a. Dar los nombres de los aeropuertos por los cuales hayan pasado pasajeros de nacionalidad japonesa y de sexo femenino.

El primer select trae los aeropuertos desde los cuales despegaron pasajeras japonesas, y el segundo select trae los aeropuertos donde aterrizaron.

b. ¿A cuántos kilómetros de su casco urbano se encuentran los aeropuertos por los cuales despegaron vuelos sin plazas vacías (completos) y que pertenecen a programas de vuelo de la aerolínea "Aerolineas Argentinas"?

Primero se agrupan los pasajeros por vuelo, y se usa la función agregada COUNT para saber la cantidad total de pasajeros por vuelo. Luego se filtran los vuelos donde ese total es igual a la capacidad máxima del modelo de avión usado.

```
SELECT count(*) as total_pasajeros, m.capacidad, pavu.id_vuelo

FROM (((((pasajeros_vuelos pavu

INNER JOIN vuelos v ON pavu.id_vuelo = v.id)

INNER JOIN programas_de_vuelo pv ON pv.id_programa = v.id_programa)

INNER JOIN aviones a ON v.id_avion = a.id_avion)

INNER JOIN modelos m ON m.id_modelo = a.id_modelo)

INNER JOIN aeropuertos ap ON ap.codigo_internacional = pv.origen)

WHERE pv.id_aerolinea = 'Aerolineas Argentinas'

GROUP BY pavu.id_vuelo, m.capacidad

HAVING count(*) = m.capacidad
```

c. Determinar la cantidad de pistas que existen por provincia.

```
SELECT COUNT(1) AS nro_pistas, a.provincia
    FROM pistas p
    INNER JOIN aeropuertos a ON p.id_aeropuerto = a.codigo_internacional
    GROUP BY a.provincia
```

d. Determinar la cantidad de controles que se le hayan efectuado a los aviones modelo Boeing 747 durante el 2013-2014, para aquellos aviones que hayan obtenido un puntaje promedio mayor a 7.

Primero se seleccionan los controles filtrando por modelo de avion. Luego se agrupa por tipo de test, y para cada grupo de tests se cuentan los controles realizados, filtrando por promedio.

e. Determinar cuáles fueron los aeropuertos que hayan obtenido el peor puntaje entre los controles que se realizaron durante el año 2014 a los aviones modelo Boeing 747

Primero (líneas 07-11) se encuentran los peores puntajes por aeropuerto, para los aviones del modelo requerido. Luego (línea 6) se selecciona el mínimo de todos esos mínimos, es decir, el peor puntaje absoluto. Luego (líneas 1-6) se agrupan los controles por aeropuerto, y nos quedamos con los aeropuertos cuyo peor puntaje sea igual al mínimo absoluto encontrado anteriormente.

```
1 SELECT MIN(c.puntaje) as peor puntaje, c.id aeropuerto AS aeropuerto
2 FROM (controles c
    INNER JOIN aviones a ON c.id avion = a.id avion AND c.id modelo = a.id modelo)
    WHERE a.id modelo = 747 AND YEAR(c.fecha) = 2014
    GROUP BY c.id_aeropuerto
HAVING MIN(c.puntaje) = (SELECT MIN(puntajes.peores) FROM
7
                                  (SELECT MIN(c.puntaje) AS peores
8
                                         FROM (controles c
                                         INNER JOIN aviones a ON c.id avion =
a.id avion
                                                             AND c.id modelo =
a.id modelo)
                                         WHERE a.id modelo = 747 AND YEAR(c.fecha) =
10
2014
11
                                         GROUP BY c.id aeropuerto) puntajes)
```

f. Pensar 3 consultas

f.i. Encontrar a los pasajeros que hayan hecho vuelos inter-provinciales (para ofrecerles descuento por viajeros frecuentes).

```
SELECT pavu.id_pasajero, p.nombre

FROM (((((pasajeros_vuelos pavu
INNER JOIN pasajeros p ON p.id = pavu.id_pasajero)
INNER JOIN vuelos v ON pavu.id_vuelo = v.id)
INNER JOIN programas_de_vuelo pv ON pv.id_programa = v.id_programa)
INNER JOIN aeropuertos a_origen ON a_origen.codigo_internacional = pv.origen)
INNER JOIN aeropuertos a_destino ON a_destino.codigo_internacional =

pv.destino)

WHERE a_origen.provincia <> a_destino.provincia
GROUP BY pavu.id pasajero, p.nombre
```

f.ii. Armar un ranking de los modelos de avión que pesen menos de 200 toneladas y que más se hayan usado en vuelos realizados durante el año pasado.

```
SELECT COUNT(*) AS cantidad_de_vuelos, a.id_modelo AS modelo, a.id_avion AS avion
FROM ((vuelos v
INNER JOIN aviones a ON v.id_avion = a.id_avion AND v.id_modelo = a.id_modelo)
INNER JOIN modelos m ON m.id_modelo = a.id_modelo)
WHERE m.peso_toneladas <= 200
GROUP BY a.id_avion, a.id_modelo
ORDER BY COUNT(*) DESC
```

f.iii. Listar los vuelos en los que haya viajado el (o la) más joven de los viajeros más frecuentes.

Primero se cuentan los vuelos por pasajero (líneas 9-12); se encuentra la cantidad máxima de vuelos entre ellos (línea 8); nos quedamos con los pasajeros que tengan la cantidad de vuelos máxima (líneas 5-8); a esos pasajeros los ordenamos por fecha de nacimiento y encontramos la máxima (i.e., el más joven) (líneas 3-13); finalmente, se buscan los vuelos de los pasajeros que tengan esa fecha de nacimiento más reciente (porque podría haber más de uno que haya nacido el mismo día).

```
    SELECT pv.id_vuelo AS vuelo, p.nombre AS viajero
    FROM (pasajeros_vuelos pv INNER JOIN pasajeros p ON p.id = pv.id_pasajero)
    WHERE p.fecha_nacimiento = (select max(p.fecha_nacimiento) from
    (SELECT p.id, p.fecha_nacimiento FROM
    (SELECT COUNT(*) AS cant_vuelos, p.fecha_nacimiento, p.id
```

6.	FROM (pasajeros_vuelos pv INNER JOIN
pasajeros p	
	ON pv.id_pasajero =
p.id)	
7.	GROUP BY p.id, p.fecha_nacimiento
8.	HAVING COUNT(*) = (SELECT
MAX(cant_vuelos) FROM	
9.	(SELECT COUNT(*) AS cant_vuelos, p.id
	FROM
10.	(pasajeros_vuelos pv INNER JOIN
pasajeros	
	p ON pv.id_pasajero =
p.id)	
11.	GROUP BY p.id, p.fecha_nacimiento
12.	ORDER BY COUNT(*) DESC)))
13.	ORDER BY p.fecha nacimiento, p.id DESC))
	-