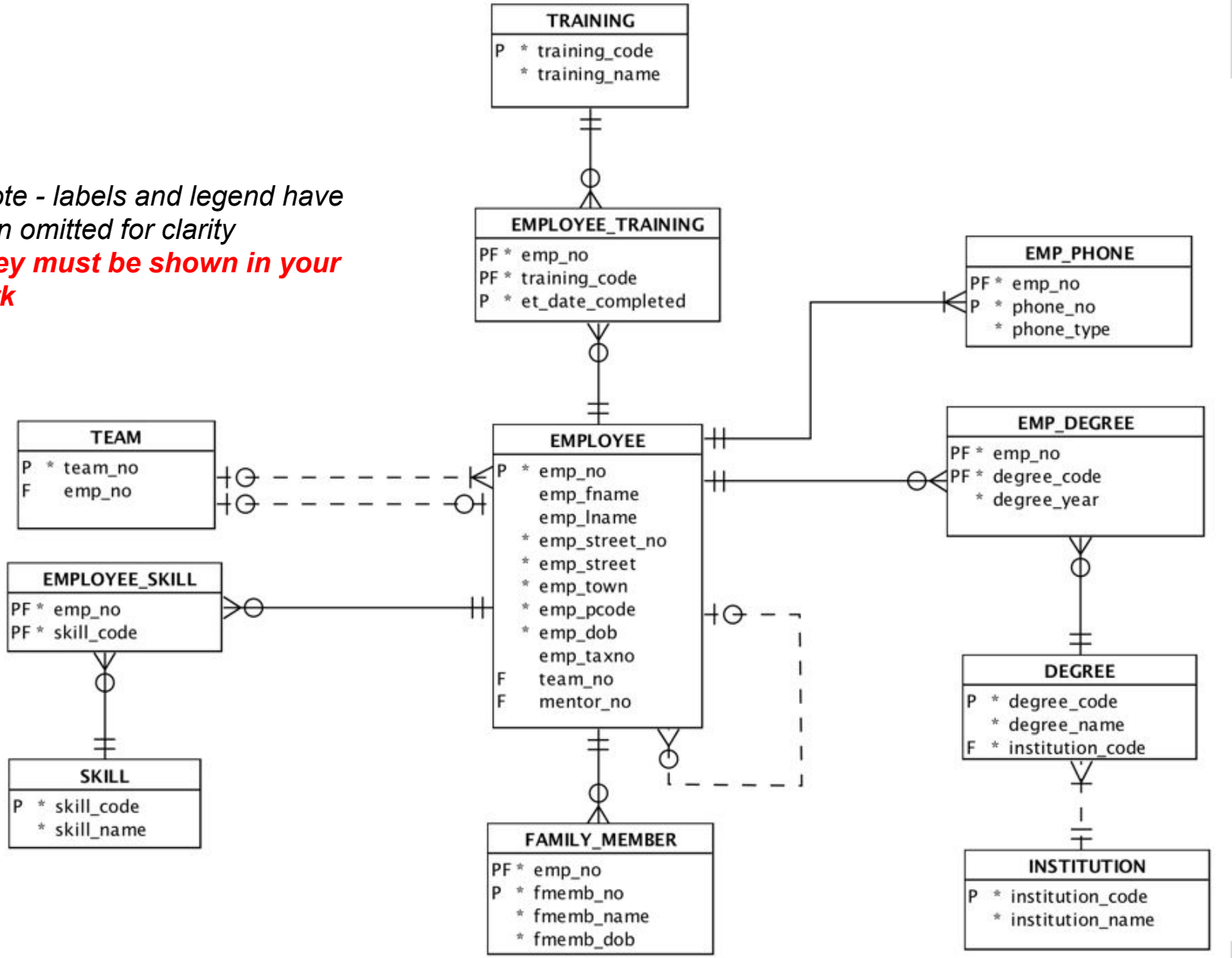
FAMILY_MEMBER (emp_no, fmemb_no, fmemb_name, fmemb_dob)

EMPLOYEE_SKILL (emp_no, *skill_code*)

SKILL (skill_code, skill_name)

***These surrogate keys must not be reflected on your conceptual model***

* *Note - labels and legend have been omitted for clarity*
- ***they must be shown in your work***



# Summary

- Things to remember
  - Primary Key selection in moving from UNF to 1NF is important, it will determine the starting point (choose your subject of interest).
  - Functional dependency
  - Process of removing attributes in relations based on the concept of 1NF, 2NF and 3NF.
    - UNF to 1NF define PK & remove repeating group.
    - 1NF to 2NF remove partial dependency.
    - 2NF to 3NF remove transitive dependency.

# FOOTNOTE: On other NFs...

- Other Normal Forms exist…
  - but for the assignment you're only expected to know up to 3NF.

- Read up on the theory of other NFs in textbook.
  - you're not expected to be assessed on > 3NF
  - good to know the very basic theory at least!
  - might help in industry -- e.g. Wu (1992)
    - ~20% of IRL business cases - tables might require advanced NFs.