

Question 1:

```
• SELECT c.name AS category,  
        AVG(f.length) AS average_length  
FROM category c  
JOIN film_category fc ON c.category_id = fc.category_id  
JOIN film f ON fc.film_id = f.film_id  
GROUP BY c.name  
ORDER BY c.name;
```

	category	average_length	
	Action	111.6094	
	Animation	111.0152	
	Children	109.8000	
	Classics	111.6667	
	Comedy	115.8276	
	Documentary	108.7500	
	Drama	120.8387	
	Family	114.7826	
	Foreign	121.6986	
	Games	127.8361	
	Horror	112.4821	
	Music	113.6471	
	New	111.1270	
	Sci-Fi	108.1967	
	Sports	128.2027	
	Travel	113.3158	

Question 2:

```

1 WITH CategoryAverages AS (
2     SELECT c.name AS category,
3           AVG(f.length) AS average_length
4     FROM category c
5     JOIN film_category fc ON c.category_id = fc.category_id
6     JOIN film f ON fc.film_id = f.film_id
7     GROUP BY c.name
8 )
9 SELECT category, average_length
10 FROM CategoryAverages
11 WHERE average_length = (SELECT MAX(average_length) FROM CategoryAverages)
12        OR average_length = (SELECT MIN(average_length) FROM CategoryAverages);
13

```

	category	average_length	
	Sci-Fi	108.1967	
	Sports	128.2027	

Question 3:

```
SELECT c.customer_id, c.first_name, c.last_name
FROM customer c
WHERE EXISTS (
    SELECT 1
    FROM rental r
    JOIN inventory i ON r.inventory_id = i.inventory_id
    JOIN film f ON i.film_id = f.film_id
    JOIN film_category fc ON f.film_id = fc.film_id
    JOIN category cat ON fc.category_id = cat.category_id
    WHERE r.customer_id = c.customer_id
    AND cat.name = 'Action'
)
AND NOT EXISTS (
    SELECT 1
    FROM rental r
    JOIN inventory i ON r.inventory_id = i.inventory_id
    JOIN film f ON i.film_id = f.film_id
    JOIN film_category fc ON f.film_id = fc.film_id
    JOIN category cat ON fc.category_id = cat.category_id
    WHERE r.customer_id = c.customer_id
    AND cat.name IN ('Comedy', 'Classic')
);
```

2	PATRICIA	JOHNSON	
8	SUSAN	WILSON	
11	LISA	ANDERSON	
17	DONNA	THOMPSON	
43	CHRISTINE	ROBERTS	
49	JOYCE	EDWARDS	
60	MILDRED	BAILEY	
69	JUDY	GRAY	
71	KATHY	JAMES	
73	BEVERLY	BROOKS	
83	LOUISE	JENKINS	
90	RUBY	WASHING...	
111	CARMEN	OWENS	
115	WENDY	HARRISON	
123	SHANNON	FREEMAN	
124	SHEILA	WELLS	
126	ELLEN	SIMPSON	
136	ANITA	MORALES	
139	AMBER	DIXON	
142	APRIL	BURNS	
164	JOANN	GARDNER	
171	DOLORES	WAGNER	
178	MARION	SNYDER	
183	IDA	ANDREWS	
185	ROBERTA	HARPER	
203	TARA	RYAN	
212	WILMA	RICHARDS	
212	GINA	WILLIAMS	

Question 4:

```

SELECT a.actor_id, a.first_name, a.last_name, COUNT(*) AS movie_count
FROM actor a
JOIN film_actor fa ON a.actor_id = fa.actor_id
JOIN film f ON fa.film_id = f.film_id
JOIN language l ON f.language_id = l.language_id
WHERE l.name = 'English'
GROUP BY a.actor_id, a.first_name, a.last_name
ORDER BY movie_count DESC
LIMIT 1;

```

	actor_id	first_name	last_name	movie_count	
	107	GINA	DEGENERES	42	

Question 5:

```

SELECT COUNT(DISTINCT r.inventory_id) AS distinct_movies_count
FROM rental r
JOIN inventory i ON r.inventory_id = i.inventory_id
JOIN store s ON i.store_id = s.store_id
JOIN staff st ON s.store_id = st.store_id
WHERE st.first_name = 'Mike'
AND DATEDIFF(r.return_date, r.rental_date) = 10;

```

	distinct_movies_co...	
	63	

Question 6:

```
-- Find the film_id of the movie with the largest cast
SELECT film_id
FROM film_actor
GROUP BY film_id
ORDER BY COUNT(actor_id) DESC
LIMIT 1;

-- List actors who appeared in the movie with the largest cast
SELECT a.actor_id, a.first_name, a.last_name
FROM actor a
JOIN film_actor fa ON a.actor_id = fa.actor_id
WHERE fa.film_id = 508
ORDER BY a.last_name;
```

	actor_id	first_name	last_name	
	47	JULIA	BARRYMORE	
	37	VAL	BOLGER	
	81	SCARLETT	DAMON	
	138	LUCILLE	DEE	
	28	WOODY	HOFFMAN	
	170	MENA	HOPPER	
	45	REESE	KILMER	
	61	CHRISTIAN	NEESON	
	150	JAYNE	NOLTE	
	75	BURT	POSEY	
	53	MENA	TEMPLE	
	102	WALTER	TORN	
	147	FAY	WINSLET	
	111	CAMERON	ZELLWEGER	
	186	JULIA	ZELLWEGER	