Database Programming

Testing

Objectives

This lesson covers the following objectives:

- Develop and apply a strategy for testing that a database functions as designed

Purpose

Most people, when they buy a car, wish to know that it is reliable and will not break down.

So the manufacturers will put the car through a number of tests before it is available to be sold.

The same is true of a database; before it is sold to a customer, it is tested to verify that it meets the business requirements.

Unit Testing

If two things are tested at once and the test fails, it is difficult or impossible to work out what has caused the failure. So it is important to test only one thing at a time. This is commonly referred to as unit testing.

What Could Be Tested?

When testing a database, a variety of things need to be tested. For example:

- Columns should be tested that they contain the correct data type.

- Columns should be tested that they can accommodate the largest amount of data that might be entered.

- Constraints should be checked that they only constrain or limit data that they are supposed to—no more and no less.

What Should Be Tested?

It is frequently unrealistic to test every column and every constraint in every table in a database if it is a large database. A random spread of tests, that check some columns and some constraints, should be carried out.

Designing Tests

Before you carry out a test, you should have a good idea of what result you expect to see if the database is working as expected. This should be documented before you carry out the test in a table similar to the one shown:

Test Number	Date	Test Description	Input	Expected Output	Result/ Discrepancy	Action
22	19/07/06	Confirm NOT NULL constraint on JOB_TITLE in JOBS table	INSERT INTO jobs (job_id, job_title, min_salary, max_salary) VALUES (222,NULL,100,200)			

Running Tests

Once you have designed your test, you can run it and record your results.

Test Number	Date	Test Description	Input	Expected Output	Result/ Discrepancy	Action
22	19/07/06	Confirm NOT NULL constraint on JOB_TITLE in JOBS table	INSERT INTO jobs (job_id, job_title, min_salary, max_salary) VALUES (222,NULL,100,200)	Cannot insert NULL…	Cannot insert NULL…	None

Summary

In this lesson, you should have learned how to:

- Develop and apply a strategy for testing that a database functions as designed