-- Find out how many rentals are still outstanding (ie. have not been returned to the store yet).

-- SELECT COUNT(\*) FROM rental; --16044

-- counting all the rows in the table

SELECT COUNT(return\_date) FROM rental; --15861

-- COUNT ONLTY THE ROWS

-- where this column is not null

SELECT COUNT(\*) FROM rental

WHERE return\_date IS NULL --183

-- Find the 30 most expensive movies which are outstanding (ie. have not been returned to the store yet)

SELECT film.title, film.replacement\_cost

FROM film

INNER JOIN inventory ON film.film\_id = inventory.film\_id

INNER JOIN rental ON rental.inventory\_id = inventory.inventory\_id

ORDER BY replacement\_cost DESC

LIMIT 30

-- The 1st film : The film is about a sumo wrestler, and one of the actors is Penelope Monroe.

SELECT title, description, fulltext

FROM film

INNER JOIN film\_actor AS fa ON film.film\_id = fa.film\_id

INNER JOIN actor ON actor.actor\_id = fa.actor\_id

WHERE description ILIKE '%sumo wrestler%'

AND first\_name ILIKE '%penelope%'

AND last\_name ILIKE '%monroe%'

-- SELECT title

-- FROM film

-- WHERE fulltext @@ tsquery('sumo') AND fulltext @@ tsquery('wrestler')

-- The 2nd film : A short documentary (less than 1 hour long), rated "R".

SELECT title, category.name, film.length

FROM film

INNER JOIN film\_category AS fc ON film.film\_id = fc.film\_id

INNER JOIN category ON category.category\_id = fc.category\_id

WHERE category.name ILIKE '%documentary%'

AND film.length < 60

AND rating = 'R'

-- The 3rd film : A film that his friend Matthew Mahan rented.

-- He paid over $4.00 for the rental,

-- and he returned it between the 28th of July and the 1st of August, 2005.

SELECT film.title, payment.amount, customer.first\_name, customer.last\_name

FROM film

INNER JOIN inventory ON film.film\_id = inventory.film\_id

INNER JOIN rental ON inventory.inventory\_id = rental.inventory\_id

INNER JOIN payment ON rental.rental\_id = payment.rental\_id

INNER JOIN customer ON customer.customer\_id = rental.customer\_id

WHERE payment.amount > 4 AND

customer.first\_name = 'Matthew' AND

customer.last\_name = 'Mahan'

AND rental.return\_date BETWEEN '2005-07-28' AND '2005-08-01'

-- The 4th film : His friend Matthew Mahan watched this film,

-- as well. It had the word "boat" in the title or description,

-- and it looked like it was a very expensive DVD to replace.

SELECT film.title, film.replacement\_cost

FROM film

INNER JOIN inventory ON film.film\_id = inventory.film\_id

INNER JOIN rental ON inventory.inventory\_id = rental.inventory\_id

INNER JOIN customer ON customer.customer\_id = rental.customer\_id

WHERE

customer.first\_name = 'Matthew'

AND

customer.last\_name = 'Mahan'

AND (title ILIKE '%boat%' OR description ILIKE '%boat%')

ORDER BY film.replacement\_cost DESC