

Toelichting technische details

INLINE QUERY & STORED PROCEDURE
FILIP ELEN

Inline query

De inline query is te vinden in de ImageRepository in garage-app-dal.

```
/// <summary>
/// get the images from the product.
/// </summary>
/// <param name="productId"></param>
/// <returns></returns>
1 reference | FilipElen, 5 hours ago | 1 author, 3 changes
public List<Image> GetImagesByProductId(int productId)
{
    return _context.Images.SqlQuery("SELECT * FROM garagedb.images WHERE ProductId = productId", new MySqlParameter("productId", productId)).ToList();
}
```

Deze query haalt al de records op in de image tabel in de database waarvan de productId overeenkomt met degene die wordt doorgegeven aan deze functie.

Stored procedure

De stored procedures zijn de laatste twee functies in de product repository.

```
1 reference | FilipElen, 10 days ago | 1 author, 2 changes
public List<Product> FilterProductBasedOnCategories(string[] categories)
{
    string categoriesAsString = "";
    foreach (string category in categories)
    {
        categoriesAsString += category + ",";
    }

    object[] storedProcedureParams =
    {
        new MySqlParameter("@arrayCategories", categoriesAsString),
        new MySqlParameter("@amountOfCategories", categories.Length)
    };
    return _context.Database
        .SqlQuery<Product>("filterProductsBasedOnCategories(@arrayCategories,@amountOfCategories)",
            storedProcedureParams).ToList();
}

1 reference | FilipElen, 10 days ago | 1 author, 2 changes
public List<Product> FilterProductBasedOnNames(string[] names)
{
    string namesAsString = "";
    foreach (string name in names)
    {
        namesAsString += name + ",";
    }

    object[] storedProcedureParams =
    {
        new MySqlParameter("@arrayNames", namesAsString), new MySqlParameter("@amountOfNames", names.Length)
    };
    return _context.Database
        .SqlQuery<Product>("filterProductsBasedOnNames(@arrayNames,@amountOfNames)", storedProcedureParams)
        .ToList();
}
```

Deze functies roepen de stored procedures aan die de database gaan filteren op producten met meegeven categorieën of de meegeven namen.

De stored procedures sturen alleen maar records terug die aan alle requirements voldoen. Dus als er meerdere categorieën worden meegegeven is het niet genoeg dat een product maar een van die categorieën bevat.

Hieronder vindt u een foto van het script dat werkt gebruikt om de stored procedures aan te maken. Het is niet nodig om dit script te runnen bij het aanmaken van de database. Dit script zit verwerkt in het `garage-app-database.sql` script.

```

1 • USE garagedb;
2 • DROP PROCEDURE IF EXISTS filterProductsBasedOnCategories;
3 DELIMITER //
4 • CREATE PROCEDURE `filterProductsBasedOnCategories`(IN arrayCategories VARCHAR(100), IN amountOfCategories int(16))
5 BEGIN
6     select products.Id, products.Name, products.Price, products.Stock from productcategories
7     inner join categories on productcategories.Category_Id = categories.Id
8     inner join products on products.Id = productcategories.Product_Id
9     WHERE FIND_IN_SET(categories.Type, arrayCategories)
10    Group By productcategories.Product_Id
11    HAVING count(distinct categories.Type) = amountOfCategories;
12    #select Array_String;
13 END//
14 DELIMITER ;
15
16 • DROP PROCEDURE IF EXISTS filterProductsBasedOnNames;
17 DELIMITER //
18 • CREATE PROCEDURE `filterProductsBasedOnNames`(IN arrayNames VARCHAR(100), IN amountOfNames int(16))
19 BEGIN
20     select products.Id, products.Name, products.Price, products.Stock from productcategories
21     inner join categories on productcategories.Category_Id = categories.Id
22     inner join products on products.Id = productcategories.Product_Id
23     WHERE FIND_IN_SET(products.Name, arrayNames)
24     Group By productcategories.Product_Id
25     HAVING count(distinct products.Name) = amountOfNames;
26    #select Array_String;
27 END//
28 DELIMITER ;

```