Component 1: Data Organisation

Key terminology

Term	Definition
Character	A letter, digit, space, punctuation mark or various other symbols.
Character set	A table that maps a character with a unique binary number.
Data structure	A specific way of organising data within memory so it can be processed efficiently.
Record	A data structure of related data of different data types.
Primary key	A unique identifier for each record.
Array	A data structure that can hold a fixed number of data items, which must be of the same data type.

Storage of characters

Characters are stored on a computer system as a binary number using a character set. Examples of character sets include ASCII and Unicode.

A small part of the ASCII character set:

Denary	Binary	Hex	Character
64	1000000	40	@
65	1000001	41	A
66	1000010	42	B
67	1000011	43	C

Character sets allow for meaningful data to be exchanged between different computer systems.

Data types

Data type	Description	Examples
Integer	Whole numbers, positive or negative	42, -11, 0
Real	Numbers, including fractions or decimal points	12.9, -17.50, 28.0
Boolean	True or false	1 or 0 `TRUE or FALSE`
Character	Letter, digit, space, punctuation mark or various other symbols	'A', 'b', '7','?'
String	A sequence of characters	‘Computer science’

Data structures

Records example

ID*	First name	Surname	Gender	Date of birth
1074	Sara	Davies	F	12/07/2004
1080	Mike	Thomas	M	31/07/1962
1093	Susan	Jones	F	16/08/1958
1123	Dianna	Glanville	F	19/07/1950
1237	Ahmed	Mushtaq	M	21/02/1973

- ID is the primary key field
- Four fields
- Five records
- Different data types.

One-dimensional array example

[0]	[1]	[2]	[3]	[4]	[5]	[6]	[7]
37	11	42	6	26	56	4	76

- 8 elements
- The index always starts at position `[0]`
- Each element can be accessed using its index
- The element at index `[4]` is 26.

Traversing

Print the contents of the array above:

```
1 for i = 0 to 7
2     output myArray[i]
3 next i
```

Insertion

Add data to an element at a given index:

```
1 myArray[4] = 67
```

This would store the value 67 at index `[4]` of the array.

Deletion

Deleting data from an element at a given index:

```
1 myArray[6] = “”
```

This would leave the memory at index 6 blank.

Searching – arrays can be searched using the index or the value stored at the index.