SQL statement om de properties "eatenby" en "eating" toe te voegen bij het inserten van een organisme.

// aanmaken van het organisme object waarvan de common name en de List properties worden gevuld.

*Organism organism = new Organism();*

*organism.setCommon\_Name("Vogelspin")*

*organism.addToEatingList(organism1, organism2, organism3)*

*organism.addToEatenByList(organism4,organism5,organism6)*

// het inserten van het organisme object in de databank.

*"INSERT INTO organism\_tbl (Column\_commoname) VALUE (organism.getCommon\_Name)"*

// Het inserten van records in de relatietabel "Food". De relaties geven hier aan welke andere organisme het toegevoegde organisme **EET**.

*foreach (org in organism.getEatingList())*

*{*

*"INSERT INTO food\_tbl (Column\_eating, Column\_eatenby) VALUE (org.getId, organism.getId)*

*}*

// Het inserten van records in de relatietabel "Food". De relaties geven hier aan **DOOR** welke andere organisme het toegevoegde organisme **GEGETEN WORDT**.

*foreach (org in organism.getEatenByList())*

*{*

*"INSERT INTO food\_tbl (Column\_eatenby, Column\_eating) VALUE (org.getId, organism.getId)*

*}*

// In het geval dat beide Lists leeg zijn wordt er gekeken naar de property String “FoodName" en String “FoodDescription" van het toegevoegde organisme object.

*Organism organism = new Organism();*

*organism.setCommon\_Name("Vogelspin")*

*organism.setFoodName("Vuilnis")*

*organism.setFoodDescription("Het organisme eet vooral organische etensresten")*

// het inserten van het organisme object in de databank.

*"INSERT INTO organism\_tbl (Column\_commoname) VALUE (organism.getCommon\_Name)"*

// Het inserten van records in de relatietabel "Food". De relatie geeft hier aan WAT het toegevoegde organisme **EET** aan de hand van een **NAAM** en **BESCHRIJVING**.

*"INSERT INTO food\_tbl (Column\_eatenby, Column\_food\_name, Column\_food\_description) VALUE (organism.getId(), organism.getFoodName, organism.getFoodDescription)”*