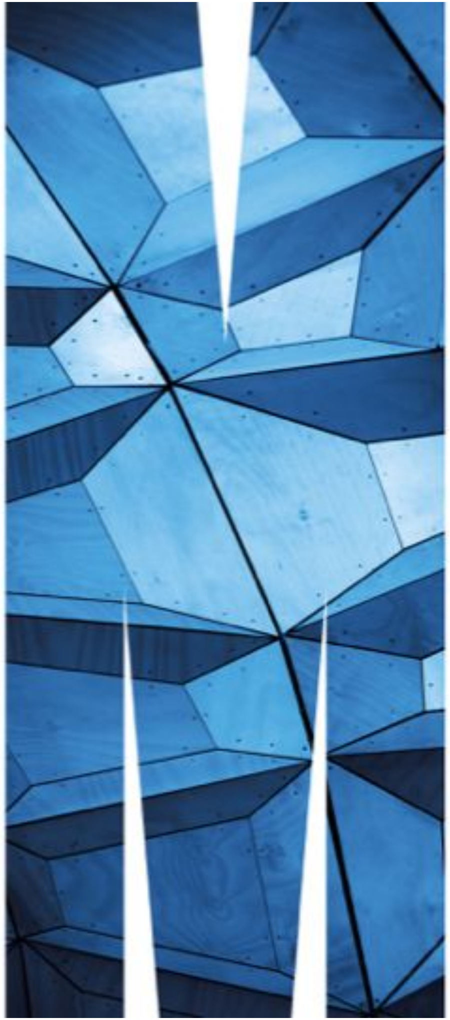
MONASH University

MONASH
INFORMATION
TECHNOLOGY



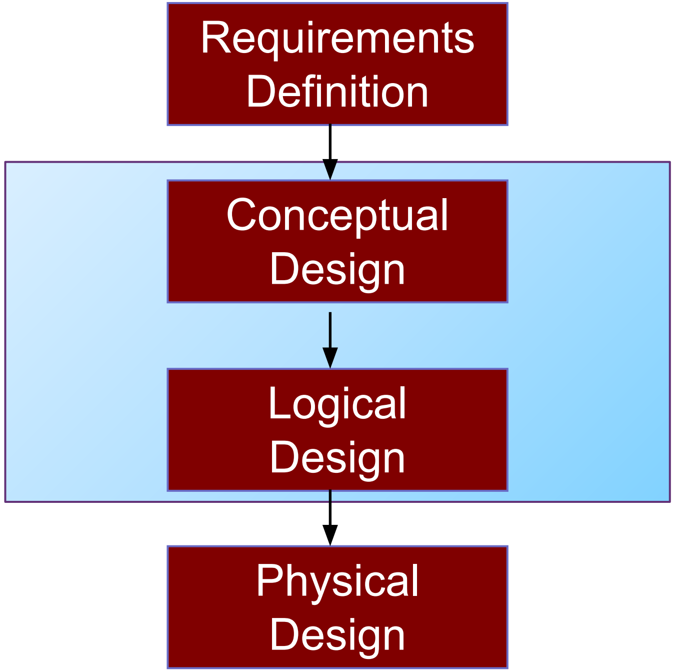
Week 2 - Database Design I

Conceptual Modelling

Workshop 2022 Semester 2

Please obtain a copy of the drone case study for this workshop from the week 2 "Workshop Resources"

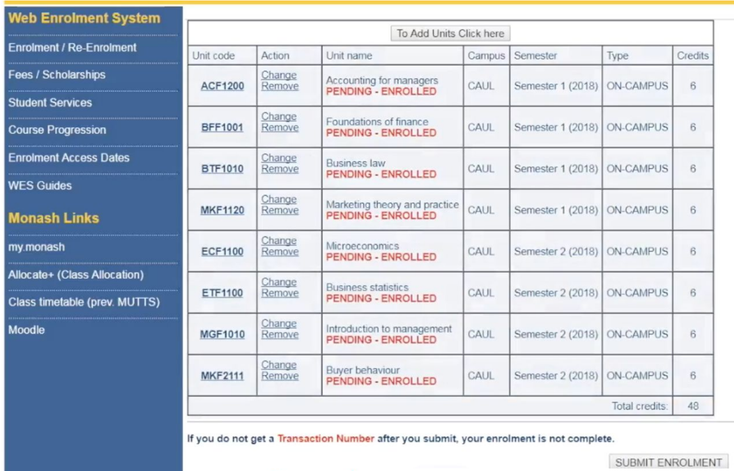
The Database Design Life Cycle



Requirements Definition

- Identify and analyse user views.
- A 'user view' may be a report to be produced or a particular type of transaction that should be supported.
- Corresponds to the external level of the ANSI/SPARC architecture.
- Output is a statement of specifications which describes the user views' particular requirements and constraints.

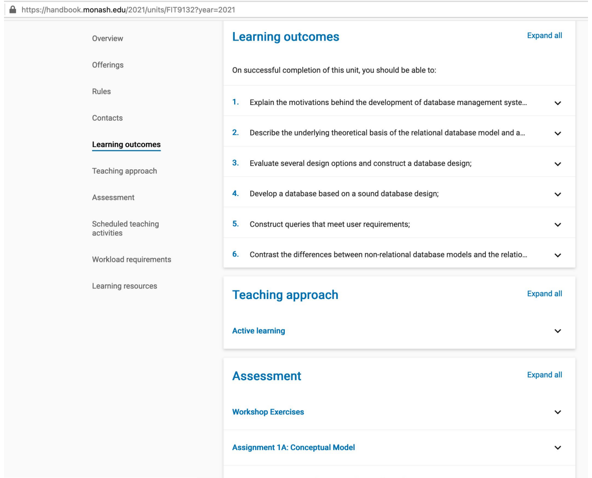
Different views of the underlying data



Unit code	Action	Unit name	Campus	Semester	Type	Credits
ACF1200	Change Remove	Accounting for managers PENDING - ENROLLED	CAUL	Semester 1 (2018)	ON-CAMPUS	6
BFF1001	Change Remove	Foundations of finance PENDING - ENROLLED	CAUL	Semester 1 (2018)	ON-CAMPUS	6
BTF1010	Change Remove	Business law PENDING - ENROLLED	CAUL	Semester 1 (2018)	ON-CAMPUS	6
MKF1120	Change Remove	Marketing theory and practice PENDING - ENROLLED	CAUL	Semester 1 (2018)	ON-CAMPUS	6
ECF1100	Change Remove	Microeconomics PENDING - ENROLLED	CAUL	Semester 2 (2018)	ON-CAMPUS	6
ETF1100	Change Remove	Business statistics PENDING - ENROLLED	CAUL	Semester 2 (2018)	ON-CAMPUS	6
MGF1010	Change Remove	Introduction to management PENDING - ENROLLED	CAUL	Semester 2 (2018)	ON-CAMPUS	6
MKF2111	Change Remove	Buyer behaviour PENDING - ENROLLED	CAUL	Semester 2 (2018)	ON-CAMPUS	6
					Total credits:	48

If you do not get a Transaction Number after you submit, your enrolment is not complete.

Student



Learning outcomes

Expand all

On successful completion of this unit, you should be able to:

1. Explain the motivations behind the development of database management syste...
2. Describe the underlying theoretical basis of the relational database model and a...
3. Evaluate several design options and construct a database design;
4. Develop a database based on a sound database design;
5. Construct queries that meet user requirements;
6. Contrast the differences between non-relational database models and the relatio...

Teaching approach

Expand all

Active learning

Assessment

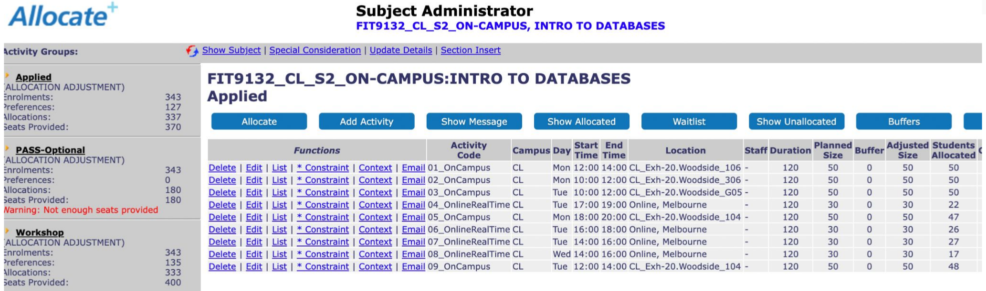
Expand all

Workshop Exercises

Assignment 1A: Conceptual Model

Assignment 1B: Normalisation and Logical Database Design

Staff & Student



Subject Administrator

FIT9132_CL_S2_ON-CAMPUS, INTRO TO DATABASES

Activity Groups:

Show Subject | Special Consideration | Update Details | Section Insert

Applied
(ALLOCATION ADJUSTMENT)

Enrolments:	343
Preferences:	127
Allocations:	337
Seats Provided:	370

PASS-Optional
(ALLOCATION ADJUSTMENT)

Enrolments:	343
Preferences:	0
Allocations:	180
Seats Provided:	180

Warning: Not enough seats provided

Workshop
(ALLOCATION ADJUSTMENT)

Enrolments:	343
Preferences:	135
Allocations:	333
Seats Provided:	400

FIT9132_CL_S2_ON-CAMPUS:INTRO TO DATABASES
Applied

Functions	Activity Code	Campus	Day	Start Time	End Time	Location	Staff	Duration	Planned Size	Buffer	Adjusted Size	Students Allocated
Delete | Edit | List | * Constraint | Context | Email	01_OnCampus	CL	Mon	12:00	14:00	CL_Exh-20.Woodside_106	-	120	50	0	50	50
Delete | Edit | List | * Constraint | Context | Email	02_OnCampus	CL	Mon	10:00	12:00	CL_Exh-20.Woodside_306	-	120	50	0	50	50
Delete | Edit | List | * Constraint | Context | Email	03_OnCampus	CL	Tue	10:00	12:00	CL_Exh-20.Woodside_G05	-	120	50	0	50	50
Delete | Edit | List | * Constraint | Context | Email	04_OnlineRealTime	CL	Tue	17:00	19:00	Online, Melbourne	-	120	30	0	30	22
Delete | Edit | List | * Constraint | Context | Email	05_OnCampus	CL	Mon	18:00	20:00	CL_Exh-20.Woodside_104	-	120	50	0	50	47
Delete | Edit | List | * Constraint | Context | Email	06_OnlineRealTime	CL	Tue	16:00	18:00	Online, Melbourne	-	120	30	0	30	26
Delete | Edit | List | * Constraint | Context | Email	07_OnlineRealTime	CL	Tue	14:00	16:00	Online, Melbourne	-	120	30	0	30	27
Delete | Edit | List | * Constraint | Context | Email	08_OnlineRealTime	CL	Wed	14:00	16:00	Online, Melbourne	-	120	30	0	30	17
Delete | Edit | List | * Constraint | Context | Email	09_OnCampus	CL	Tue	12:00	14:00	CL_Exh-20.Woodside_104	-	120	50	0	50	48

Admin

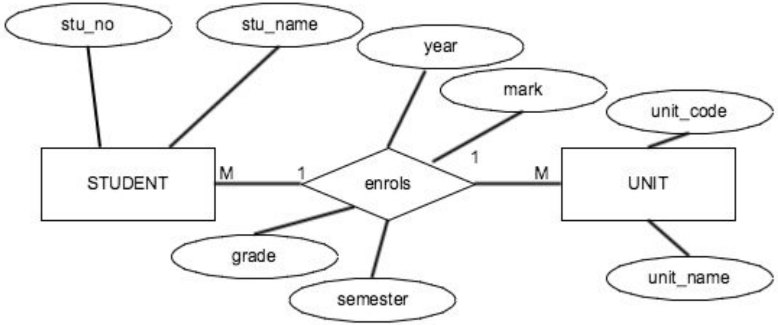
ER Modeling

- ER (Entity-Relationship) model developed by Peter Chen in 1976 to aid database design.
- Used for conceptual model (ERD).
- ER diagrams give a visual indication of the design.
- Basic components:
  - Entity
  - Attribute
  - Relationship

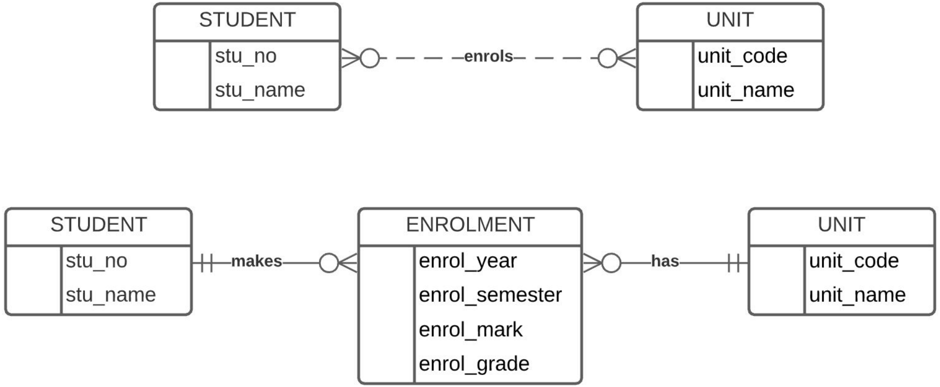
Conceptual Design

- Develop the enterprise data model.
- *Independent of all physical implementation considerations (the type of database to be used)*.
- In development try to ***minimise redundancy within provided scenario however must not add anything outside scenario***
- Various design methodologies may be employed such as UML, ER (Entity-Relationship) Modelling and Semantic Modelling.
- ER consists of ENTITIES and RELATIONSHIPS between entities
  - –An ENTITY will have attributes (things we wish to record), one or more of which will identify an entity instance (called the KEY)

ERD - Notation



Chen



Information Engineering/James Martin/Crows foot

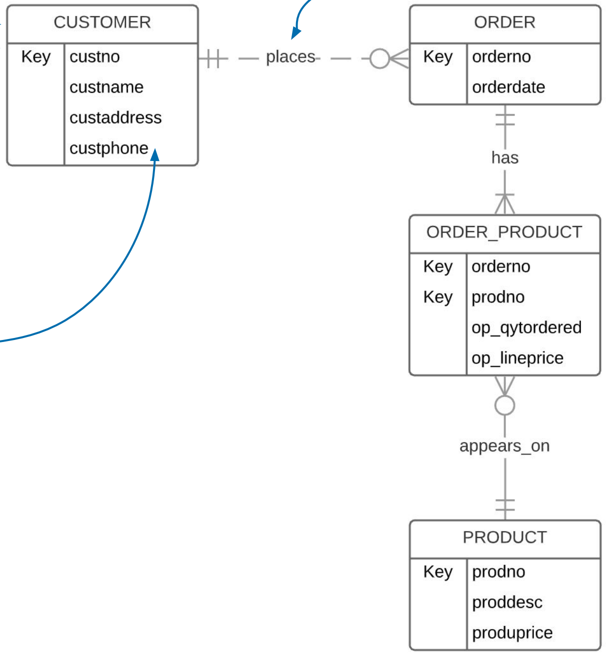
**** This is what we will be using***

Conceptual Level (ER Model)

ENTITY
Collection of "Customer(s)"

KEY ATTRIBUTE(S)
Instance identifier

NON KEY ATTRIBUTE
Other non-key attributes



RELATIONSHIP
*Connects entities - on a conceptual model this is the **ONLY** manner in which entities are connected*

Assignment 1A

Q1 Based on your pre workshop examination of the Monash Software Case Study - what entities and keys did you identify?

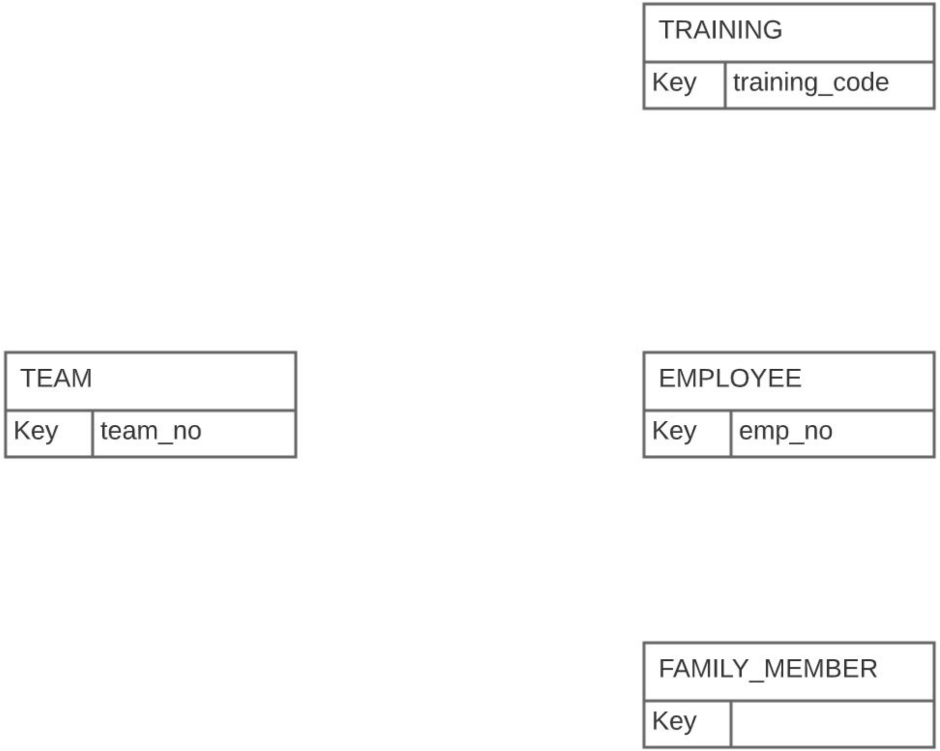
Case study: https://lms.monash.edu/mod/resource/view.php?id=10628502

List the entity in capitals followed by the key attribute - for example:

CUSTOMER, custno

Vote up or down any values with are already present

Conceptual Level (Monash Software Entities)



TRAINING

Key training_code

TEAM

Key team_no

EMPLOYEE

Key emp_no

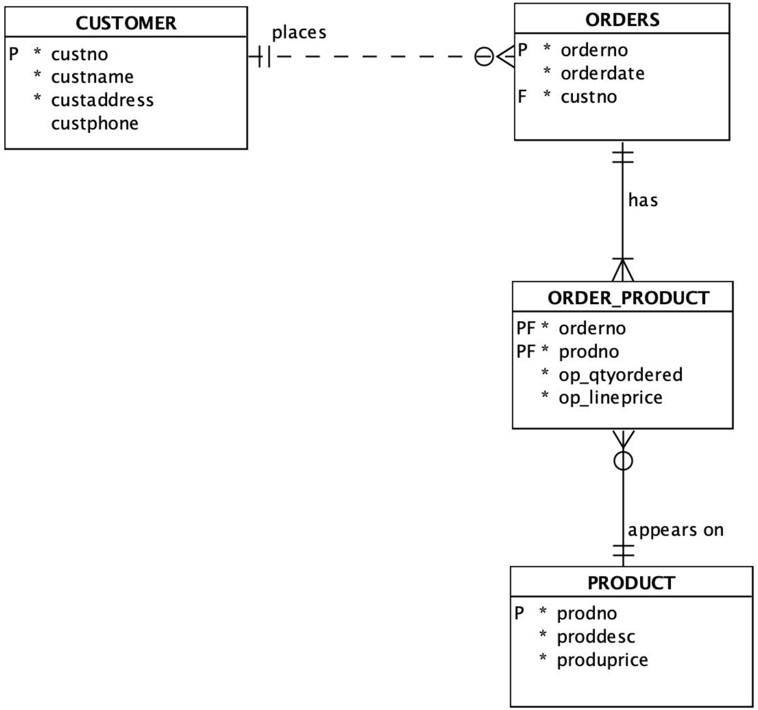
FAMILY_MEMBER

Key

Logical Design

- Develop a data model which targets a particular database type (e.g. relational, hierarchical, network, object-oriented, noSQL).
- Independent of any implementation details which are specific to any particular vendors DBMS package.
- Normalisation technique (see week 4) is used to test the correctness of a relational logical model.

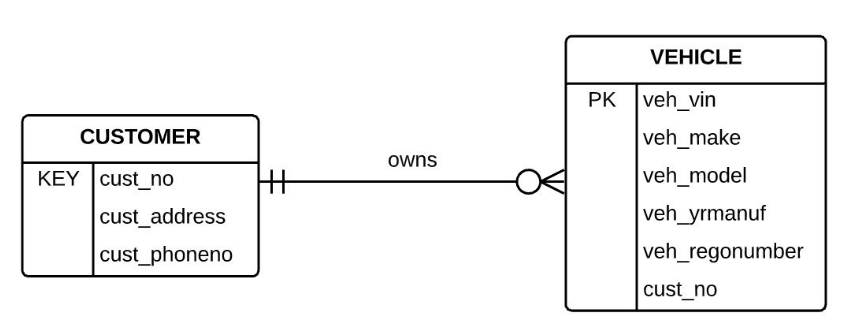
Logical Level (Logical Model - Relational)



Assignment 1B

Q2. Is the diagram shown below a valid *Conceptual Model*?

Be prepared to justify your answer with why you chose this option



CUSTOMER	
KEY	cust_no
	cust_address
	cust_phoneno

owns

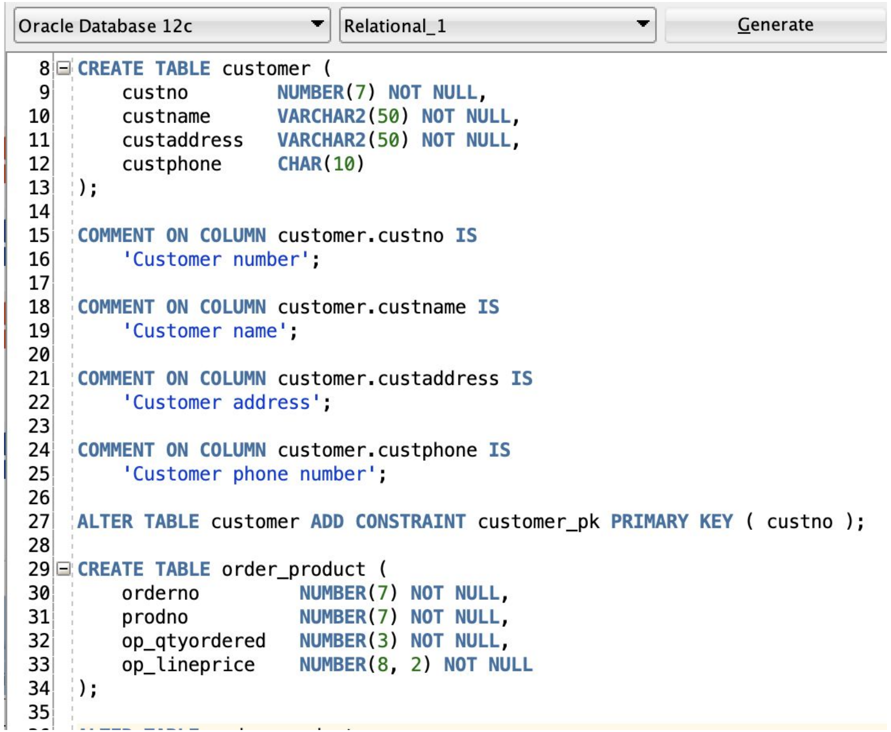
VEHICLE	
PK	veh_vin
	veh_make
	veh_model
	veh_yrmanuf
	veh_regonumber
	cust_no

A. Yes
B. No
C. Depends on how it is implemented in the database

Physical Design

- Develop a strategy for the physical implementation of the logical data model.
- Choose appropriate storage structures, indexes, file organisations and access methods which will most efficiently support the user requirements (not part of unit).
- Physical design phase is dependent on the particular DBMS in use.

Physical Level – Starting point



```
CREATE TABLE customer (
    custno         NUMBER(7) NOT NULL,
    custname       VARCHAR2(50) NOT NULL,
    custaddress    VARCHAR2(50) NOT NULL,
    custphone      CHAR(10)
);

COMMENT ON COLUMN customer.custno IS
    'Customer number';

COMMENT ON COLUMN customer.custname IS
    'Customer name';

COMMENT ON COLUMN customer.custaddress IS
    'Customer address';

COMMENT ON COLUMN customer.custphone IS
    'Customer phone number';

ALTER TABLE customer ADD CONSTRAINT customer_pk PRIMARY KEY ( custno );

CREATE TABLE order_product (
    orderno          NUMBER(7) NOT NULL,
    prodno           NUMBER(7) NOT NULL,
    op_qtyordered    NUMBER(3) NOT NULL,
    op_lineprice     NUMBER(8, 2) NOT NULL
);
```

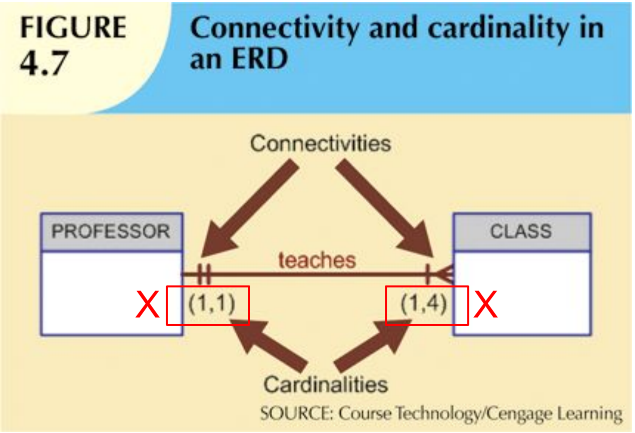
The database *schema*

Important rule for Conceptual Modelling

- ***All that is described in the brief has been included and all that has been included was described in the brief***
  - Every entity, attribute and relationship described in the brief **has been included**, and
  - **Must not add** entities, attributes and relationships which are not included as part of the brief, and
- In a real life scenario if there are concerns about features of the brief, discuss with client
  - For assignments:
    - your client will be the ed forum
    - may make assumptions provided they do not violate this rule

CONNECTIVITY/CARDINALITY

FIGURE 4.7 Connectivity and cardinality in an ERD



SOURCE: Course Technology/Cengage Learning

For Crows Foot notation specific cardinalities are not shown as above eg. (1,4), instead cardinality is depicted via min and max using standard symbols (Inside symbol = min, outside symbol = max)

CONNECTIVITY

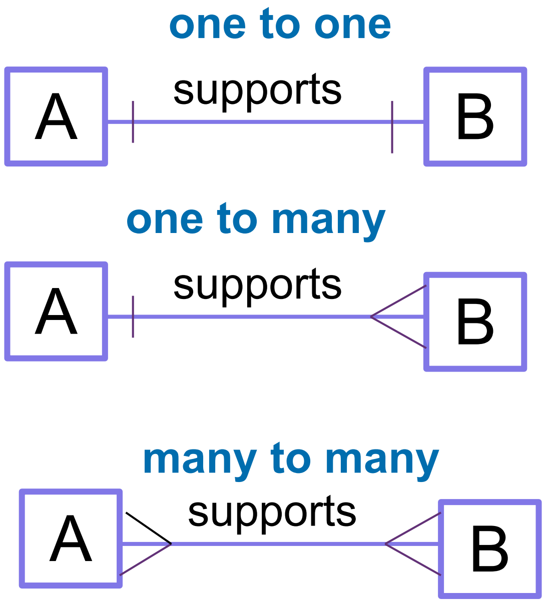
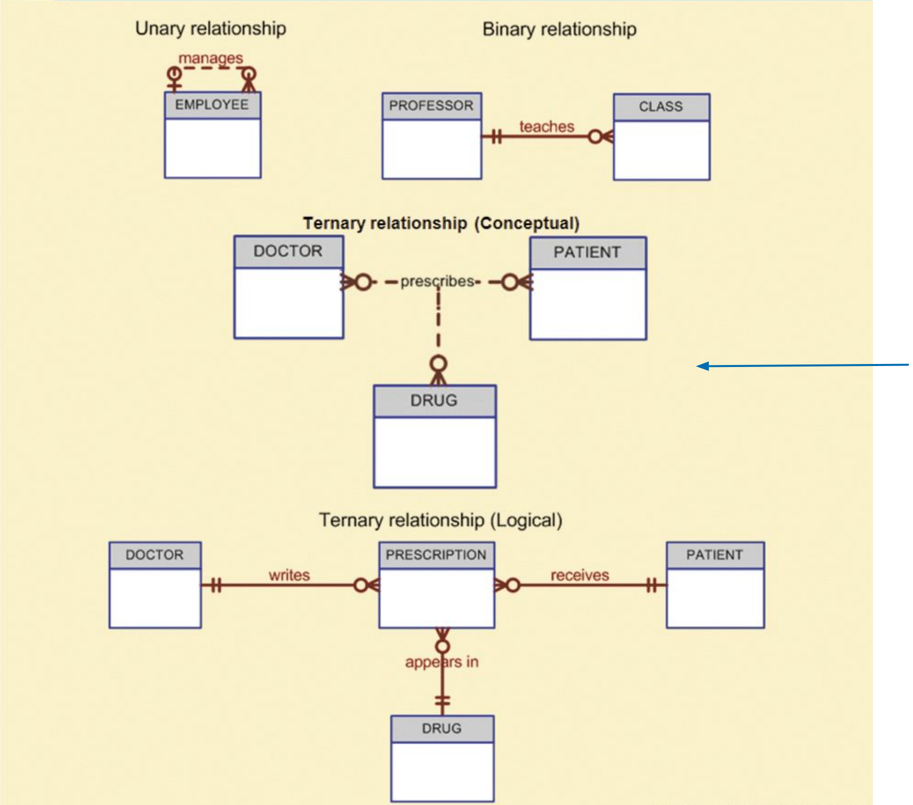


FIGURE 4.15 Three types of relationship degree



SOURCE: Course Technology/Cengage Learning

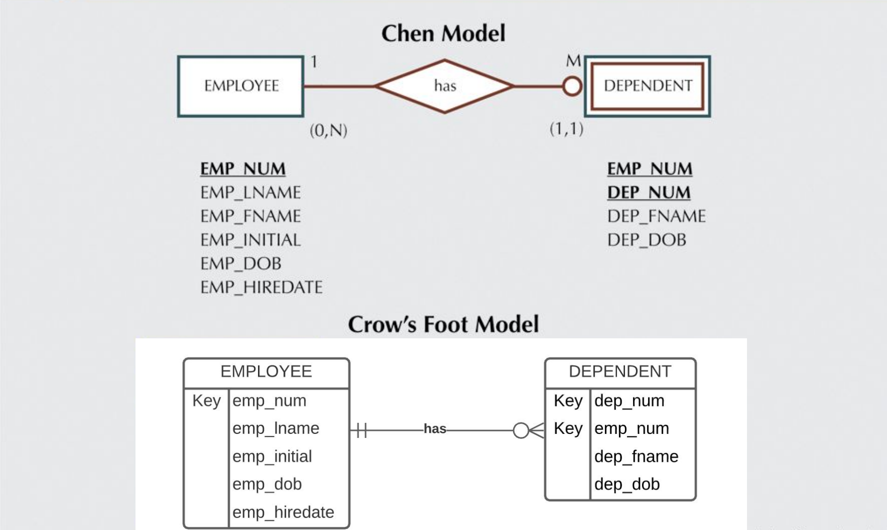
Note this is not an acceptable form of a conceptual model in Crow's Foot notation (*relationship lines must not join*)

Weak vs Strong Entity

- Strong entity
  - Has a key which may be defined without reference to other entities.
  - For example EMPLOYEE entity.
- Weak entity
  - Has a key which requires the existence of one or more other entities.
  - For example FAMILY entity - need to include the key of employee to create a suitable key for family
- Database designer often determines whether an entity can be described as weak based on business rules
  - customer pays monthly account
    - Key: cust_no, date_paid, or
    - Invented Key: payment_no (surrogate – not at conceptual level)

Weak vs Strong Entity

FIGURE 4.10 A weak entity in an ERD



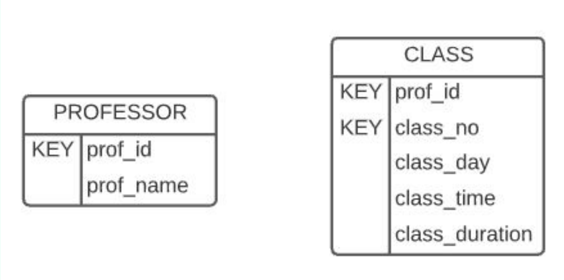
SOURCE: Course Technology/Cengage Learning

Note the Crow's Foot model shown here has been modified from the text version

Q3.The client indicates that a CLASS is identified by a combination of the the prof_id and the assigned class number for the professor (1st class, 2nd class, 3rd class etc):

prof_id,	class_no,	class_day, ...
1,	1,	Tue
1,	2,	Tue
1,	3,	Wed
2,	1,	Thu
2,	2,	Tue

....



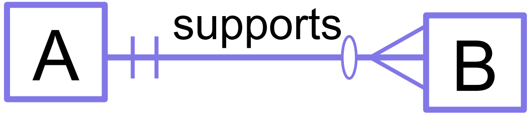
This business rule is captured in the provided diagram. Pick the correct statement for this diagram.

A. Both entities are strong entities
B. PROFESSOR is a strong entity, CLASS is a weak entity
C. CLASS is a strong entity, PROFESSOR is a weak entity
D. Both entities are weak entities

Identifying vs Non-Identifying Relationship

- **Identifying**

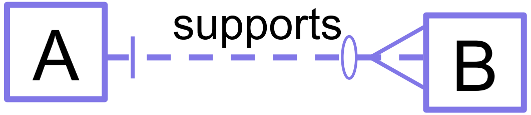
- Identifier of A is part of identifier of B.



- Shown with solid line
- ENROLMENT - STUDENT Enrolment key includes student id, which is an identifier of student.

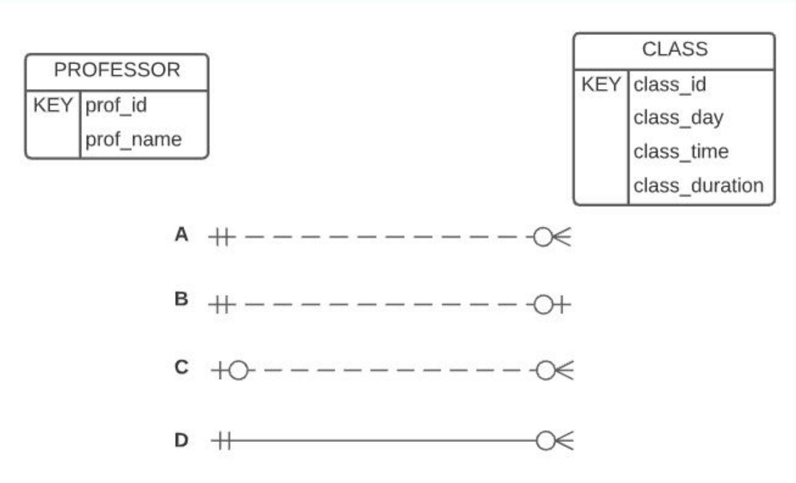
- **Non-identifying**

- Identifier of A is NOT part of identifier of B.



- Shown with broken line
- Department no (identifier of department) is not part of Employee’s identifier.

Q4. The client indicates that a professor may teach several classes, but some professors do not have any assigned classes. Each class is taken by only one professor. Note that in this diagram, each class has a unique class id (class_id). Pick the most appropriate relationship for this business rule.



Types of Attributes

- Simple
  - Cannot be subdivided
  - Age, sex, marital status

vs

- Composite
  - Can be subdivided into additional attributes
  - Address into street, city, zip

- Single-valued
  - Can have only a single value
  - Person has one social security number

vs

- Multi-valued
  - Can have many values
  - Person may have several college degrees

Types of Attributes continue

- Derived
  - Can be derived with algorithm
  - Age can be derived from date of birth

- Attribute classification is driven by your Client requirements
  - Phone Number?

Q5. The HiFlying case study indicates "*HiFlying establishes a drone hire rate as a cost per hour for customers to rent this particular drone (rates per hour are often changed over the life of the drone, as it ages, although they are only interested in recording the current cost per hour for the drone).* "

Note that although the hire rate may change over the life of the drone, it is not directly related to the hours flown.

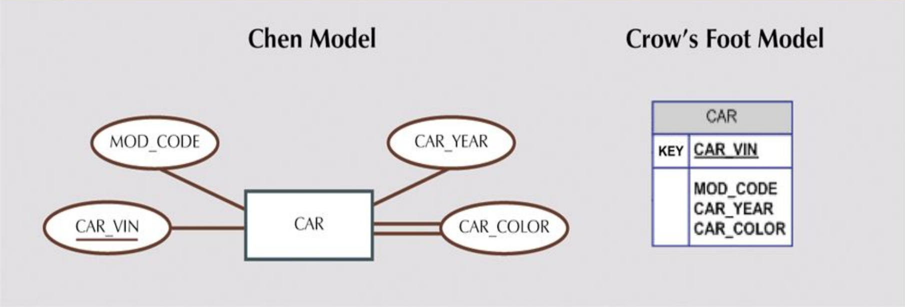
What type of attribute is the drone hire rate?

A. Simple
B. Composite
C. Single-valued
D. Multi-valued
E. Derived

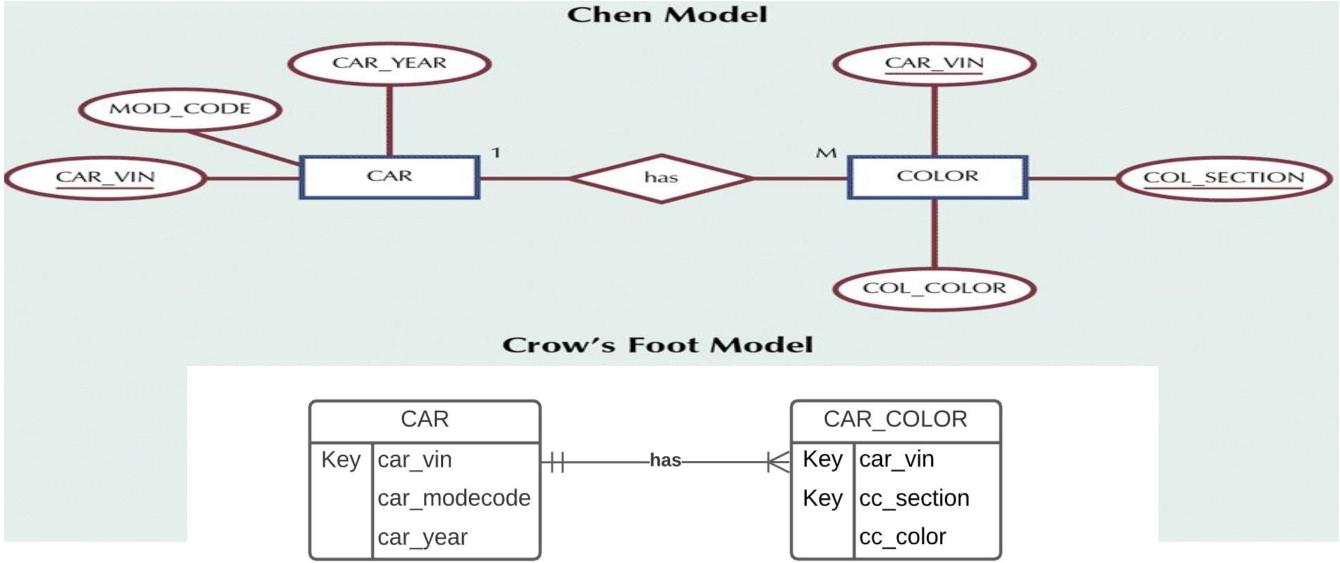
Multivalued Attribute

- An attribute that has a list of values.
- For example:
  - Car colour may consist of body colour, trim colour, bumper colour.
- Crow’s foot notation does not support multivalued attributes. Values are listed as a separate attribute.

FIGURE 4.3 A multivalued attribute in an entity

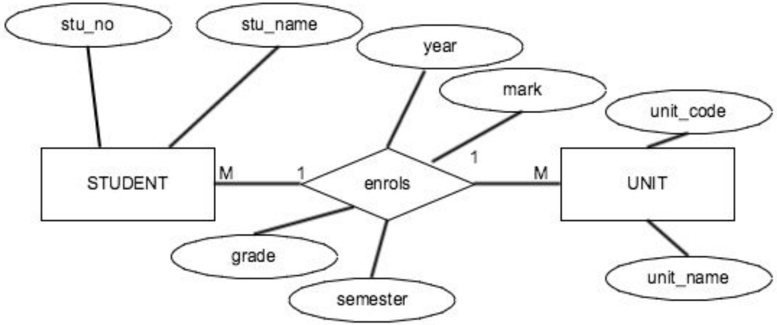


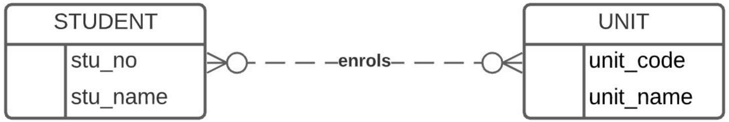
Resolving Multivalued Attributes

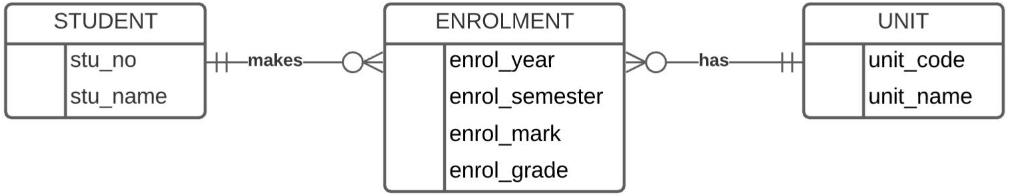


Note the Crow's Foot model shown here has been modified from the text version

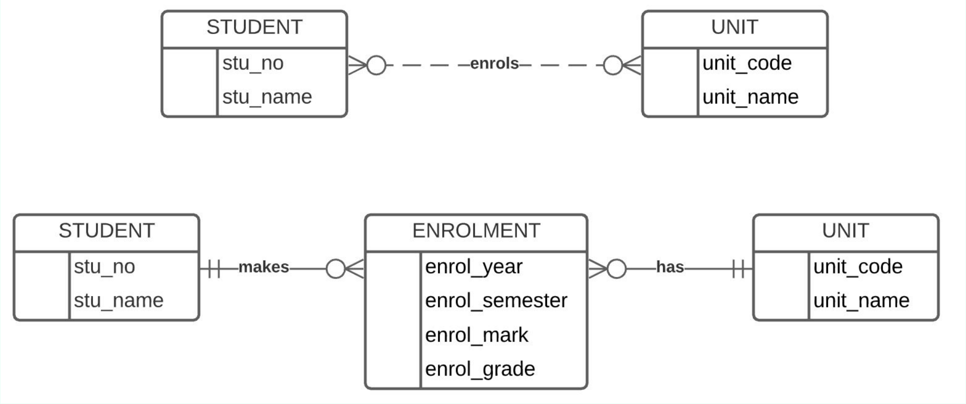
Associative (or Composite) Entity



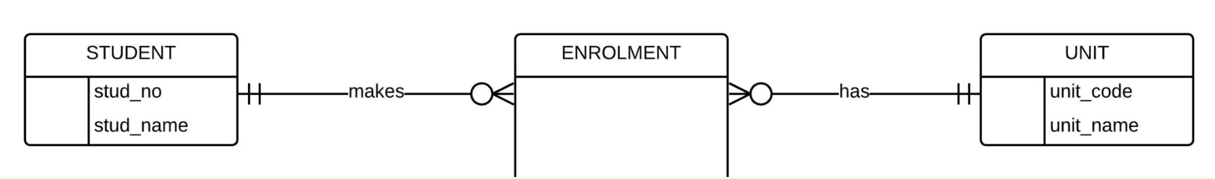




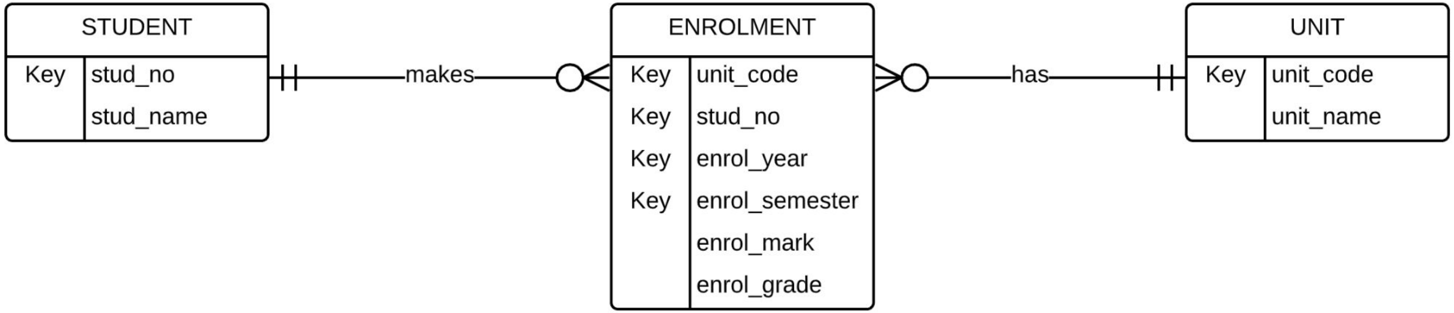
Associative (or Composite) Entity



Q6 Show all attributes for the three entities and add KEYS:



Associative or Composite Entities



STUDENT	
Key	stud_no
	stud_name

makes

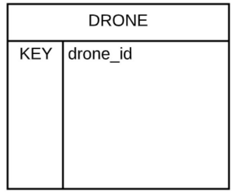
ENROLMENT	
Key	unit_code
Key	stud_no
Key	enrol_year
Key	enrol_semester
	enrol_mark
	enrol_grade

has

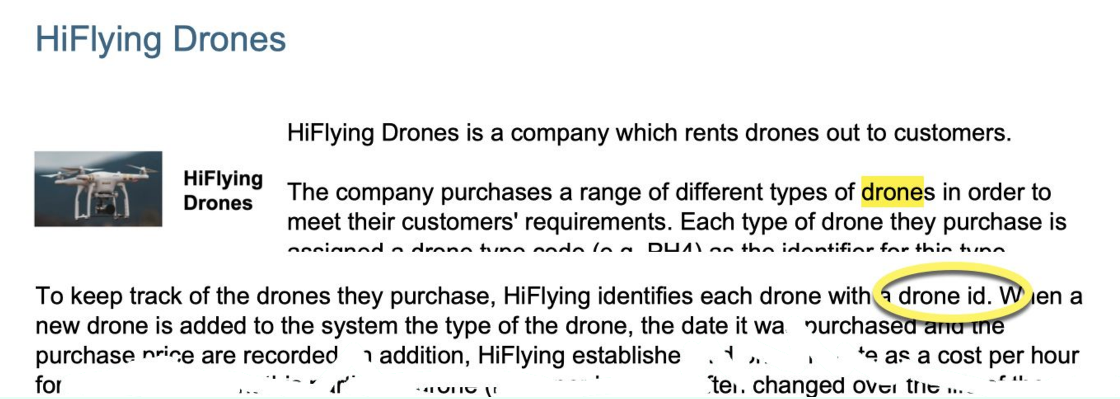
UNIT	
Key	unit_code
	unit_name

Q7. STEP 1: List ALL entities and their key attribute/s which exist in the DRONE case study: https://lms.monash.edu/mod/resource/view.php?id=10628515

For example:



DRONE	
KEY	drone_id



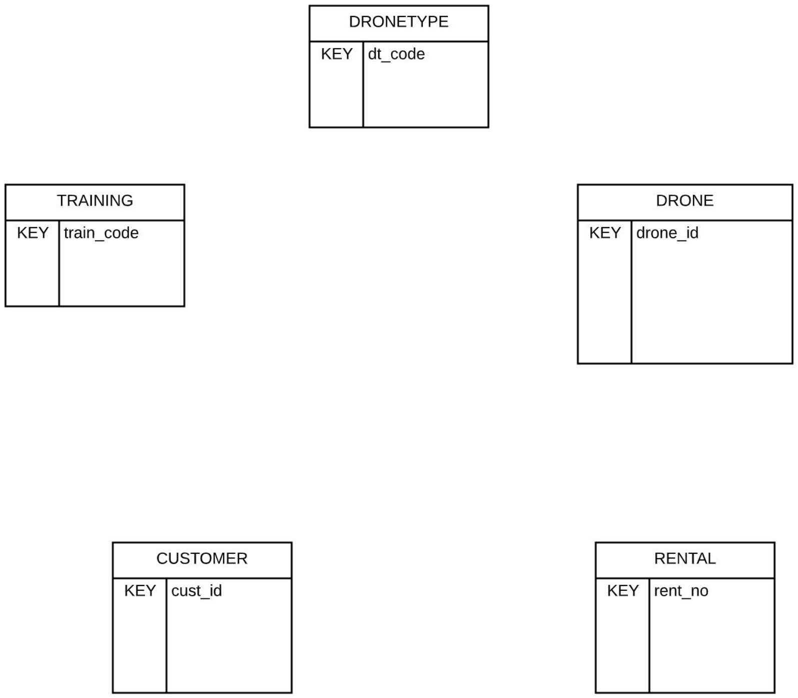
HiFlying Drones

HiFlying Drones is a company which rents drones out to customers.

The company purchases a range of different types of drones in order to meet their customers' requirements. Each type of drone they purchase is assigned a drone type code (e.g. PH4) as the identifier for this type

To keep track of the drones they purchase, HiFlying identifies each drone with a drone id. When a new drone is added to the system the type of the drone, the date it was purchased and the purchase price are recorded. In addition, HiFlying establishe ... as a cost per hour for ... ften changed over the ...

HiFlying Drones - Step 1 Identify Main Entities



DRONETYPE

KEY dt_code

TRAINING

KEY train_code

DRONE

KEY drone_id

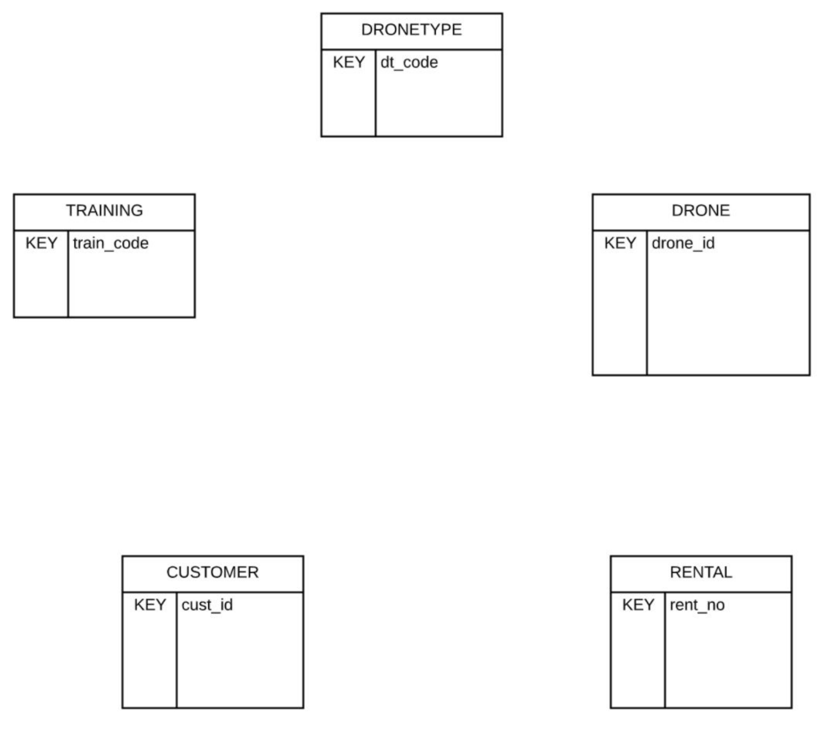
CUSTOMER

KEY cust_id

RENTAL

KEY rent_no

Q8. STEP 2: Identify the relationships which exist between these entities (remember to add an appropriate verb):



DRONETYPE	
KEY	dt_code

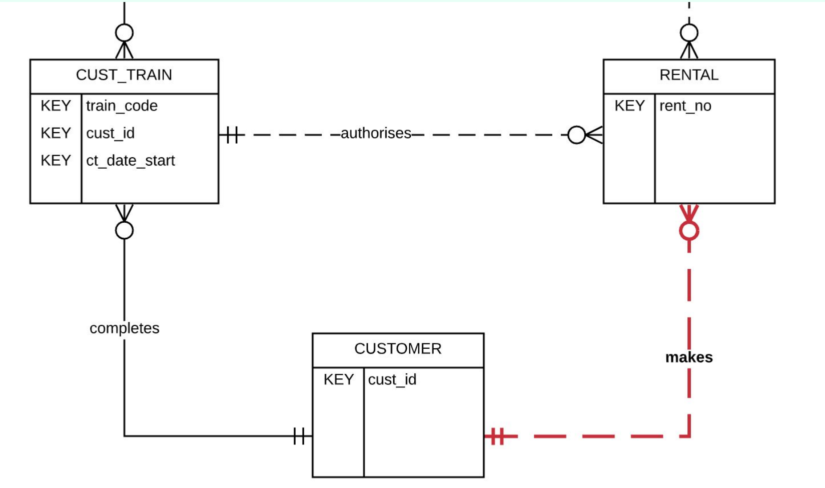
TRAINING	
KEY	train_code

DRONE	
KEY	drone_id

CUSTOMER	
KEY	cust_id

RENTAL	
KEY	rent_no

Q9. Since a customer makes a rental, should the database designer include a relationship between RENTAL and CUSTOMER?

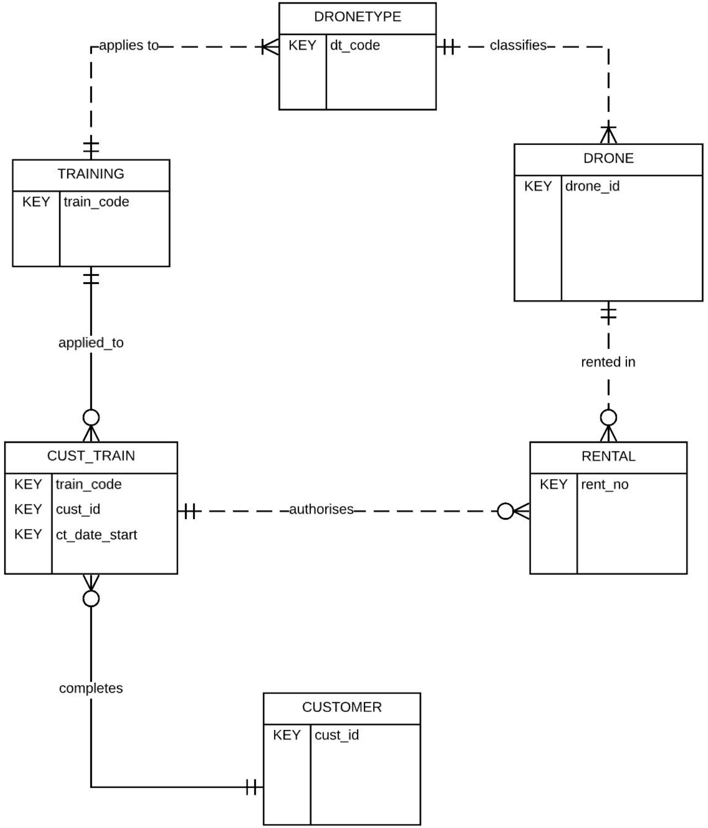


A. Yes, it is an important relationship to capture

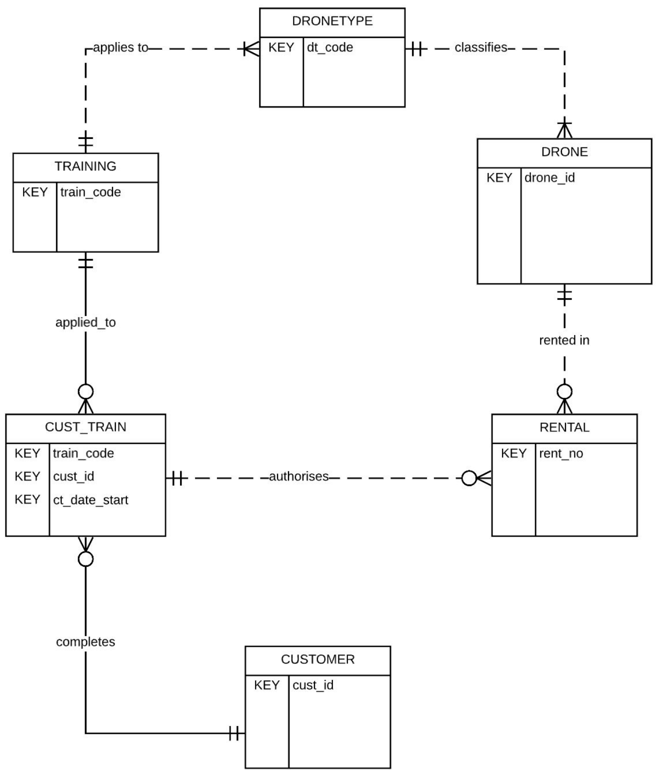
B. No, it is redundant information

C. It depends on the client's requirements

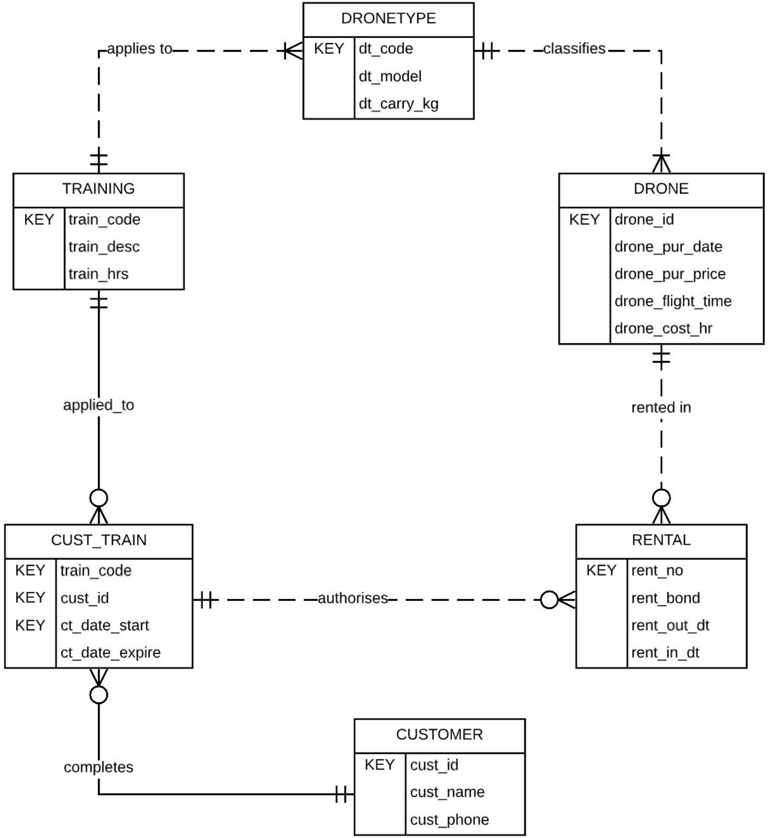
HiFlying Drones - Step 2 Identify Relationships



Q10. Step 3 Add Non-Key Attributes



HiFlying Drones - Step 3 Add Non-Key Attributes - Final Model



DRONETYPE	
KEY	dt_code
	dt_model
	dt_carry_kg

TRAINING	
KEY	train_code
	train_desc
	train_hrs

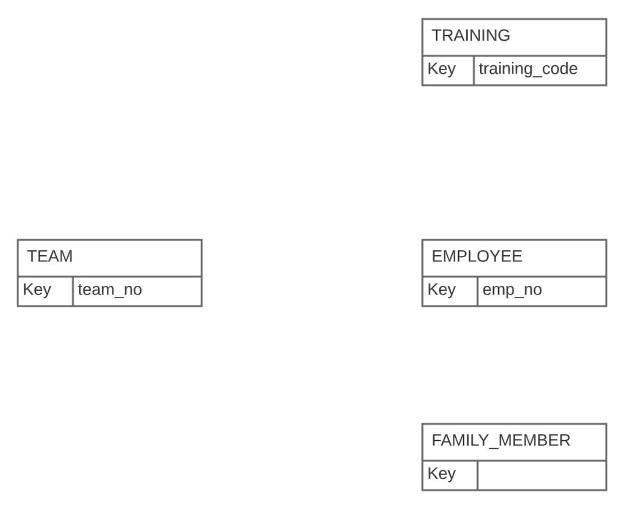
DRONE	
KEY	drone_id
	drone_pur_date
	drone_pur_price
	drone_flight_time
	drone_cost_hr

CUST_TRAIN	
KEY	train_code
KEY	cust_id
KEY	ct_date_start
	ct_date_expire

RENTAL	
KEY	rent_no
	rent_bond
	rent_out_dt
	rent_in_dt

CUSTOMER	
KEY	cust_id
	cust_name
	cust_phone

Conceptual Model (Monash Software)



You have completed

- **Step 1 identify entities and keys**

of the modelling process for Monash Software

After the workshop please proceed and complete:

- **Step 2 Identify Relationships**, and
- **Step 3 Add all non key attributes**

A video will be provided showing the full process at the start of next week.