Graduate Test Brief

# Requirements Overview

We require you to write a “C# Console Application” for a basic Customer Relationship Manager (CRM) application.

As Pinewood write software that is used for Dealership Management in the Automotive industry, this CRM will be targeted towards this scenario.

# Requirements Analysis

You have been provided with a text document containing customer and vehicle information. This document will need to be imported into the application, processed and then stored in such a way that various reports can be run on this data.

## Data

### Customer Information

* Forename
* Surname
* Date of Birth

### Vehicle Information

* Manufacturer
* Model
* Registration number
* Registration date
* Engine size (in cc)
* Owner
* InteriorColour (Car only)
* Has Helmet Storage (Motorcycle only)

### Required Relationships

* Customers can have 1 to many vehicles.
* Vehicle must have exactly one owner.
* Vehicle type cannot be changed once it is created.

## Reports

* We will require reports to be designed to contain:
  + All known customers and any vehicles they own.
  + All customers between the age of 20 and 30.
  + All Vehicles registered before 1st January 2010.
  + All Vehicles with an engine size over 1100.

## Expected demonstrable skills

* Good understanding of C#.
* Good understanding of Object Orientated (OO) principles.
* Good understanding of relational data principles.
* Reusable code – low coupling, high cohesion.

## Important Considerations before you begin

There are two supported types of vehicle required in this example; cars and motorcycles.

As this is to be produced in a console application, no GUI is expected or required. The data imported from the CSV does not need to be persisted outside of the runtime of the application; no need to create a database.

If you wish to produce a menu system that allows different reports to be run independently then you can but this is not required.

All reports must be either accessed through a menu, or automatically run upon execution of the code.

We expect that this test will last around 2 – 3 hours but you can take longer or shorter if you wish. If you run out of time then please feel free to include notes that detail what you would do next, or what you would have liked to have done given more time.

Good luck.