

Grading Rubric: Sakila Video — Rentals and Revenue

Schema: sakila | 8 questions | 100 points total

Question Requirements

Q	Pts	Techniques	Required tables	Required fields	Clauses
1	8	select, where	sakila.film	film.title, film.rental_rate, film.rating	WHERE
2	8	select, like, order_by	sakila.customer	customer.first_name, customer.last_name, customer.email	WHERE, ORDER BY
3	8	select, where, order_by	sakila.film	film.title, film.length, film.replacement_cost	WHERE, ORDER BY
4	8	select, where	sakila.staff	staff.first_name, staff.last_name, staff.email	WHERE
5	12	join, group_by, aggregate	sakila.category, sakila.film_category	category.name, film_category.film_id	JOIN, GROUP BY
6	14	join, aggregate, group_by, order_by	sakila.customer, sakila.payment	customer.first_name, customer.last_name, payment.amount	JOIN, GROUP BY, ORDER BY
7	14	join, group_by, aggregate, having	sakila.actor, sakila.film_actor	actor.first_name, actor.last_name, film_actor.film_id	JOIN, GROUP BY
8	28	join, group_by, aggregate, having, order_by	sakila.category, sakila.film_category, sakila.film, sakila.inventory, sakila.rental, sakila.payment	category.name, film.rental_rate, payment.amount	JOIN, GROUP BY, ORDER BY

Deduction Schedule

Each question starts at full credit. The deductions below are applied against that question's point value and cannot take it below zero.

Issue	Deduction
Required table missing from the query	-3 per table
Required field missing from the SELECT list	-2 per field
Required WHERE condition missing or using the wrong operator	-3 per condition
WHERE clause omitted entirely when one is required	-5
JOIN omitted when the question requires one	-5

Issue	Deduction
GROUP BY omitted when the question requires one	-4
ORDER BY omitted when one is required	-4
ORDER BY on the wrong field	-2 per field
ORDER BY in the wrong direction	-1 per field

Reference Solutions

For instructor and TA use. A student answer that differs from the reference but returns the correct result set earns full credit.

Q1 (8 pts)

```
SELECT film.title, film.rental_rate, film.rating FROM sakila.film WHERE film.rental_rate > 3.00 ORDER BY film.rental_rate
```

Q2 (8 pts)

```
SELECT customer.first_name, customer.last_name, customer.email FROM sakila.customer WHERE customer.last_name LIKE 'M'
```

Q3 (8 pts)

```
SELECT film.title, film.length, film.replacement_cost FROM sakila.film WHERE film.length > 120 AND film.replacement_cost > 19.99
```

Q4 (8 pts)

```
SELECT staff.first_name, staff.last_name, staff.email FROM sakila.staff WHERE staff.active = 1 ORDER BY staff.last_name
```

Q5 (12 pts)

```
SELECT category.name, COUNT(film_category.film_id) AS total_films FROM sakila.category JOIN sakila.film_category ON category.category_id = film_category.category_id
```

Q6 (14 pts)

```
SELECT customer.first_name, customer.last_name, SUM(payment.amount) AS total_paid FROM sakila.customer JOIN sakila.payment ON customer.customer_id = payment.customer_id
```

Q7 (14 pts)

```
SELECT actor.first_name, actor.last_name, COUNT(film_actor.film_id) AS total_films FROM sakila.actor JOIN sakila.film_actor ON actor.actor_id = film_actor.actor_id
```

Q8 (28 pts)

```
SELECT category.name, ROUND(AVG(film.rental_rate), 2) AS avg_rental_rate, SUM(payment.amount) AS total_revenue FROM sakila.category JOIN sakila.film ON category.category_id = film.category_id JOIN sakila.payment ON film.film_id = payment.film_id
```