Data Designs

1.) Data Dictionary

EXAMPLE

? What is a 'Data Dictionary'?

A Data Dictionary is a document which sets out all of the data elements that a system will be required to store or process.

Field Name	Data Type	Data Format	Field Size	Description	Example

ⓘ *Click the link above to download the template* ⓘ

Here some *examples* of typical data items which would be recorded in a Data Dictionary…

Customers

Customer ID, forename, surname, date of birth, email, telephone, address Line 1, address line 2, town/city, postcode

Users

Username, password, forename, surname, date of birth, email, telephone, address Line 1, address line 2, town/city, postcode

Products

Product ID, name, category, no. in stock, colour, size, length, width, weight, volume, price (£), weighting, star-rating

Orders

Order ID, item(s) ordered, quantity, order total (£), date ordered, date shipped, estimated delivery date, delivery notes

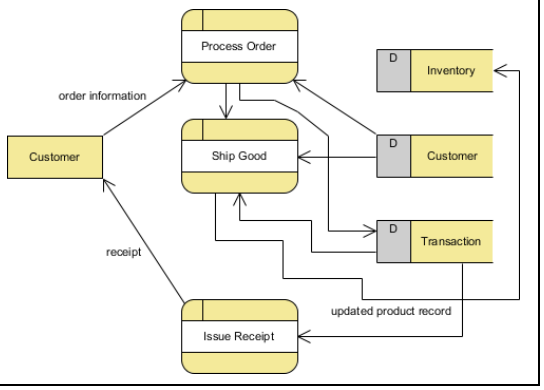
2.) Data Flow Diagrams (DFDs)

EXAMPLE

? What is a 'Data Flow Diagram'?

A Data Flow Diagram represents what data flows into and out of a system.

Like flowcharts, DFDs use specific shapes to represent different things…

Shape	What's it for?	What might a DFD look like?
Process	Represents the procedures performed by the system itself.	
External Entity	Represents people or other systems which send/receive information in/out of the system.	
D Data Store	Represents where data will be stored within the system (e.g. text files, databases tables.	
Flow of Data	Shows the direction in which the data is flowing.	

3.) Entity Relationship Diagrams (ERDs)

EXAMPLE

? What is an 'Entity Relationship Diagram'?

An Entity Relationship Diagram (ERD) is a tool commonly used when designing relational databases. In a relational database, data is stored in the form of multiple inter-related tables. These tables mimic the real-life objects about which data is stored.

When designing a database, ERDs are used to visually represent the tables that might be needed and the type of relationship between each of them.

	What is it?	Examples
Entities	Entities represent objects from the real world about which data will be stored.	**CUSTOMERS** **PRODUCTS** **ORDERS**
Attributes	Attributes are the specific items of data which need to be stored about each Entity. *You should have already worked these out when you created your Data Dictionary!*	***Customer ID*** ***Forename*** ***Surname***

4.) Class Diagram(s)   EXAMPLE

? What is a 'Class Diagram'?

Class diagrams are used to design a software program based on Object-Oriented Programming (OOP) principles.

Basic principles of OOP

Classes & Objects

Unlike procedural programming, where the code you write runs in sequence from top to bottom, OOP techniques involve you creating **Classes** and **Objects:**

	What is it?	Example(s)
Class	A blueprint/template for designing objects	***Person***
Object	An actual instance of an object (based on a Class)	***Person 1*** | ***Person 2*** | ***Person 3***

Attributes & Methods

When creating a **Class**, you outline what **Attributes** and **Behaviours** the objects in that **Class** will have:

	What is it?	Examples
Attributes (Properties)	The information stored about a particular object. *You can think of these like empty fields in a database table.*	The person's ***Name***, ***Age***, ***Gender***, ***Height***, ***Weight*** etc.
Methods (Behaviors)	What actions the object can perform? What can be done with the object?	A person might be able to ***Walk***, ***Run***, ***Eat***, ***Sleep***, ***Speak*** etc.

Creating instances of Objects ('Instantiation')

The table below illustrates 3 objects being created (***Person 1, Person 2, Person 3***). All objects belong to the ***Person*** class.

Attributes:

Every object belonging to the person class will have the same type of attributes (a ***Name***, an ***Age***, a **Gender** etc.).
The exact details of these are chosen when an instance of an object is created, based on the class.

	Person Class
Name	
Age	
Gender	
Height	
Weight	

Person class is the empty template

	Person 1	*Person 2*	*Person 3*
Name	Alice Smith	Paige Turner	Ben Dover
Age	35	42	58
Gender	Female	Male	Male
Height	1.71m	1.59m	1.76
Weight	60kg	55kg	72kg

Actual instances of objects

Methods:

`.sort()`	`.upper()`	`.lower()`	`.randint()`	`.round()`

↑ You probably recognise some of these methods from Python. ↑
Some clever person had to write the code to control how these operate.

OOP is a bit like creating your own programming language. You can code your own methods for your objects!
It's not as hard as it sounds. The methods you code are written inside functions.

Take a look at examples on the *Python Cheat Sheet*