**Introduction**  
This case study focuses on analyzing a DVD Rentals database to extract meaningful insights about customer behavior and overall business performance. By diving into this pre-Netflix era business data, we aim to demonstrate the power of SQL in transforming raw data into actionable intelligence.

**Tools Used for Analysis**  
**-**SQL

**About the Dataset**  
The DVD Rentals database comprises 15 tables:  
- actor  
- address  
- category  
- city  
- country  
- customer  
- film  
- film\_actor  
- film\_category  
- inventory  
- language  
- payment  
- rental  
- staff  
- store

**Entity Relationship Diagram**  
(Include a diagram here if available)

**Case Study Questions and Solutions**

1. **List All Actors in the Database**  
   **Question:**How can I list all actors in the database?

SELECT actor\_id, first\_name, last\_name  
FROM actor;

A screenshot of a computer

AI-generated content may be incorrect.

This query will return all actors’ IDs, first names, and last names.

**2. Find Movies Starring a Specific Actor**  
**Question:** How can I find all movies starring a specific actor, say, ‘Nick Wahlberg’?

SELECT f.film\_id, f.title  
FROM film f  
JOIN film\_actor fa ON f.film\_id = fa.film\_id  
JOIN actor a ON fa.actor\_id = a.actor\_id  
WHERE a.first\_name = 'Nick' AND a.last\_name = 'Wahlberg';

A screenshot of a computer

AI-generated content may be incorrect.

This query returns the IDs and titles of all films starring Nick Wahlberg

**3. List All Categories of Films**  
**Question**: How can I list all categories of films available?

SELECT category\_id, name  
FROM category;

A screenshot of a computer

AI-generated content may be incorrect.

This query lists all film categories by ID and name.

**4. Find All Movies in a Specific Category**  
**Question:** How can I find all movies in the ‘Comedy’ category?

SELECT f.film\_id, f.title  
FROM film f  
JOIN film\_category fc ON f.film\_id = fc.film\_id  
JOIN category c ON fc.category\_id = c.category\_id  
WHERE c.name = 'Comedy';

A screenshot of a computer

AI-generated content may be incorrect.

This query returns the IDs and titles of all films in the Comedy category.

**5. Get Customer Details with Rental History**  
**Question:** How can I get details of customers along with their rental history?

SELECT c.customer\_id, c.first\_name, c.last\_name, r.rental\_id, r.rental\_date, r.return\_date  
FROM customer c  
JOIN rental r ON c.customer\_id = r.customer\_id  
ORDER BY c.customer\_id, r.rental\_date;

A screenshot of a computer

AI-generated content may be incorrect.

This query lists customer details along with their rental IDs, rental dates, and return dates.

**6. Calculate Total Payments by Customer**  
**Question:** How can I calculate the total amount paid by each customer?

SELECT c.customer\_id, c.first\_name, c.last\_name, SUM(p.amount) AS total\_paid  
FROM customer c  
JOIN payment p ON c.customer\_id = p.customer\_id  
GROUP BY c.customer\_id, c.first\_name, c.last\_name  
ORDER BY total\_paid DESC;

A screenshot of a computer

AI-generated content may be incorrect.

This query calculates the total amount paid by each customer.

**7. Find the Most Popular Movie**  
**Question: How can I find the most rented movie?**

SELECT f.film\_id, f.title, COUNT(r.rental\_id) AS rental\_count  
FROM film f  
JOIN inventory i ON f.film\_id = i.film\_id  
JOIN rental r ON i.inventory\_id = r.inventory\_id  
GROUP BY f.film\_id, f.title  
ORDER BY rental\_count DESC  
LIMIT 1;

A screenshot of a computer

AI-generated content may be incorrect.

This query finds the most rented movie by counting rentals

**8. Get Details of All Staff Members**  
**Question:** How can I get details of all staff members?

SELECT staff\_id, first\_name, last\_name, email, store\_id  
FROM staff;

A screenshot of a computer

AI-generated content may be incorrect.

This query lists all staff members’ details.

**9. List All Films Along with Their Language**  
**Question:** How can I list all films along with their language?

SELECT f.film\_id, f.title, l.name AS language  
FROM film f  
JOIN language l ON f.language\_id = l.language\_id;

A screenshot of a computer

AI-generated content may be incorrect.

This query returns film titles along with their languages.

**10. Find Rentals by Store**  
\*Question:\* How can I find all rentals made from a specific store, say store\_id 1?

SELECT r.rental\_id, r.rental\_date, r.return\_date, r.customer\_id  
FROM rental r  
JOIN inventory i ON r.inventory\_id = i.inventory\_id  
WHERE i.store\_id = 1;

A screenshot of a computer

AI-generated content may be incorrect.

This query returns all rentals from store\_id 1.

**11. Get Address Details of All Customers**  
**Question:** How can I get the address details of all customers?

SELECT c.customer\_id, c.first\_name, c.last\_name, a.address, a.city\_id  
FROM customer c  
JOIN address a ON c.address\_id = a.address\_id;

A screenshot of a computer

AI-generated content may be incorrect.

This query lists customer addresses.

**12. Count Movies by Category**  
**Question:** How can I count the number of movies in each category?

SELECT c.name AS category, COUNT(fc.film\_id) AS movie\_count  
FROM category c  
JOIN film\_category fc ON c.category\_id = fc.category\_id  
GROUP BY c.name  
ORDER BY movie\_count DESC;

A screenshot of a computer

AI-generated content may be incorrect.

This query counts movies in each category

**13. Get the Total Number of Rentals by Each Staff Member**  
**Question:** How can I get the total number of rentals handled by each staff member?

SELECT s.staff\_id, s.first\_name, s.last\_name, COUNT(r.rental\_id) AS rental\_count  
FROM staff s  
JOIN rental r ON s.staff\_id = r.staff\_id  
GROUP BY s.staff\_id, s.first\_name, s.last\_name  
ORDER BY rental\_count DESC;

A screenshot of a computer

AI-generated content may be incorrect.

This query counts rentals handled by each staff member.

**Findings and Insights**  
**- Most Popular Actor:** The actor with the most appearances in films is revealed, providing insight into casting trends.  
**- Comedy Films:** The Comedy category is one of the most popular, indicating customer preferences towards this genre.  
**- Top Customers:** Analysis shows which customers have rented the most DVDs, highlighting the most active users.  
**- Overdue Rentals:** A significant number of rentals are overdue, suggesting a need for better return reminders or policies.  
**- Total Payments:** The total payments data helps identify the most valuable customers by rental revenue.

**Recommendations**  
**- Enhance Customer Engagement:** Implement targeted marketing campaigns focusing on the top genres and actors to increase rentals.  
**- Loyalty Programs:** Introduce or improve loyalty programs to reward frequent renters and encourage more rentals.  
**- Improve Return Policies:** Develop better reminder systems or penalties for overdue rentals to reduce the number of late returns.  
**- Customer Surveys:** Collect feedback from top customers to understand their preferences and improve the overall service experience.

**Conclusion**  
This analysis of the DVD Rentals database provided valuable insights into customer behavior and business performance. By leveraging SQL to uncover patterns and trends, actionable recommendations can be made to enhance customer engagement, optimize rental policies, and ultimately boost business growth. This case study demonstrates the power of SQL in transforming data into strategic business decisions.