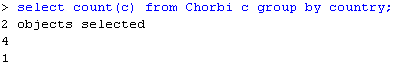
**C.1.1 - A listing with the number of chorbies per country.**

Seleccionamos el número de Chorbies que hay por cada país.

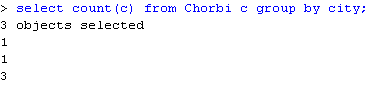
*select count(c) from Chorbi c group by country;*



**C.1.2 - A listing with the number of chorbies per city.**

Seleccionamos en número de Chorbies que hay por cada ciudad.

*select count(c) from Chorbi c group by city;*



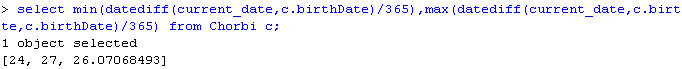
**C.2 - The minimum, the maximum, and the average ages of the chorbies.**

Calculamos la diferencia que hay entre la fecha actual y la fecha de nacimiento de un Chorbie y lo normalizamos a años y seleccionamos el valor mínimo, máximo y la media.

*select min(datediff(current\_date,c.birthDate)/365),*

*max(datediff(current\_date,c.birthDate)/365),*

*avg(datediff(current\_date,c.birthDate)/365) from Chorbi c;*



**C.3 - The ratio of chorbies who have not registered a credit card or have registered an invalid credit card.**

Contamos cuantos Chorbies tienen CreditCards nulas o no tienen y lo dividimos entre el total de Chorbies.

*select 1.0\*(select count(c) from Chorbi c where c.creditCard=null)/ count(c2) from Chorbi c2;*



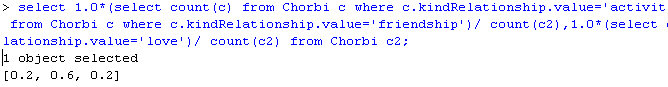
**C.4 - The ratios of chorbies who search for “activities”, “friendship”, and “love”.**

Calculamos el número de Chorbies que tienen como KindRelationship igual a activities, a friendship y a love y lo dividimos cada uno por el total de Chorbies.

*select 1.0\*(select count(c) from Chorbi c where c.kindRelationship.value='activities')/ count(c2),*

*1.0\*(select count(c) from Chorbi c where c.kindRelationship.value='friendship')/ count(c2),*

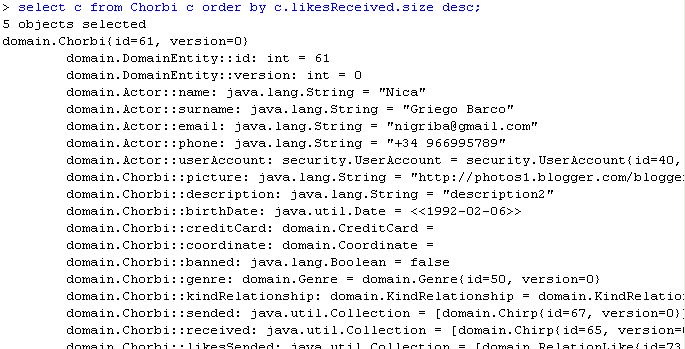
*1.0\*(select count(c) from Chorbi c where c.kindRelationship.value='love')/ count(c2) from Chorbi c2;*



**B.1 - The list of chorbies, sorted by the number of likes they have got.**

Listamos los Chorbies ordenados por el atributo likesReceived que nos dice cuántos likes tienen los Chorbies.

*select c from Chorbi c order by c.likesReceived.size desc;*



**B.2 - The minimum, the maximum, and the average number of likes per chorbi.**

Seleccionamos el tamaño mínimo, máximo y la media del atributo likeReceived que contiene los likes que tiene un Chorbi.

*select min(c.likesReceived.size), max(c.likesReceived.size), avg(c.likesReceived.size) from Chorbi c;*



**A.1 - The minimum, the maximum, and the average number of chirps that a chorbi receives from other chorbies.**

Seleccionamos el tamaño mínimo, máximo y medio del atributo recieved que nos dice el número de chirps recibidos por Chorbi.

*select min(c.received.size), max(c.received.size), avg(c.received.size) from Chorbi c;*



**A.2 - The minimum, the maximum, and the average number of chirps that a chorbi sends to other chorbies.**

Seleccionamos el tamaño mínimo, máximo y medio del atributo sended que nos dice el número de chirps enviados por Chorbi.

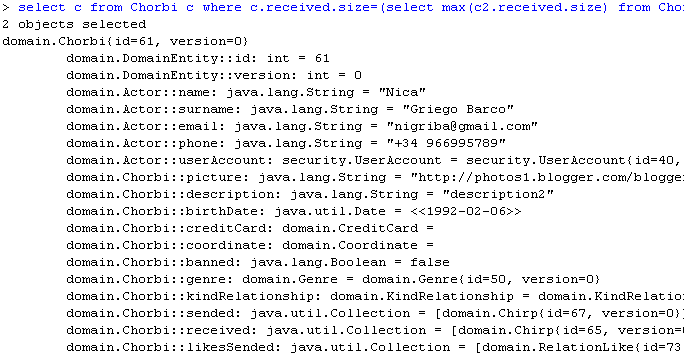
*select min(c.sended.size), max(c.sended.size), avg(c.sended.size) from Chorbi c;*



**A.3 - The chorbies who have got more chirps.**

Vamos a devolver una lista con uno o más Chorbies, seleccionando aquellos/as que han recibido más chirps viendo el atributo received, que nos indica el número de chirps recibidos.

*select c from Chorbi c where c.received.size=(select max(c2.received.size) from Chorbi c2);*



**A.4 - The chorbies who have sent more chirps.**

Vamos a devolver una lista con uno o más Chorbies, seleccionando aquellos/as que han enviado más chirps viendo el atributo sended, que nos indica el número de chirps enviados.

*select c from Chorbi c where c.sended.size=(select max(c2.sended.size) from Chorbi c2);*

