WEEK 3 CHALLENGE SOLUTIONS

Below are suggested answers. Note, in some cases, there's more than one way of getting the correct answer so check to see if your codes will produce the same results as the codes below.

1. Find the top 10 customers who paid the most in a single transaction. **Show their customer ID, first name, last name and amount paid.** Sort the results by amount paid (highest on top) and by customer ID.

```
SELECT c.customer_id, first_name, last_name, amount
FROM customer c
JOIN payment p
ON c.customer_id = p.customer_id
ORDER BY 4 DESC, 1
LIMIT 10;
```

2. For the customer named Elizabeth Brown, find all **the amounts she paid, the payment dates as well as the staff's first and last name** who served her on each transaction. Sort the results by payment date (latest date on top).

Also add another column named "Amount > 2.99" which performs mathematical operation (amount > 2.99). It will show either 'true' or 'false' under that column.

> *Hint: The customer's first name is Elizabeth and last name is Brown. Note that both the customer table and staff table have attributes named 'first name' and 'last name' so make sure to use the appropriate table name/alias when referring to these attributes.*

```
SELECT amount, payment_date,
        s.first_name,  s.last_name,
        (amount > 2.99) "Amount > 2.99"
FROM customer c
JOIN payment p
ON c.customer_id = p.customer_id
JOIN staff s
ON p.staff_id = s.staff_id
WHERE c.first_name = 'Elizabeth' AND c.last_name = 'Brown'
ORDER BY 2 DESC;
```

3. Produce a table showing the film ID, title, category name (renamed as category) and inventory ID of films that do not have any inventory.

```
SELECT f.film_id, title, name AS category, inventory_id
FROM category c
JOIN film_category fc
ON c.category_id = fc.category_id
JOIN film f
ON fc.film_id = f.film_id
LEFT JOIN inventory i
ON f.film_id = i.film_id
WHERE inventory_id IS NULL;
```

4. Using either left join or right join clause, produce a table that shows the film ID title, language name (renamed as language) and rental date of film IDs ranging from 144 to 149.

Hint: Make sure to show all the films including those that were not rented out.

Question: How many of these films were not rented out?

Answer: Two were not rented out

```
SELECT f.film_id, title, name AS language, rental_date
FROM language l
JOIN film f
ON l.language_id = f.language_id
LEFT JOIN inventory i
ON f.film_id = i.film_id
LEFT JOIN rental r
ON i.inventory_id = r.inventory_id
WHERE f.film_id BETWEEN 144 AND 149
ORDER BY 1;
```

BONUS Exercise (Optional): Create a table showing the customer ID, email address, city and country of origin of customers. Sort by customer ID and limit your results to the first 10 customers.

Note: Some tables in your query may start with the same letter. Use different aliases.

Question: What's the country of origin of customer ID no. 1?

Answer: Japan

```
SELECT customer_id, email, g.city, country
    FROM customer c
    JOIN address a
    ON c.address_id = a.address_id
    JOIN city g
    ON a.city_id = g.city_id
    JOIN country y
    ON g.country_id = y.country_id
    ORDER BY 1
    LIMIT 10;
```