

Design documentation: test plans

Testing is an important part of the development process as it is essential that you are reasonably sure the database works before people begin to use it.

What do you need to test?

There are a number of things you need to test:

- the validation applied to the database tables (**data testing**)
- the database forms and user interface
- the queries and reports
- that all the required **functionality** works as it should
- that the **usability** and **accessibility** of the user interface is appropriate. For example, is your choice of colours on forms suitable for people with visual impairments?

Create your test plans at the design stage, when you decide what kind of result for each test you are expecting to see when the database is completed.

Data testing

Make sure your test plans include tests of the validation applied to individual fields in a table. For each field, select test data which fits into the following categories:

- 1 Normal data** – values which would normally be expected and which should be accepted.
- 2 Extreme data** – values at the bounds of what is valid. For example, if a field should only accept values in the range 10 to 20, extreme data would be 9 and 21 (both of which should be rejected), 10 and 20 (both of which should be accepted).
- 3 Erroneous data** – data which is completely incorrect such as text entries in numeric fields.

What does a test plan look like?

Create your data test plans in conjunction with your data dictionary. Here's an example of the data dictionary entry for a field in a Cars table.

Table Name:	Cars
Field name	Attributes
No_of_doors	Integer, must be between 2 and 5

This column will be completed when the database is actually tested.

Here are the test plan entries for that field:

Tests 4 and 5 also use extreme data – on the borderline of values which should be rejected.

Test no.	Table, field	Input data	Expected outcome	Actual outcome
1	Cars, No_of_doors	2	Accepted	
2	Cars, No_of_doors	5	Accepted	
3	Cars, No_of_doors	4	Accepted	
4	Cars, No_of_doors	1	Rejected	
5	Cars, No_of_doors	6	Rejected	
6	Cars, No_of_doors	Two	Rejected	

Tests 1 and 2 use extreme data – on the borderline of the acceptable values.

Test 6 uses erroneous data – text values are not allowed in a numeric field.

Links To revise test plans for other aspects of your database, see page 77.

Now try this

Imagine you have been asked to develop a database with a table of BTEC course units. This has fields for Unit number, Unit name, GLH, Type of assessment, Type of unit. Create a data dictionary and a test plan for this table.

GLH can only be 60, 90 or 120.
Type of assessment is Internal or External.
Type of unit is Mandatory or Optional.