# Delphi Coding Test

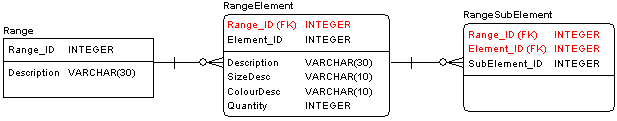
This test is designed to gain an insight into your ability to write Delphi code and your approach to problem solving and communication/collaboration skills. There is no “correct” answer and often the approach is more important than the resulting code. We will be looking for your ability to design a *simple* and *elegant* solution.

**Instructions**

Please read the example below, which is a simplified technical specification of one of our internal applications. Data is stored in SQL Server tables. A meaningful subset is retrieved into memory for manipulation, then rewritten back to the database.

A “Range” is a collection of elements that define products that are sold in stores over a time period. A “Range Element” can be an individual product or a collection of other Range elements. Since Range Elements are stored in a single database table the references described below can be (and often are) *recursive*.

ER diagram showing 3 relevant tables:



Examples of records are:

RangeElement: element\_id = 10000

element\_id = 10001

element\_id = 10002

element\_id = 20000

element\_id = 30000

RangeSubElement: element\_id = 10000, subelement\_id = 10001

element\_id = 10000, subelement\_id = 10002

element\_id = 30000, subelement\_id = 10000

element\_id = 30000, subelement\_id = 20000

The above example all have Range\_ID = 1.

Consider these as the definition of a tree view with [30000] as the parent having [10000] & [20000] as children.

**Write Delphi code targeting 32-bit VCL Windows to:**

1. Define classes to manage the above information in memory, i.e. no access to DB apart from point 3 below. Hint: there is no need to define a separate data object for the “RangeSubElement” because it is only a relationship.
2. Implement a method for RangeElement object to get the total planned quantity. Also a method to get total planned quantity for a Range.
3. Implement LoadFromDB and SaveToDB methods as *appropriate* for the above classes.

As a guideline, it is expected that you shouldn’t need to spend more than one hour on the coding of a solution to this test – but feel free to spend as long as you want.

**Outputs**

Please provide all source code created (including Delphi project), hosted in GitHub, with instructions on how to compile the code. Also provide all related documentation and any supporting information regarding the design and implementation of your code. You can assume that the database tables mentioned above already exists.

**Help / Queries**

You are encouraged to ask questions if any information regarding this test is unclear. Feel free to reach out to us at dennis.chuah@thewarehouse.co.nz Mon-Fri 8:00-16:00 NZ time.

If you do not have Delphi, you can download an evaluation version of Delphi from Embarcadero to help you with this test.