Table of Contents

Data Integrity; Redundancy & Duplication: ................................................................. 1
Entities and Entity Relationship Diagrams (ERDs) .......................................................... 1
Sample Exam Question Q 40 – ERD; Q41 - DFD ............................................................... 4

Data Integrity; Redundancy & Duplication:

Redundancy is where the same data is stored in more than one file leading to a waste of space and possible integrity errors. Redundancy is not necessarily a bad thing in its own right; however, the database designer must be aware of it.

Duplication is where more than one copy (record) of the same data occurs or there is duplication of at least one attribute value. If the data can be removed without causing a loss of data, then this duplication is acceptable.

Integrity is the problem of ensuring that the data is accurate. Inconsistencies between two entries that purport to represent the same “fact” is known as an integrity error. This can only arise when there is data redundancy or, worse still, data duplication. (A non-redundant database can also have integrity errors by just having incorrect data stored. An example of this is a mis-keyed phone number.)

Entities and Entity Relationship Diagrams (ERDs)

When a system is first examined we look at the Processes and the Flow of data in the system. This is called the Process Model and we draw a (Logical) Data flow Diagram to represent it.

The next step is to draw a data model of the system we require, and associate with that a Data Dictionary. The Data Model we draw is an Entity Relationship Diagram (ERD).

An Entity is an object about which we want to store data. An object is something that has a number of attributes that we use to describe it. Entities may include such things as:

♦ People: suppliers; customers; students; employees; patients; teachers; lecturers.
♦ Physical Objects: printers; buildings; products; cars; books; boats.
♦ Data signifying events: orders; deliveries; statements; contracts; invoices; report cards; subject selections; (note that most of these are usually associated with a piece of paper!)

Hence an entity such as Student will have attributes such as first name; family name; address; phone number; post code; date of birth and so on.

Entities rarely exist by themselves; they are usually linked or related to other entity types. So consider a student at a school.

⇒ Students select subjects
⇒ Subjects have students select them.

The relationship between the entity types has to do with “selecting”. Hence we have:
Now this diagram shows a basic relationship, but we have some variations that must be considered.
1. A student might select 5, 6 or 7 subjects in Upper school.
2. A year 10 student may be enrolled in the School, but be absent for one semester (hence no subject selection).
3. A subject may have many students select it (Computer Science).
4. A subject may be offered, but no students select it.

Hence when we draw in the links between the entities and relationship we must have a notation to be able to show these possibilities.

The Curriculum Council has specified the following standard for Information Systems, and the relationships. Note that it does not support the “zero” option.

**Symbols and characteristics**

| DOCTOR | ENTITIES: An object, person or thing which has attributes. The identifier is typically in uppercase. Optionally, foreign keys can be indicated in the entity-box:
| DOCTOR_NAME | RELATIONSHIPS: These exist between entities, and can be of four forms (or degrees, or cardinality): one-to-one (1:1), one-to-many (1:M, or 1:∞ in Access®), many-to-one (M:1, or ∞:1 in Access®) and many-to-many (M:N). Note that there cannot be M:N relationships in a relational database. In this circumstance, the relations need to be resolved to form two or more 1:M or M:1 relationships.
| treat | The relationship type is written in the diamond in lowercase.
| M | The relationship degree (or cardinality) is written at the extremities if the connectors to the entities.
| DOCTOR_ADDRESS | ATTRAIBUTES: are typically written alongside the entity to which they belong, with the primary key underlined. Foreign keys should have the same identifiers in each entity's attributes.
| DOCTOR_PHONE |
The ERD can be read as saying:
Left to right: ONE student has MANY student-subject combinations.
Right to Left: ONE subject has MANY subject – student combinations.
Normalisation in 3AB will show why this is necessary in a relational database.

Sample ERD
This example is based upon that given in Kendall & Kendall (1992) p. 525
Sample Exam Question Q 40 – ERD; Q41 - DFD

Do the DFD Question first, then attempt the database / ERD question.

Scenario
Spoilt Rott'n Boarding Kennels provides both long term and short term boarding for dogs of all breeds. The business is situated on a 5 acre property in Canning Vale. Prior to the dog’s arrival the owner allocates it to one of the 4 kennel blocks on the property. Each kennel block is capable of housing 20 dogs, and the dogs remain in the same kennel block for the duration of their stay.

Currently the owner of the business, Cathy Barker, uses a spreadsheet to keep track of the bookings. A sample of the data is shown below.

<table>
<thead>
<tr>
<th>OwnerSurname</th>
<th>OwnerFirstname</th>
<th>OwnerAddress</th>
<th>OwnerSuburb</th>
<th>OwnerPhone</th>
<th>DogName</th>
<th>Sex</th>
<th>Breed</th>
<th>Special Diet</th>
<th>Kennel Block</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parker</td>
<td>Min</td>
<td>12 Johnston St</td>
<td>Harris</td>
<td>9476 5643</td>
<td>Beau</td>
<td>M</td>
<td>Poodle</td>
<td>No</td>
<td>Camelot</td>
</tr>
<tr>
<td>Bolt</td>
<td>Sissy</td>
<td>3/2 Hardy Rd</td>
<td>Belmont</td>
<td>9470 1344</td>
<td>Boomer</td>
<td>M</td>
<td>Great Dane</td>
<td>Yes</td>
<td>Camelot</td>
</tr>
<tr>
<td>Clark</td>
<td>Janie</td>
<td>35 Croy St</td>
<td>Cottesloe</td>
<td>9384 3854</td>
<td>Boris</td>
<td>M</td>
<td>Labrador</td>
<td>No</td>
<td>Riverside</td>
</tr>
<tr>
<td>Clarke</td>
<td>Simon</td>
<td>1 James Pl</td>
<td>Como</td>
<td>9450 1686</td>
<td>Beena</td>
<td>F</td>
<td>German Shepherd</td>
<td>Yes</td>
<td>Appleby</td>
</tr>
<tr>
<td>Clarke</td>
<td>Simon</td>
<td>1 James Pl</td>
<td>Como</td>
<td>9451 1686</td>
<td>Heidi</td>
<td>F</td>
<td>German Shepherd</td>
<td>No</td>
<td>Appleby</td>
</tr>
<tr>
<td>Jacks</td>
<td>Lynne</td>
<td>55 Farley Dr</td>
<td>Milletton</td>
<td>9457 4545</td>
<td>Pete</td>
<td>M</td>
<td>Doberman</td>
<td>Yes</td>
<td>Riverside</td>
</tr>
<tr>
<td>Marschel</td>
<td>Con</td>
<td>26 Parsons Av</td>
<td>Manning</td>
<td>9350 7887</td>
<td>Rusty</td>
<td>M</td>
<td>Red Heeler</td>
<td>No</td>
<td>Bankroal</td>
</tr>
<tr>
<td>Evans</td>
<td>Christine</td>
<td>30 Apsley Rd</td>
<td>Milletton</td>
<td>9457 2365</td>
<td>Lucky</td>
<td>F</td>
<td>Maltese</td>
<td>No</td>
<td>Camelot</td>
</tr>
<tr>
<td>Barry</td>
<td>Simon</td>
<td>15 Hope Ave</td>
<td>Manning</td>
<td>9350 1180</td>
<td>Jack</td>
<td>M</td>
<td>Jack Russell</td>
<td>No</td>
<td>Camelot</td>
</tr>
<tr>
<td>Owens</td>
<td>Glenda</td>
<td>3 Wright St</td>
<td>Caversdale</td>
<td>9470 3562</td>
<td>Peaches</td>
<td>F</td>
<td>Shitzu</td>
<td>Yes</td>
<td>Riverside</td>
</tr>
<tr>
<td>Scott</td>
<td>Michael</td>
<td>14 Second Ave</td>
<td>Mt Lawley</td>
<td>9272 1488</td>
<td>Max</td>
<td>M</td>
<td>Blue Heeler</td>
<td>No</td>
<td>Bankroal</td>
</tr>
<tr>
<td>Hurphries</td>
<td>Marcia</td>
<td>255 Burt St</td>
<td>Cottesloe</td>
<td>9364 6222</td>
<td>Tinkerbell</td>
<td>F</td>
<td>Ochnauwahua</td>
<td>Yes</td>
<td>Appleby</td>
</tr>
<tr>
<td>Evans</td>
<td>Christine</td>
<td>30 Apsley Rd</td>
<td>Milletton</td>
<td>9457 2365</td>
<td>Belle</td>
<td>F</td>
<td>Schnauzer</td>
<td>No</td>
<td>Camelot</td>
</tr>
</tbody>
</table>

Question 40  
(28 marks)

(a) Identify two problems that exist with the data table above. Give an example of each problem. 
(4 marks)

Problem 1:

Example:

Problem 2:

Example:
The owners have been advised that they should create a relational database to store the data. It has been suggested that three tables be used: Owner, Dog, KennelBlock.

- An owner may have many dogs
- A dog has only one owner
- A kennel block can hold many dogs
- A dog will only be housed in one kennel block during their stay.

(b) Use the information above to draw an E-R diagram of the structure of the new database. Identify primary keys by underlining them and write FK next to any foreign keys. Ensure you include a list of suitable fields for each entity from the sample data on page 18.

(13 marks)
Question 41  

When a customer rings to enquire about price and availability of kennel stays, they tell the office clerk the number of dogs, the breed and the dates required for the stay. The office clerk uses the price list to prepare a verbal quote for the customer.

If the customer wishes to make a booking, the office clerk asks if there is any special dietary requirements for the dog and enters the booking details into the bookings database.

When the dog(s) arrive at the kennel to begin their stay, the office clerk retrieves the booking details from the bookings database and attaches an ID tag to the dog(s). Any changes to the booking details are updated in the Bookings database. The customer pays the amount owing for the stay and is given an invoice marked 'Paid' as well as a receipt. A copy of the receipt and invoice is placed in the Paid Invoices file.

At the end of each day the office clerk retrieves the invoice and receipt copies from the Paid Invoices file and uses these to reconcile the daily takings. The clerk writes out a deposit slip for the takings and writes the deposit amount into the Accounts book.

(a) Complete the context diagram for the customer service system.  

![Context Diagram](Diagram.png)
(b) Draw the Level 0 data flow diagram for the customer service system. (18 marks)