**(Notes:** Formated queries are at the bottom of this file)

**Question 1:** Which are the top 3 demanding services on board?

**Script:**

*select*

*i.item\_name,*

*sum(s.quantity) as total\_volume*

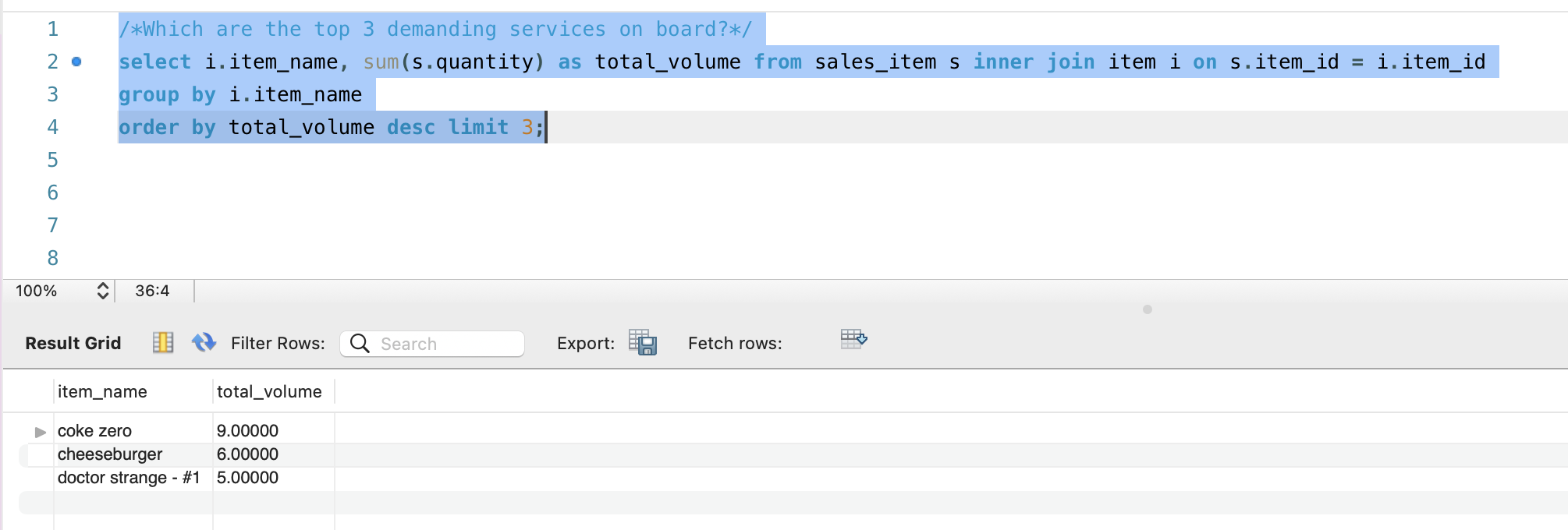
*from sales\_item s*

*inner join item i on s.item\_id = i.item\_id*

*group by i.item\_name*

*order by total\_volume desc limit 3;*

**Output screen:**



**Question 2:** Show the average number of tons of food & drinks needed per day.

**Script:**

*select s.sales\_date\_time, i.seg\_name, (avg(i.weight)/1000) as average\_food\_beverages\_in\_tons from sales\_item si, onboard\_sales s,*

*(select i.item\_id, i.item\_name, i.weight, u.metric\_name, c.cat\_name, cl.class\_name, f.fam\_name, s.seg\_name from item i, unit\_measure u, category c, class cl, family f, segment s*

*where (i.unit\_measure\_id = u.unit\_measure\_id) and*

*(i.category\_id = c.category\_id) and*

*(c.class\_id = cl.class\_id) and*

*(cl.family\_id = f.family\_id) and*

*(f.segment\_id = s.segment\_id) and*

*(i.category\_id in (*

*select category\_id from category where class\_id in (*

*select class\_id from class where family\_id in (*

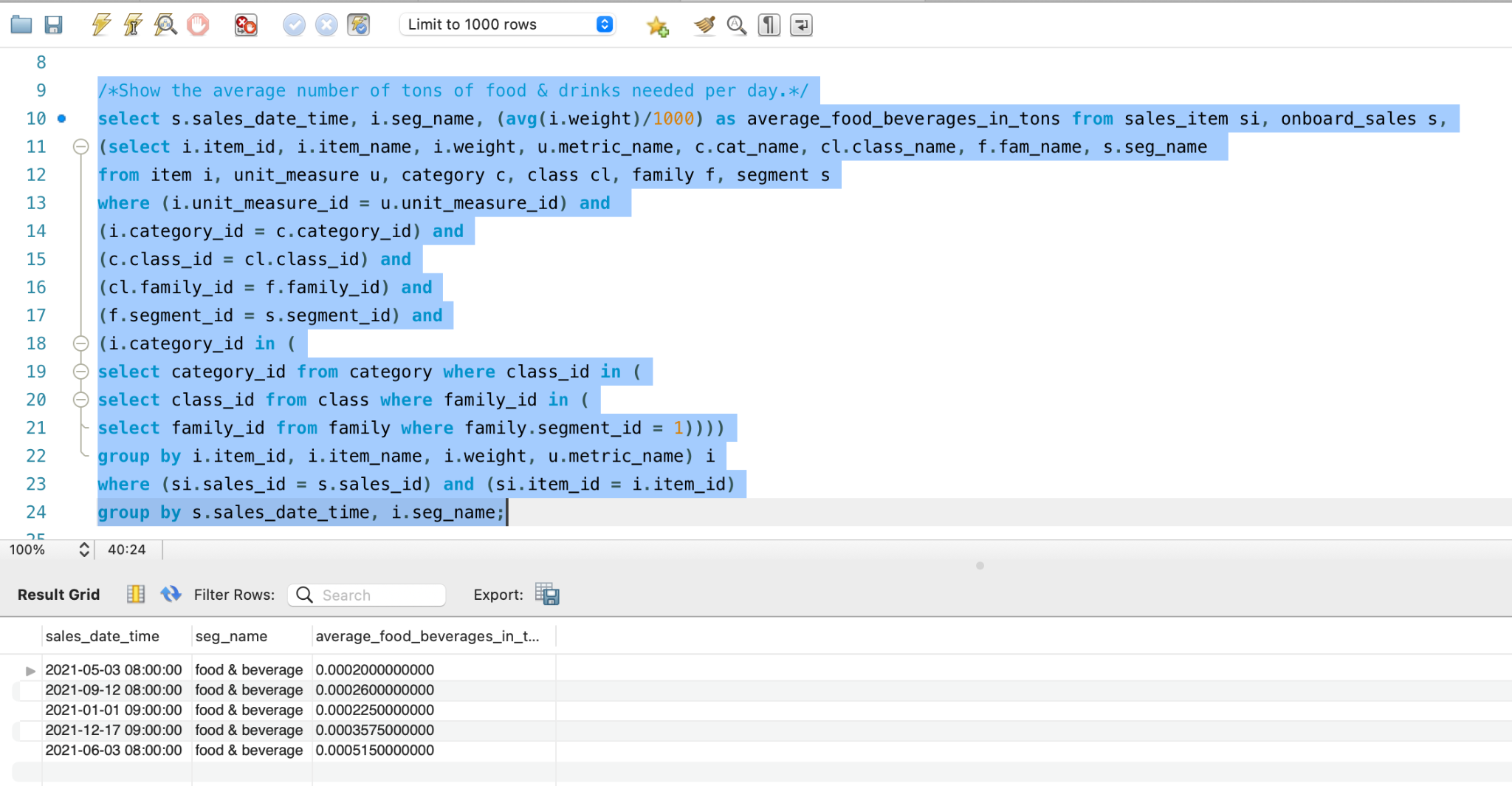
*select family\_id from family where family.segment\_id = 1))))*

*group by i.item\_id, i.item\_name, i.weight, u.metric\_name) i*

*where (si.sales\_id = s.sales\_id) and (si.item\_id = i.item\_id)*

*group by s.sales\_date\_time, i.seg\_name;*

**Output result:**



**Question 3**: How much money will be lost per day, if a ship is under repair?

In order to answer this question, we summed total revenue from tickets and onboard sales and calculated average revenue per day based on actual transaction days, i.e. total revenue divided by the number of days (actual days when transactions occurred).

The result is the average revenue per day and if the ship is in repair, this average revenue will be lost.

**Script:**

*select (sum(t.total) / count(t.revenue\_date)) from*

*(*

*select t.date\_time as revenue\_date, c.currency\_code, sum(t.total\_order) as total from tickets t, currency c*

*where t.currency\_id = c.currency\_id*

*group by t.date\_time, c.currency\_code*

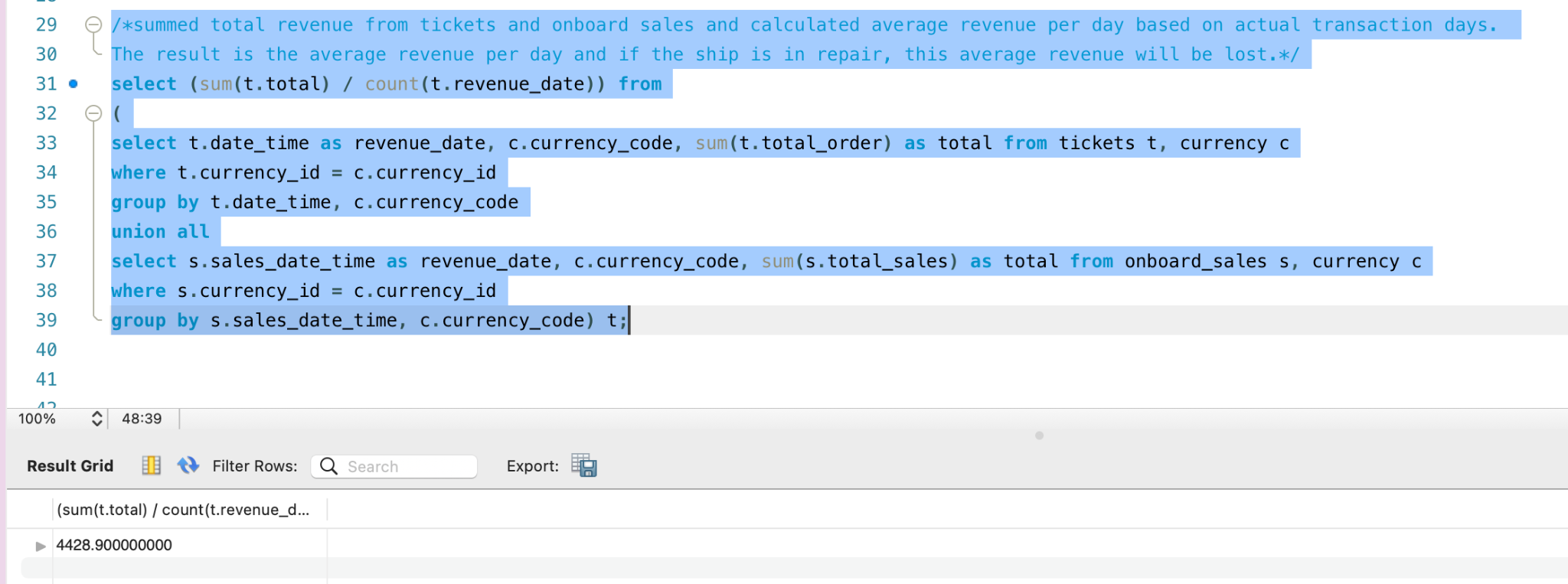
*union all*

*select s.sales\_date\_time as revenue\_date, c.currency\_code, sum(s.total\_sales) as total from onboard\_sales s, currency c*

*where s.currency\_id = c.currency\_id*

*group by s.sales\_date\_time, c.currency\_code) t;*

**Output results:**



**Question 4**: Which is the most profitable cruise route (include season)?

Based on the business model our team built - the total revenue from the cruise consists of route price + price for room in a ship. Hence, to calculate the most profitable cruise, we took the total sales consisting of these two elements (routes + rooms).

**Script:**

*select se.route\_name, se.season\_name, c.currency\_code, sum(t.total\_order) as total\_revenue from tickets t, currency c,*

*(select r.\*, s.season\_name from routes r, seasons s*

*where r.season\_id = s.season\_id) se*

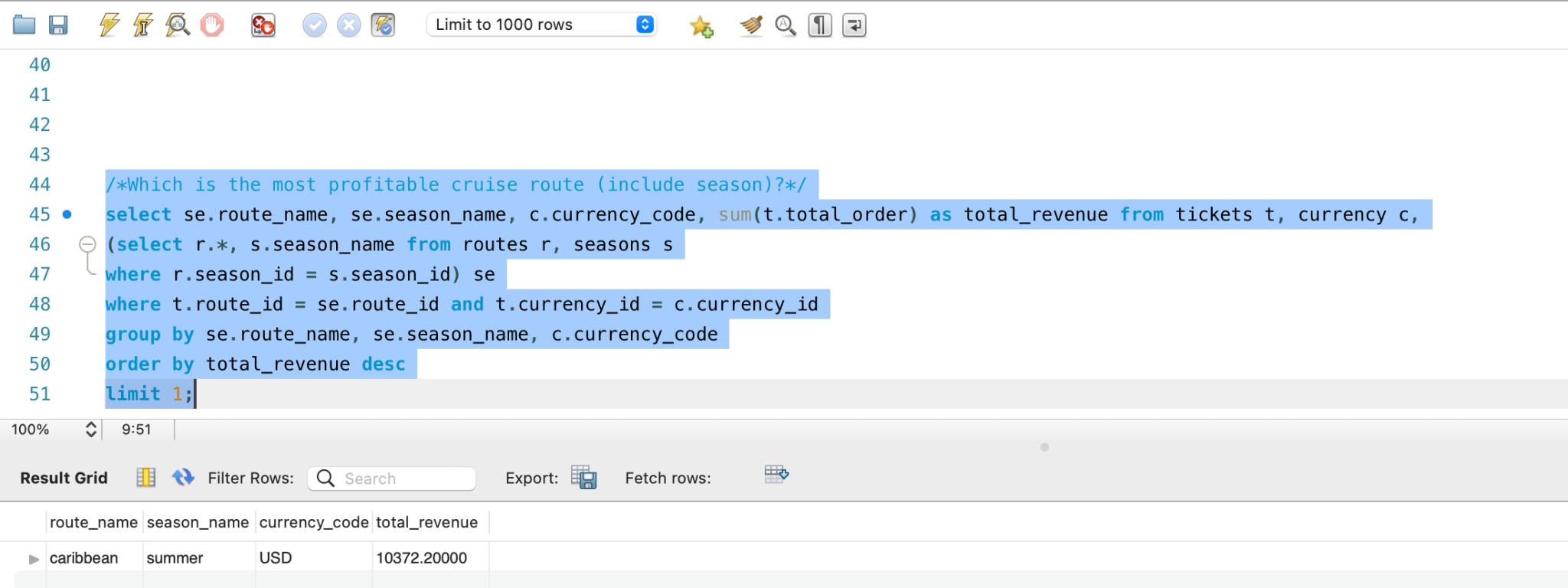
*where t.route\_id = se.route\_id and t.currency\_id = c.currency\_id*

*group by se.route\_name, se.season\_name, c.currency\_code*

*order by total\_revenue desc*

*limit 1;*

**Output results:**

****

**Formatted queries:**

**Q1:**

***select***

***i.item\_name,***

***sum(s.quantity) as total\_volume***

***from sales\_item s***

***inner join item i on s.item\_id = i.item\_id***

***group by i.item\_name***

***order by total\_volume desc limit 3;***

***Q2:***

***select***

***s.sales\_date\_time,***

***i.seg\_name,***

***(avg(i.weight)/1000) as average\_food\_beverages\_in\_tons***

***from***

***sales\_item si,***

***onboard\_sales s,***

***(select i.item\_id,***

***i.item\_name,***

***i.weight,***

***u.metric\_name,***

***s.seg\_name***

***from item i,***

***unit\_measure u,***

***category c,***

***class cl,***

***family f,***

***segment s***

***where (i.unit\_measure\_id = u.unit\_measure\_id) and***

***(i.category\_id = c.category\_id) and***

***(c.class\_id = cl.class\_id) and***

***(cl.family\_id = f.family\_id) and***

***(f.segment\_id = s.segment\_id) and***

***(i.category\_id in***

***(***

***select category\_id from category where class\_id in***

***(***

***select class\_id from class where family\_id in (***

***select family\_id from family where family.segment\_id = 1)***

***)***

***)***

***)***

***group by i.item\_id, i.item\_name, i.weight, u.metric\_name***

***) i***

***where (si.sales\_id = s.sales\_id) and (si.item\_id = i.item\_id)***

***group by s.sales\_date\_time, i.seg\_name;***

***Q3:***

***select***

***(sum(t.total) / count(t.revenue\_date))***

***from***

***(***

***select t.date\_time as revenue\_date,***

***c.currency\_code,***

***sum(t.total\_order) as total***

***from tickets t, currency c***

***where t.currency\_id = c.currency\_id***

***group by t.date\_time, c.currency\_code***

***union all***

***select s.sales\_date\_time as revenue\_date,***

***c.currency\_code,***

***sum(s.total\_sales) as total***

***from onboard\_sales s, currency c***

***where s.currency\_id = c.currency\_id***

***group by s.sales\_date\_time, c.currency\_code***

***) t;***

***Q4:***

***select***

***se.route\_name,***

***se.season\_name,***

***c.currency\_code,***

***sum(t.total\_order) as total\_revenue***

***from tickets t,***

***currency c,***

***(select r.\*, s.season\_name from routes r, seasons s***

***where r.season\_id = s.season\_id) se***

***where t.route\_id = se.route\_id and t.currency\_id = c.currency\_id***

***group by se.route\_name, se.season\_name, c.currency\_code***

***order by total\_revenue desc***

***limit 1;***