# Example 03 - Select by Example

This example shows how to select rows from tables using the out-of-the-box Select by Example functionality. This functionality is available on all database tables (and views) even if they don't have a PK.

## How to Run this example

The **Example 03** is included in the download package. To run this example please refer to the section How to Run the Examples above.

## Case #1: Filtering by a single column

To select using a single column you must instantiate a new DAO to use as the example and set the desired value on the property you want to filter by. Leave the rest of the properties untouched.

This is demonstrated selecting from the table VEHICLE all the rows with type 'CAR'.

## Case #2: Filtering by multiple columns

To select using multiple columns you must instantiate a new DAO to use as the example and set the desired values on the properties you want to filter by. An AND logic condition is used to filter by all the columns. Leave the rest of the properties untouched.

This is demonstrated selecting from the table VEHICLE all the rows with BRAND 'Toyota' and MODEL 'Tercel'.

## Case #3: Using null values

To select using null values you must explicitly set the null value into the properties(s) you want to filter by. An AND logic condition is used to filter by all the columns. Leave the rest of the properties untouched.

This is demonstrated selecting from the table VEHICLE all the rows with BRAND 'Toyota' and MODEL 'Tercel', with null values on the BRANCH\_ID column.

## Case #4: Adding ordering

To specify the order of the returned columns you can add extra parameters when calling the selectByExample() method.

This is demonstrated selecting from the table VEHICLE by adding values from the generated VehicleDAOOrderBy enum.

## Case #5: Using case-insensitive ordering

Case-insensitive ordering is available for String-like columns. The generated enum for the DAO includes a variant for case-insensitivity.

This is demonstrated selecting from the table VEHICLE by adding values from the generated VehicleDAOOrderBy enum.

## Case #6: Using stable case-insensitive ordering

Case-insensitive ordering is available for String-like columns. The generated enum for the DAO includes a variant for stable case-insensitivity. This variant sorts by the case-insensitive value, and additionally sort by plain value.

This is demonstrated selecting from the table VEHICLE by adding values from the generated VehicleDAOOrderBy enum.