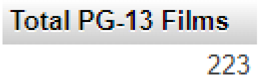
SQL ACTIVITIES

Count the Number of Films of a Specific Rating

Create a query that counts the total number of films with a PG-13 rating in the film table. Rename the column to Total PG-13 Films.

Your result should look like this:



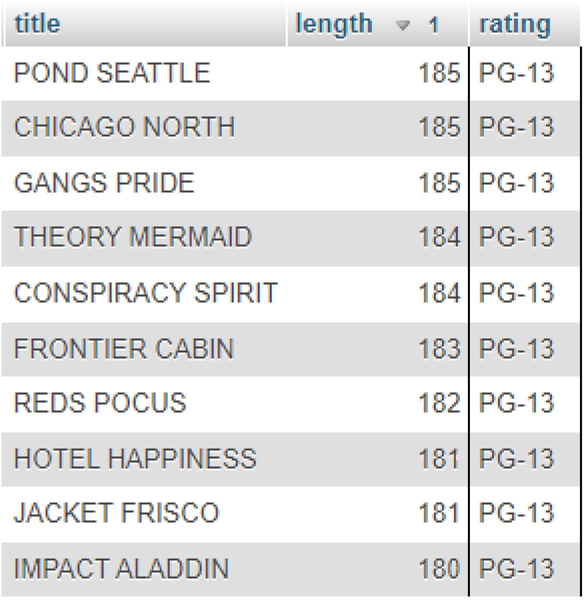
Total PG-13 Films
223

```
SELECT COUNT(*) AS "Total PG-13 Films"

FROM film

WHERE rating = 'PG-13';
```

Get the top 10 max length films

In the last activity, you counted the number of films with a PG-13 rating. Now, create a query that returns the top 10 longest length films in the film table which have a PG-13 rating. Include the title, length, and rating columns in your result.

Your result should look like this:



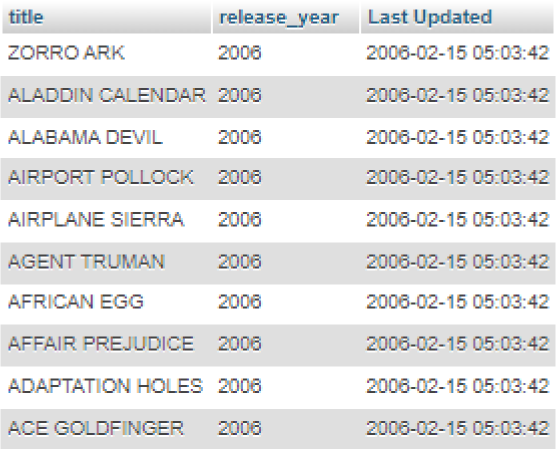
title	length ▼ 1	rating
POND SEATTLE	185	PG-13
CHICAGO NORTH	185	PG-13
GANGS PRIDE	185	PG-13
THEORY MERMAID	184	PG-13
CONSPIRACY SPIRIT	184	PG-13
FRONTIER CABIN	183	PG-13
REDS POCUS	182	PG-13
HOTEL HAPPINESS	181	PG-13
JACKET FRISCO	181	PG-13
IMPACT ALADDIN	180	PG-13

```
SELECT title, length, rating

FROM film

WHERE rating = 'PG-13'

ORDER BY length DESC

LIMIT 10;
```

Get the 10 latest updated films

Create a query that returns the 10 most recently updated films in the film table. Include the title, release_year, and last_update columns in your result. Rename the last_update column to Last Updated.

Your result should look like this:



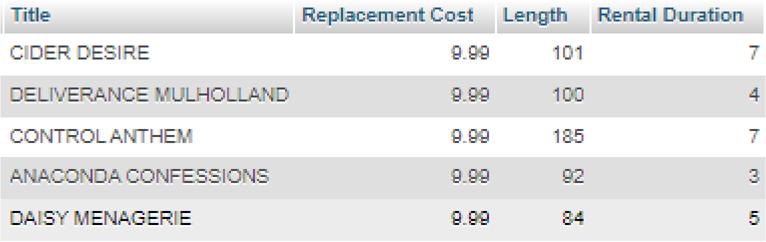
title	release_year	Last Updated
ZORRO ARK	2006	2006-02-15 05:03:42
ALADDIN CALENDAR	2006	2006-02-15 05:03:42
ALABAMA DEVIL	2006	2006-02-15 05:03:42
AIRPORT POLLOCK	2006	2006-02-15 05:03:42
AIRPLANE SIERRA	2006	2006-02-15 05:03:42
AGENT TRUMAN	2006	2006-02-15 05:03:42
AFRICAN EGG	2006	2006-02-15 05:03:42
AFFAIR PREJUDICE	2006	2006-02-15 05:03:42
ADAPTATION HOLES	2006	2006-02-15 05:03:42
ACE GOLDFINGER	2006	2006-02-15 05:03:42

```
SELECT title, release_year, last_update AS "Last Updated"

FROM film

ORDER BY last_update DESC

LIMIT 10;
```

English films with lowest replacement cost

Create a query that returns the 5 English films with the lowest replacement cost in the film table. Include the title, replacement_cost, length, and rental_duration columns in your result. Rename the title, replacement_cost, length, and rental_duration columns to Title, Replacement Cost, Length, and Rental Duration, respectively.

Your result should look like this:



Title	Replacement Cost	Length	Rental Duration
CIDER DESIRE	9.99	101	7
DELIVERANCE MULHOLLAND	9.99	100	4
CONTROL ANTHEM	9.99	185	7
ANACONDA CONFESSIONS	9.99	92	3
DAISY MENAGERIE	9.99	84	5

SELECT language_id

FROM language

WHERE name = 'English';

SELECT title **AS** "Title", replacement_cost **AS** "Replacement Cost", length **AS** "Length", rental_duration **AS** "Rental Duration"

FROM film

WHERE language_id = 1

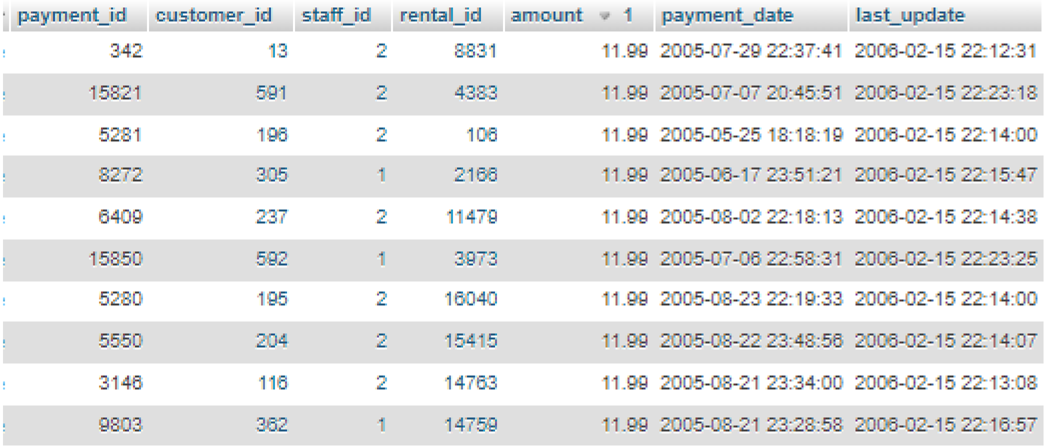
ORDER BY replacement_cost **ASC**

LIMIT 5;

Retrieve the 10 Highest Payment Amounts

Construct a query that retrieves all the details for the top 10 highest payments from the payment table.

Your result should look like this:



payment_id	customer_id	staff_id	rental_id	amount ▼ 1	payment_date	last_update
342	13	2	8831	11.99	2005-07-29 22:37:41	2006-02-15 22:12:31
15821	591	2	4383	11.99	2005-07-07 20:45:51	2006-02-15 22:23:18
5281	196	2	106	11.99	2005-05-25 18:18:19	2006-02-15 22:14:00
8272	305	1	2166	11.99	2005-06-17 23:51:21	2006-02-15 22:15:47
6409	237	2	11479	11.99	2005-08-02 22:18:13	2006-02-15 22:14:38
15850	592	1	3973	11.99	2005-07-06 22:58:31	2006-02-15 22:23:25
5280	195	2	16040	11.99	2005-08-23 22:19:33	2006-02-15 22:14:00
5550	204	2	15415	11.99	2005-08-22 23:48:56	2006-02-15 22:14:07
3146	116	2	14763	11.99	2005-08-21 23:34:00	2006-02-15 22:13:08
9803	362	1	14759	11.99	2005-08-21 23:28:58	2006-02-15 22:16:57

SELECT *

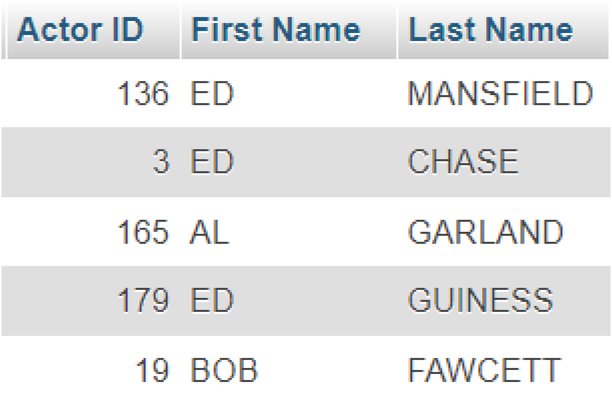
```
FROM payment

ORDER BY amount DESC

LIMIT 10;
```

Retrieve Actors with Shortest First Names

Craft a query that retrieves the top 5 actors with the shortest first names from the actor table. Include the actor_id, first_name, and last_name columns in your result set. Sort the result by the length of the first_name in ascending order and rename the actor_id column to Actor ID, first_name to First Name, and last_name to Last Name.

Expected Output

Your result should display the actors with the shortest first names at the top, limited to 5 rows.



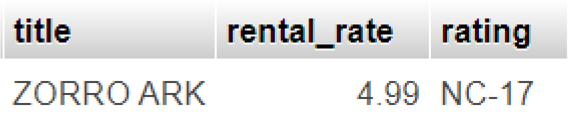
Actor ID	First Name	Last Name
136	ED	MANSFIELD
3	ED	CHASE
165	AL	GARLAND
179	ED	GUINESS
19	BOB	FAWCETT

```
SELECT actor_id AS "Actor ID", first_name AS "First Name", last_name AS "Last Name"

FROM actor

ORDER BY LENGTH(first_name) ASC

LIMIT 5;
```

Find the Film with the Maximum Rental Rate

Write a query to fetch the details of the film with the maximum rental rate from the film table. The query should display the columns title, rental_rate, and rating. If multiple films have the maximum rental rate, return the one that is alphabetically last by title.

Your result should look like this:



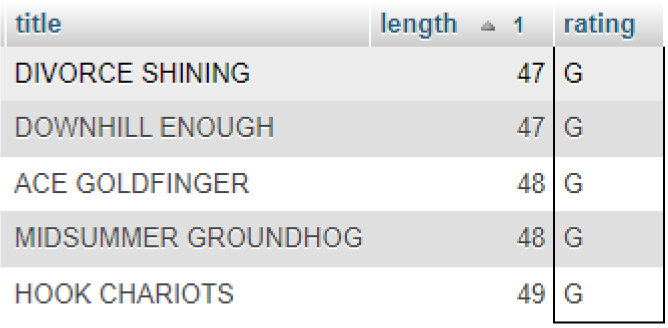
title	rental_rate	rating
ZORRO ARK	4.99	NC-17

```
SELECT MAX(rental_rate)

FROM film;

SELECT title, rental_rate, rating

FROM film

WHERE rental_rate=4.99

ORDER BY title DESC

LIMIT 1;
```

Retrieve the Shortest Films for Children

Did you know that the Children's films are those with a rating of 'G'? Off course you did! So, let's find the 5 shortest films in the film table that are rated 'G'. Display columns title, length, and rating. Sort the result by the length of the film in ascending order.

Your result should look like this:



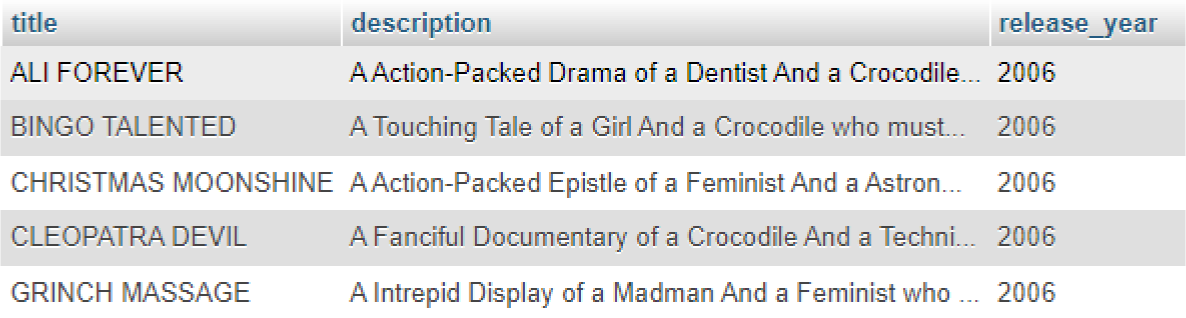
title	length ▲ 1	rating
DIVORCE SHINING	47	G
DOWNHILL ENOUGH	47	G
ACE GOLDFINGER	48	G
MIDSUMMER GROUNDHOG	48	G
HOOK CHARIOTS	49	G

```
SELECT title, length, rating

FROM film

WHERE rating = 'G'

ORDER BY length ASC

LIMIT 5;
```

List Films with a Specific Length

Create a query that returns the details of films from the film table that have a length of exactly 150 minutes. Include title, description, and release_year columns in your result. Display only the first 5 films that satisfy this condition.

Your result should look like this:



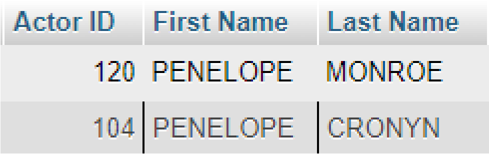
title	description	release_year
ALI FOREVER	A Action-Packed Drama of a Dentist And a Crocodile...	2006
BINGO TALENTED	A Touching Tale of a Girl And a Crocodile who must...	2006
CHRISTMAS MOONSHINE	A Action-Packed Epistle of a Feminist And a Astron...	2006
CLEOPATRA DEVIL	A Fanciful Documentary of a Crocodile And a Techni...	2006
GRINCH MASSAGE	A Intrepid Display of a Madman And a Feminist who ...	2006

```
SELECT title, description, release_year

FROM film

WHERE length = 150

LIMIT 5;
```

List Last 2 Actors with a Specific First Name

Create a query that retrieves last 2 actors from the actor table with the first name PENELOPE. Include actor_id, first_name, and last_name columns in your result. Rename the actor_id, first_name and last_name columns to Actor ID, First Name and Last Name, respectively.

Your result should look like this:



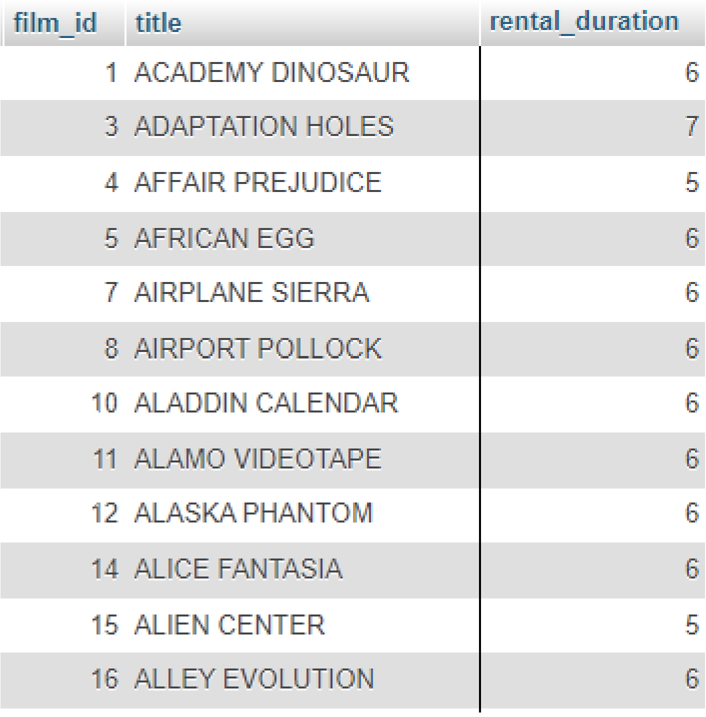
Actor ID	First Name	Last Name
120	PENELOPE	MONROE
104	PENELOPE	CRONYN

```
SELECT actor_id AS "Actor ID", first_name AS "First Name", last_name AS "Last Name"

FROM actor

WHERE first_name = 'PENELOPE'

ORDER BY actor_id DESC

LIMIT 2;
```

Identify Films with Rental Duration Longer Than Average

Construct a query that displays the film_id, title, and rental_duration of films in the film table where the rental_duration is greater than the average rental duration across all films. Do not display the average itself; just the films exceeding this value.

Your result should look like this:



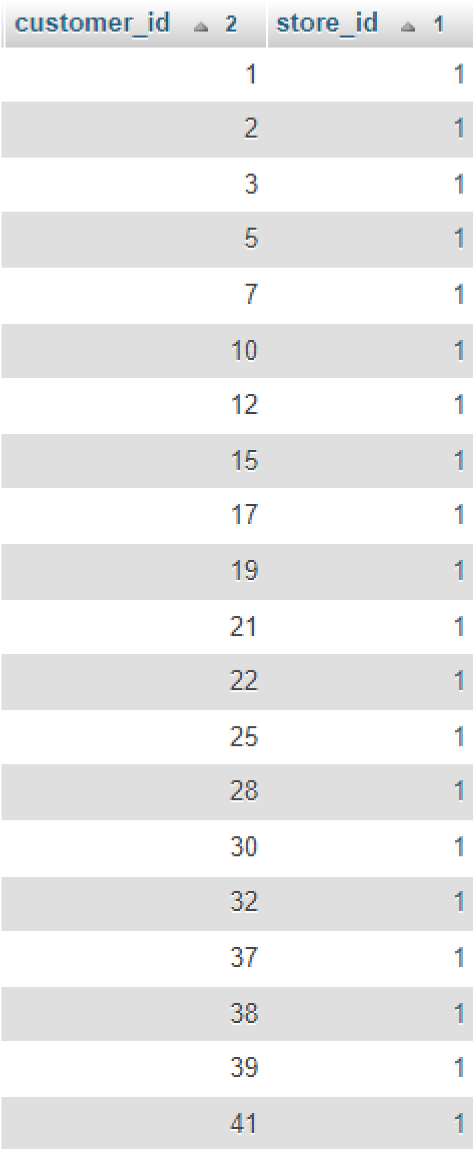
film_id	title	rental_duration
1	ACADEMY DINOSAUR	6
3	ADAPTATION HOLES	7
4	AFFAIR PREJUDICE	5
5	AFRICAN EGG	6
7	AIRPLANE SIERRA	6
8	AIRPORT POLLOCK	6
10	ALADDIN CALENDAR	6
11	ALAMO VIDEOTAPE	6
12	ALASKA PHANTOM	6
14	ALICE FANTASIA	6
15	ALIEN CENTER	5
16	ALLEY EVOLUTION	6

```
SELECT AVG(rental_duration) AS "Average Rental Duration"

FROM film;

SELECT film_id, title, rental_duration

FROM film

WHERE rental_duration > 4.985;
```

Display Customers by Store in Ascending Order

Create a query that displays the customer_id and store_id from the customer table. Order the results first by store_id in ascending order and then by customer_id in ascending order. Add a limit to show only the first 20 customers.

Your result should look like this:



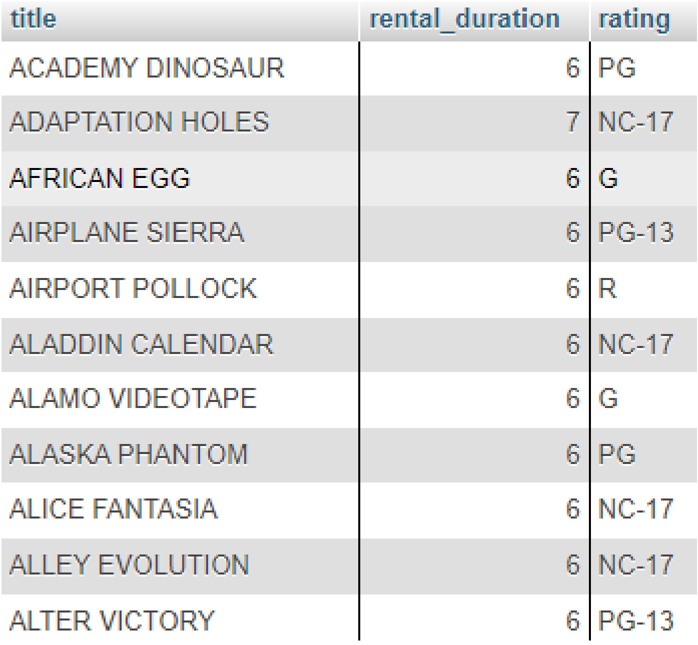
customer_id ▲ 2	store_id ▲ 1
1	1
2	1
3	1
5	1
7	1
10	1
12	1
15	1
17	1
19	1
21	1
22	1
25	1
28	1
30	1
32	1
37	1
38	1
39	1
41	1

```
SELECT customer_id, store_id

FROM customer

ORDER BY store_id ASC, customer_id ASC

LIMIT 20;
```

List Films with a Rental Duration Over 5 Days

For this activity, write a query that returns the title, rental_duration, and rating of films in the film table where the rental_duration is greater than 5 days.

Your result should look like this:



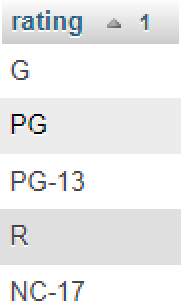
title	rental_duration	rating
ACADEMY DINOSAUR	6	PG
ADAPTATION HOLES	7	NC-17
AFRICAN EGG	6	G
AIRPLANE SIERRA	6	PG-13
AIRPORT POLLOCK	6	R
ALADDIN CALENDAR	6	NC-17
ALAMO VIDEOTAPE	6	G
ALASKA PHANTOM	6	PG
ALICE FANTASIA	6	NC-17
ALLEY EVOLUTION	6	NC-17
ALTER VICTORY	6	PG-13

```
SELECT title, rental_duration, rating

FROM film

WHERE rental_duration > 5;
```

Retrieve Unique Film Ratings

Create a query that returns the unique film ratings from the film table. Order the results by the rating in ascending order.

Your result should look like this:



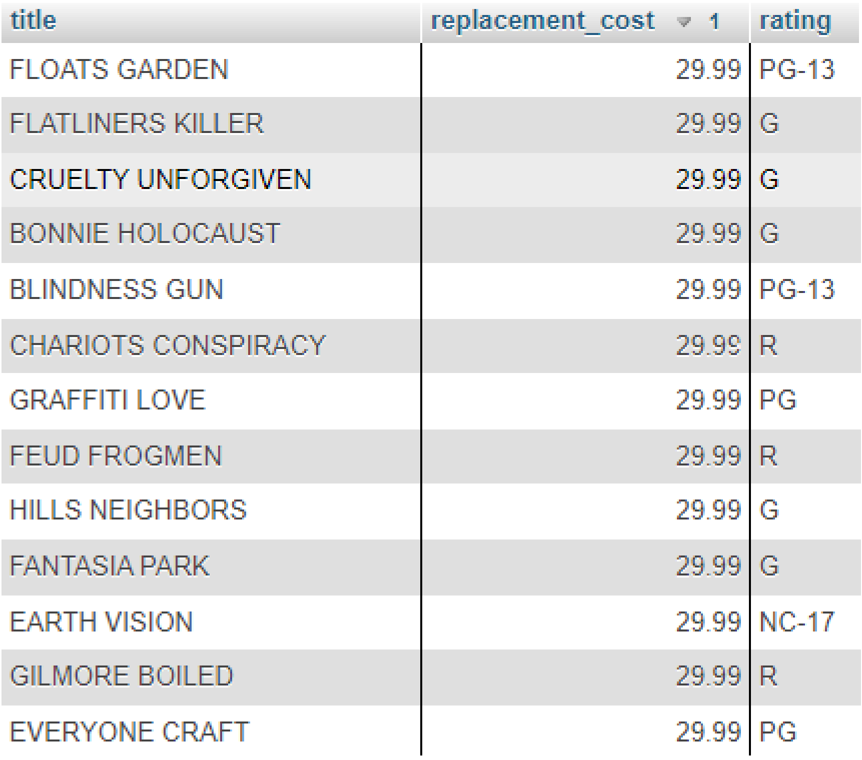
rating ▲ 1
G
PG
PG-13
R
NC-17

```
SELECT DISTINCT rating

FROM film

ORDER BY rating ASC;
```

Retrieve Films with Replacement Cost Greater Than $15

Create a query that returns the title, replacement_cost, and rating of films in the film table where the replacement_cost is greater than $15.00. Order the results by the replacement_cost in descending order.

Your result should look like this:



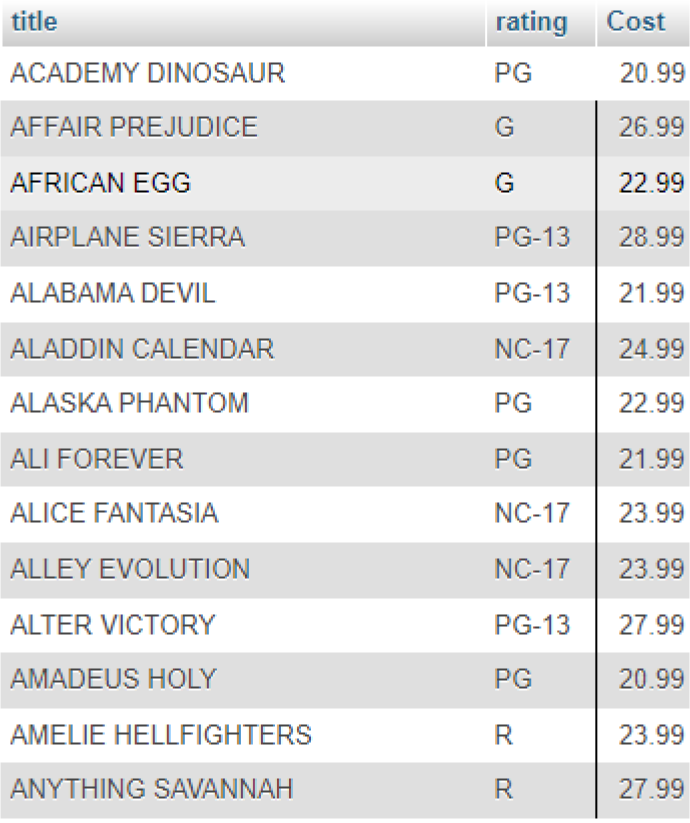
title	replacement_cost ▼ 1	rating
FLOATS GARDEN	29.99	PG-13
FLATLINERS KILLER	29.99	G
CRUELTY UNFORGIVEN	29.99	G
BONNIE HOLOCAUST	29.99	G
BLINDNESS GUN	29.99	PG-13
CHARIOTS CONSPIRACY	29.99	R
GRAFFITI LOVE	29.99	PG
FEUD FROGMEN	29.99	R
HILLS NEIGHBORS	29.99	G
FANTASIA PARK	29.99	G
EARTH VISION	29.99	NC-17
GILMORE BOILED	29.99	R
EVERYONE CRAFT	29.99	PG

```
SELECT title, replacement_cost, rating

FROM film

WHERE replacement_cost > 15.00

ORDER BY replacement_cost DESC;
```

Retrieve Films with a Replacement Cost Above a Certain Amount

Create a query that retrieves all films from the film table with a replacement cost greater than $20.00. Only include the title, rating, and replacement_cost columns in your result set. Use an alias to rename replacement_cost to Cost.

Your result should look similar to this:



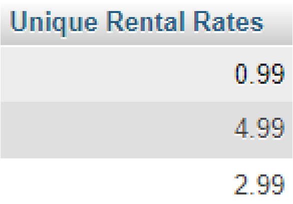
title	rating	Cost
ACADEMY DINOSAUR	PG	20.99
AFFAIR PREJUDICE	G	26.99
AFRICAN EGG	G	22.99
AIRPLANE SIERRA	PG-13	28.99
ALABAMA DEVIL	PG-13	21.99
ALADDIN CALENDAR	NC-17	24.99
ALASKA PHANTOM	PG	22.99
ALI FOREVER	PG	21.99
ALICE FANTASIA	NC-17	23.99
ALLEY EVOLUTION	NC-17	23.99
ALTER VICTORY	PG-13	27.99
AMADEUS HOLY	PG	20.99
AMELIE HELLFIGHTERS	R	23.99
ANYTHING SAVANNAH	R	27.99

```
SELECT title, rating, replacement_cost AS "Cost"

FROM film

WHERE replacement_cost > 20.00;
```

List the Distinct Rental Rates of Films

Create a query that lists all distinct rental rates available in the film table. Use the alias Unique Rental Rates for the resulting column.

Your result should look similar to this:



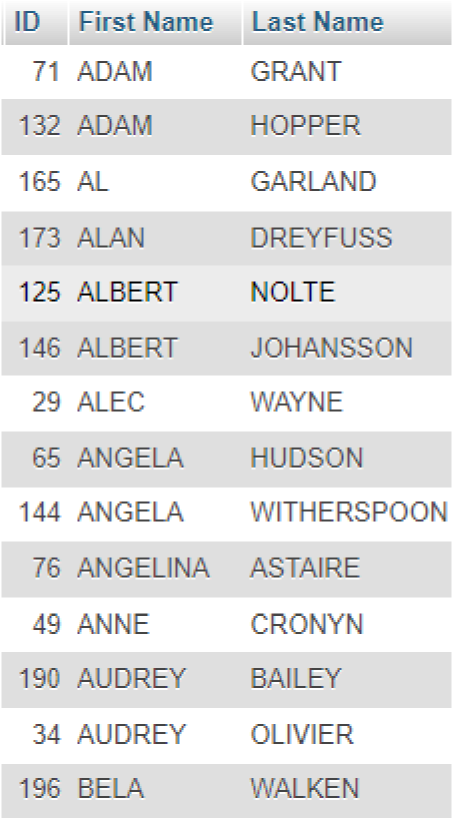
Unique Rental Rates
0.99
4.99
2.99

```
SELECT DISTINCT rental_rate AS "Unique Rental Rates"

FROM film;
```

List Actors Ordered by First Name

Create a query that lists all actors from the actor table, ordered by their first name in ascending order. Only include actor_id, first_name, and last_name columns in your result set. Optionally, use the aliases ID, First Name, and Last Name for the respective columns.

Your result should look something like this:



ID	First Name	Last Name
71	ADAM	GRANT
132	ADAM	HOPPER
165	AL	GARLAND
173	ALAN	DREYFUSS
125	ALBERT	NOLTE
146	ALBERT	JOHANSSON
29	ALEC	WAYNE
65	ANGELA	HUDSON
144	ANGELA	WITHERSPOON
76	ANGELINA	ASTAIRE
49	ANNE	CRONYN
190	AUDREY	BAILEY
34	AUDREY	OLIVIER
196	BELA	WALKEN

SELECT actor_id AS "ID", first_name AS "First Name", last_name AS "Last Name"

FROM actor

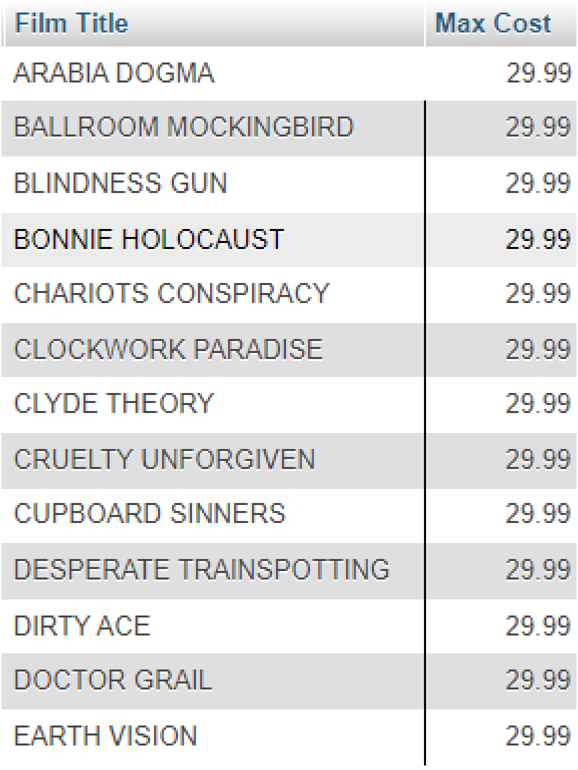
ORDER BY first_name ASC;

Single Film with the Maximum Replacement Cost

Firstly, find out the maximum replacement cost from the film table.

Then, create a separate query to retrieve the title and replacement_cost of the film that has this maximum replacement cost. Use aliases to rename title to Film Title and replacement_cost to Max Cost.

Your result should look like this:



Film Title	Max Cost
ARABIA DOGMA	29.99
BALLROOM MOCKINGBIRD	29.99
BLINDNESS GUN	29.99
BONNIE HOLOCAUST	29.99
CHARIOTS CONSPIRACY	29.99
CLOCKWORK PARADISE	29.99
CLYDE THEORY	29.99
CRUELTY UNFORGIVEN	29.99
CUPBOARD SINNERS	29.99
DESPERATE TRAINSPOTTING	29.99
DIRTY ACE	29.99
DOCTOR GRAIL	29.99
EARTH VISION	29.99

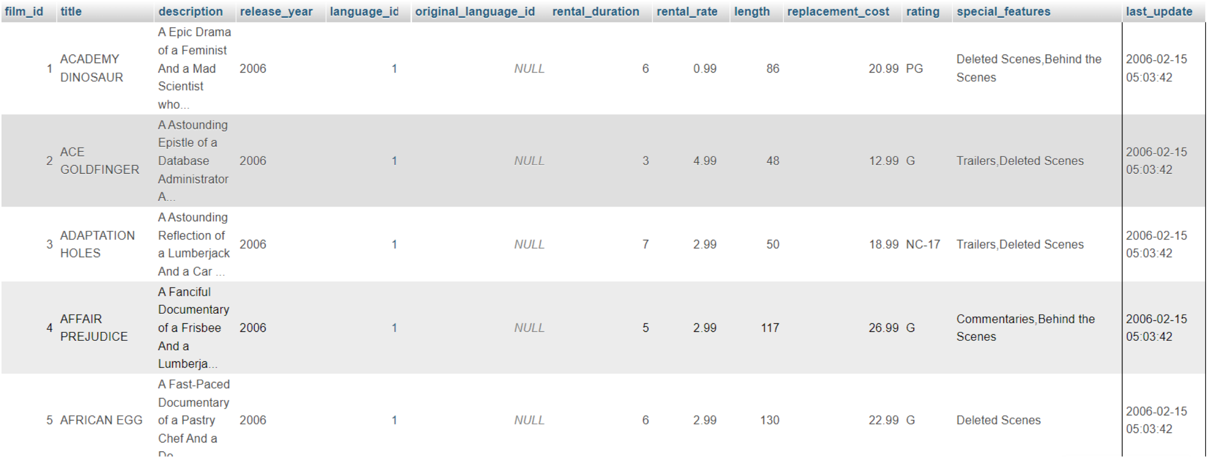
```
SELECT MAX(replacement_cost) AS "Max Cost"

FROM film;

SELECT title AS "Film Title", replacement_cost AS "Max Cost"

FROM film

WHERE replacement_cost = 29.99;
```

Retrieve the Details of the Most Recent Film Added to the Inventory

For the first part, find the latest last_update timestamp in the film table.

Then, write a query to retrieve all details for the film(s) with this latest last_update timestamp.

Your result should look like this:



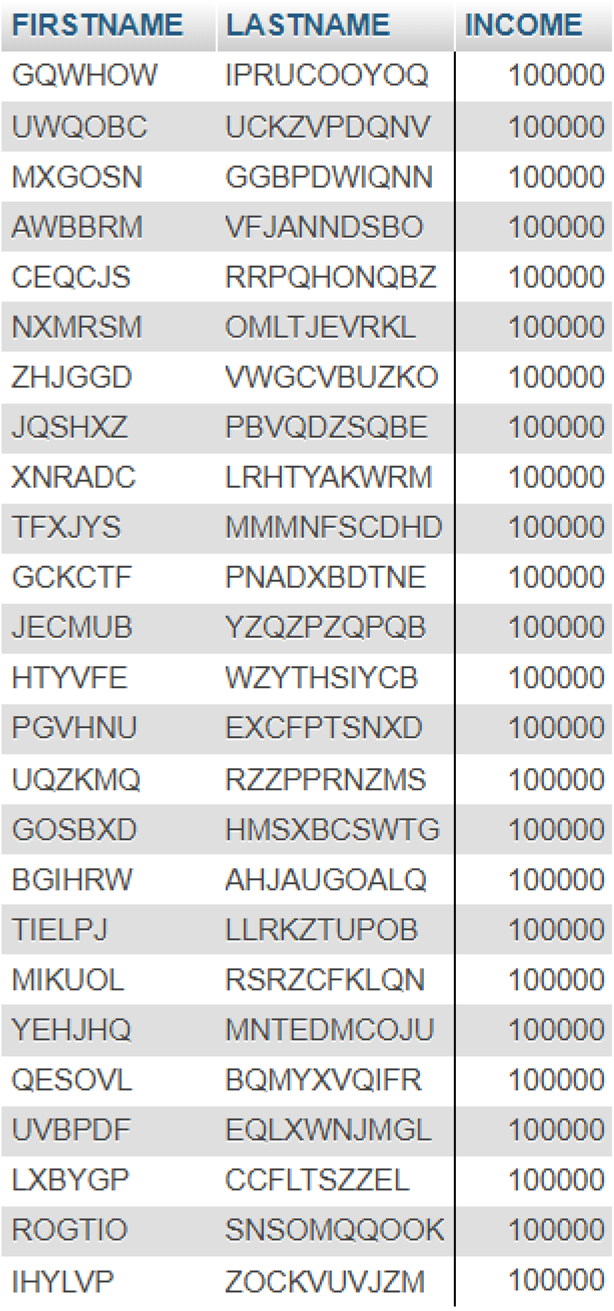
film_id	title	description	release_year	language_id	original_language_id	rental_duration	rental_rate	length	replacement_cost	rating	special_features	last_update
1	ACADEMY DINOSAUR	A Epic Drama of a Feminist And a Mad Scientist who...	2006	1	*NULL*	6	0.99	86	20.99	PG	Deleted Scenes,Behind the Scenes	2006-02-15 05:03:42
2	ACE GOLDFINGER	A Astounding Epistle of a Database Administrator A...	2006	1	*NULL*	3	4.99	48	12.99	G	Trailers,Deleted Scenes	2006-02-15 05:03:42
3	ADAPTATION HOLES	A Astounding Reflection of a Lumberjack And a Car ...	2006	1	*NULL*	7	2.99	50	18.99	NC-17	Trailers,Deleted Scenes	2006-02-15 05:03:42
4	AFFAIR PREJUDICE	A Fanciful Documentary of a Frisbee And a Lumberja...	2006	1	*NULL*	5	2.99	117	26.99	G	Commentaries,Behind the Scenes	2006-02-15 05:03:42
5	AFRICAN EGG	A Fast-Paced Documentary of a Pastry Chef And a Do...	2006	1	*NULL*	6	2.99	130	22.99	G	Deleted Scenes	2006-02-15 05:03:42

```
SELECT MAX(last_update) AS "Latest Last Update"

FROM film;

SELECT *

FROM film

WHERE last_update = '2006-02-15 05:03:42';
```

Identify High-Value Customers

Write a query to find all customers whose income is above 80000 AND who have made at least one purchase with a total amount exceeding 100. Display their first name, last name, and income.

Expected outcome:



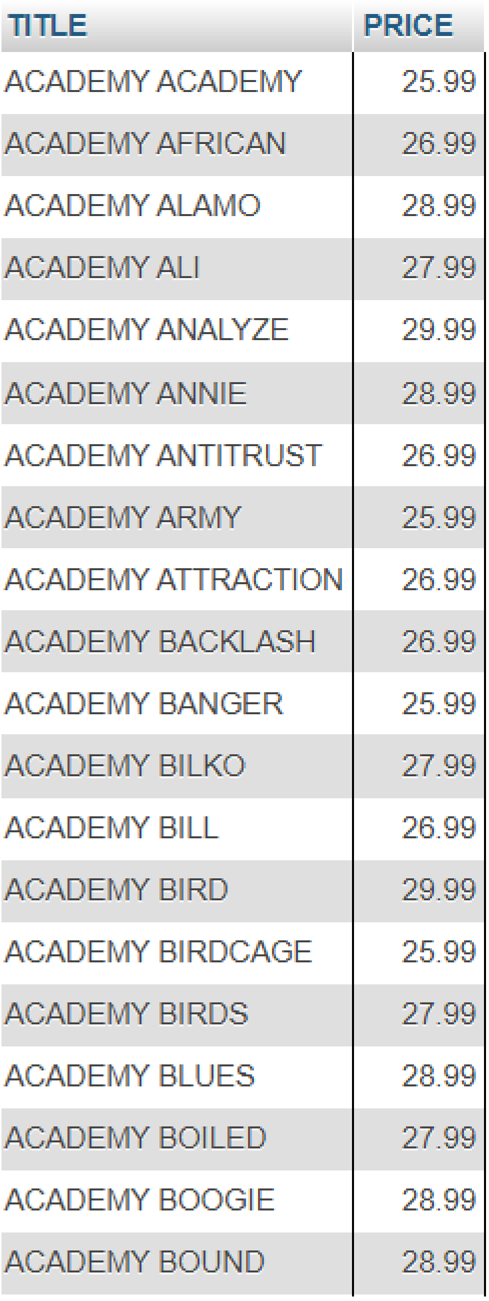
FIRSTNAME	LASTNAME	INCOME
GQWHOW	IPRUCOOYOQ	100000
UWQOBC	UCKZVPDQNV	100000
MXGOSN	GGBPDWIQNN	100000
AWBBRM	VFJANNDSBO	100000
CEQCJS	RRPQHONQBZ	100000
NXMRSM	OMLTJEVRKL	100000
ZHJGGD	VWGCVBUZKO	100000
JQSHXZ	PBVQDZSQBE	100000
XNRADC	LRHTYAKWRM	100000
TFXJYS	MMMNFSCDHD	100000
GCKCTF	PNADXBDTNE	100000
JECMUB	YZQZPZQPQB	100000
HTYVFE	WZYTHSIYCB	100000
PGVHNU	EXCFPTSNXD	100000
UQZKMQ	RZZPPRNZMS	100000
GOSBXD	HMSXBCSWTG	100000
BGIHRW	AHJAUGOALQ	100000
TIELPJ	LLRKZTUPOB	100000
MIKUOL	RSRZCFKLQN	100000
YEHJHQ	MNTEDMCOJU	100000
QESOVL	BQMYXVQIFR	100000
UVBPDF	EQLXWNJMGL	100000
LXBYGP	CCFLTSZZEL	100000
ROGTIO	SNSOMQQOOK	100000
IHYLVP	ZOCKVUVJZM	100000

```
SELECT FIRSTNAME, LASTNAME, INCOME

FROM CUSTOMERS c

JOIN ORDERS o ON c.CUSTOMERID = o.CUSTOMERID

WHERE INCOME > 80000 AND o.TOTALAMOUNT > 100;
```

Premium Movie Selection

Find all movies that are either priced above 25 OR are marked as both special (SPECIAL=1) AND membership items (MEMBERSHIP_ITEM = 1). Display the title and price.

Expected outcome:



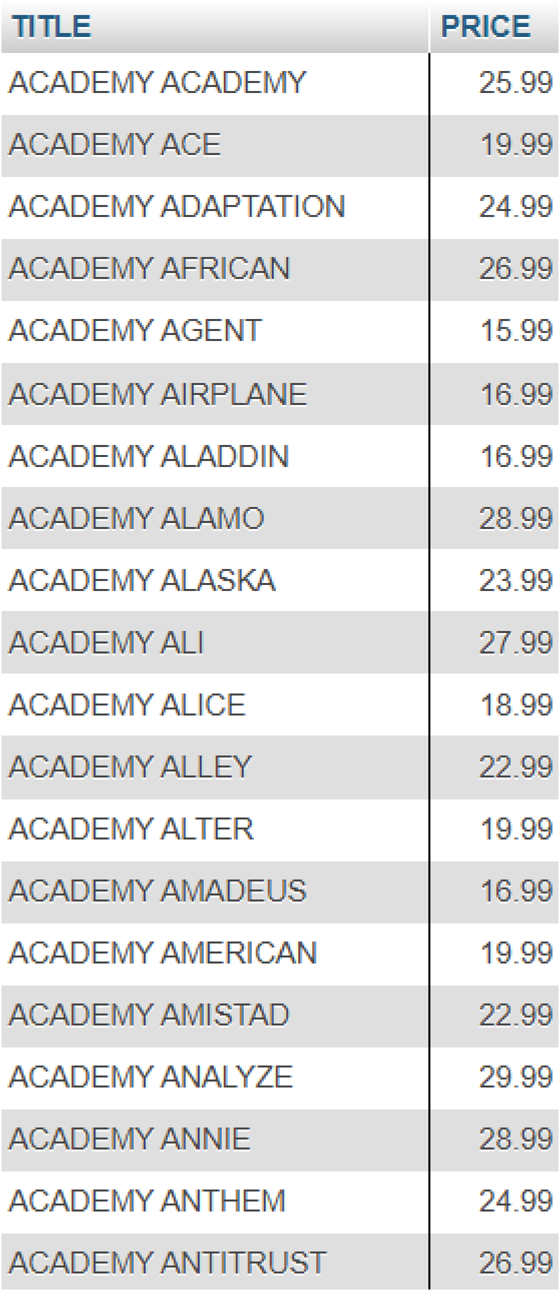
TITLE	PRICE
ACADEMY ACADEMY	25.99
ACADEMY AFRICAN	26.99
ACADEMY ALAMO	28.99
ACADEMY ALI	27.99
ACADEMY ANALYZE	29.99
ACADEMY ANNIE	28.99
ACADEMY ANTITRUST	26.99
ACADEMY ARMY	25.99
ACADEMY ATTRACTION	26.99
ACADEMY BACKLASH	26.99
ACADEMY BANGER	25.99
ACADEMY BILKO	27.99
ACADEMY BILL	26.99
ACADEMY BIRD	29.99
ACADEMY BIRDCAGE	25.99
ACADEMY BIRDS	27.99
ACADEMY BLUES	28.99
ACADEMY BOILED	27.99
ACADEMY BOOGIE	28.99
ACADEMY BOUND	28.99

```
SELECT TITLE, PRICE

FROM PRODUCTS

WHERE PRICE > 25 OR (SPECIAL = 1 AND MEMBERSHIP_ITEM = 1);
```

Price Range Analysis

Find all products with prices between 15 and 30 using the BETWEEN operator. Display the title and price.

Expected outcome:



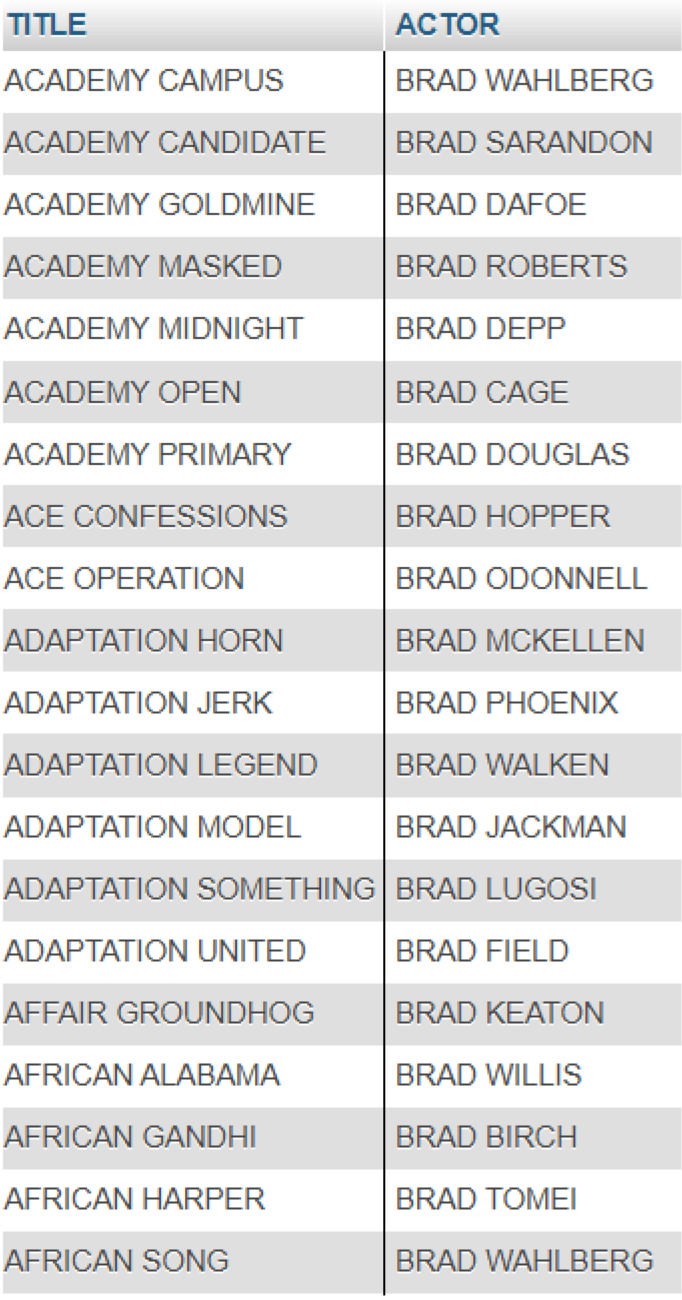
TITLE	PRICE
ACADEMY ACADEMY	25.99
ACADEMY ACE	19.99
ACADEMY ADAPTATION	24.99
ACADEMY AFRICAN	26.99
ACADEMY AGENT	15.99
ACADEMY AIRPLANE	16.99
ACADEMY ALADDIN	16.99
ACADEMY ALAMO	28.99
ACADEMY ALASKA	23.99
ACADEMY ALI	27.99
ACADEMY ALICE	18.99
ACADEMY ALLEY	22.99
ACADEMY ALTER	19.99
ACADEMY AMADEUS	16.99
ACADEMY AMERICAN	19.99
ACADEMY AMISTAD	22.99
ACADEMY ANALYZE	29.99
ACADEMY ANNIE	28.99
ACADEMY ANTHEM	24.99
ACADEMY ANTITRUST	26.99

```
SELECT TITLE, PRICE

FROM PRODUCTS

WHERE PRICE BETWEEN 15 AND 30;
```

Find Movies with Actors Named Brad

Find all movies where the actor's name starts with Brad using the LIKE operator. Show the title and actor name.

Expected outcome:



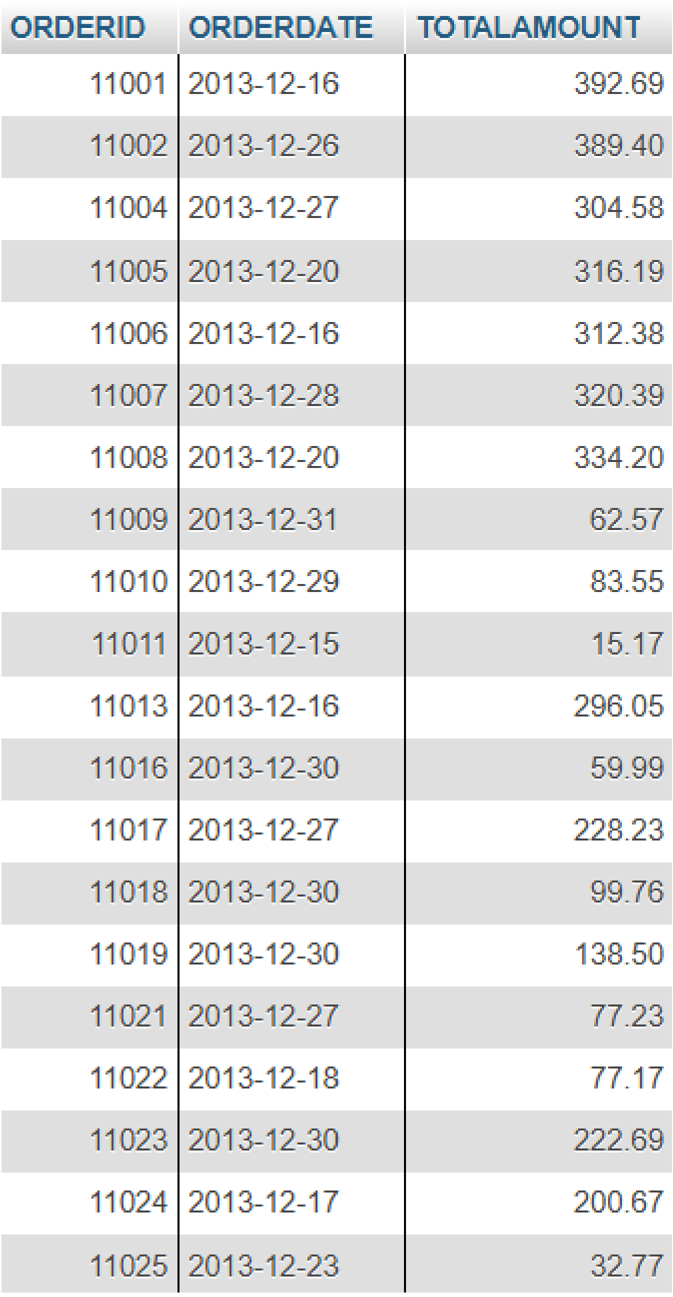
TITLE	ACTOR
ACADEMY CAMPUS	BRAD WAHLBERG
ACADEMY CANDIDATE	BRAD SARANDON
ACADEMY GOLDMINE	BRAD DAFOE
ACADEMY MASKED	BRAD ROBERTS
ACADEMY MIDNIGHT	BRAD DEPP
ACADEMY OPEN	BRAD CAGE
ACADEMY PRIMARY	BRAD DOUGLAS
ACE CONFESSIONS	BRAD HOPPER
ACE OPERATION	BRAD ODONNELL
ADAPTATION HORN	BRAD MCKELLEN
ADAPTATION JERK	BRAD PHOENIX
ADAPTATION LEGEND	BRAD WALKEN
ADAPTATION MODEL	BRAD JACKMAN
ADAPTATION SOMETHING	BRAD LUGOSI
ADAPTATION UNITED	BRAD FIELD
AFFAIR GROUNDHOG	BRAD KEATON
AFRICAN ALABAMA	BRAD WILLIS
AFRICAN GANDHI	BRAD BIRCH
AFRICAN HARPER	BRAD TOMEI
AFRICAN SONG	BRAD WAHLBERG

```
SELECT TITLE, ACTOR

FROM PRODUCTS

WHERE ACTOR LIKE 'Brad%';
```

Recent Order Analysis

Find all orders placed between 2013-12-15 and 2013-12-31. Display the order ID, order date, and total amount.

Expected outcome:



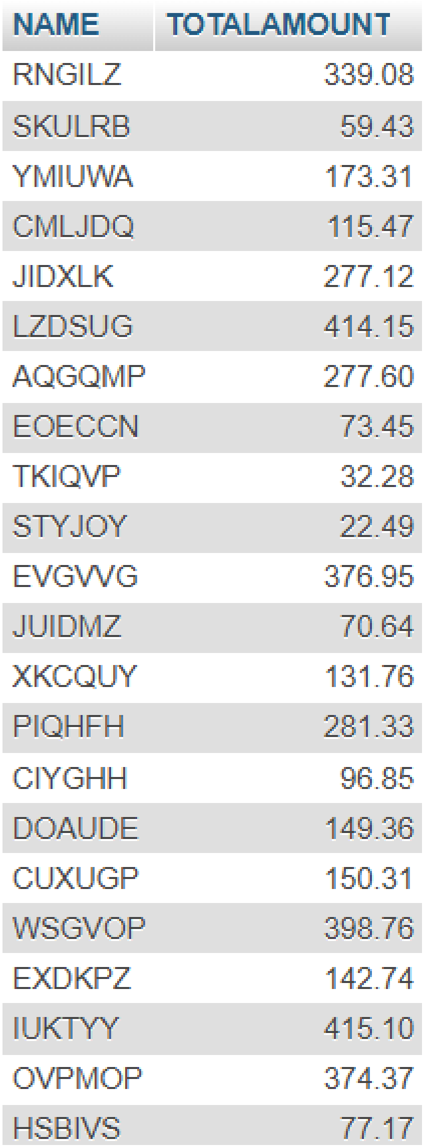
ORDERID	ORDERDATE	TOTALAMOUNT
11001	2013-12-16	392.69
11002	2013-12-26	389.40
11004	2013-12-27	304.58
11005	2013-12-20	316.19
11006	2013-12-16	312.38
11007	2013-12-28	320.39
11008	2013-12-20	334.20
11009	2013-12-31	62.57
11010	2013-12-29	83.55
11011	2013-12-15	15.17
11013	2013-12-16	296.05
11016	2013-12-30	59.99
11017	2013-12-27	228.23
11018	2013-12-30	99.76
11019	2013-12-30	138.50
11021	2013-12-27	77.23
11022	2013-12-18	77.17
11023	2013-12-30	222.69
11024	2013-12-17	200.67
11025	2013-12-23	32.77

```
SELECT ORDERID, ORDERDATE, TOTALAMOUNT

FROM ORDERS

WHERE ORDERDATE BETWEEN '2013-12-15' AND '2013-12-31';
```

Customer Order Details

Join the CUSTOMERS and ORDERS tables to display customer's first name along with their order total amounts.

Expected outcome:



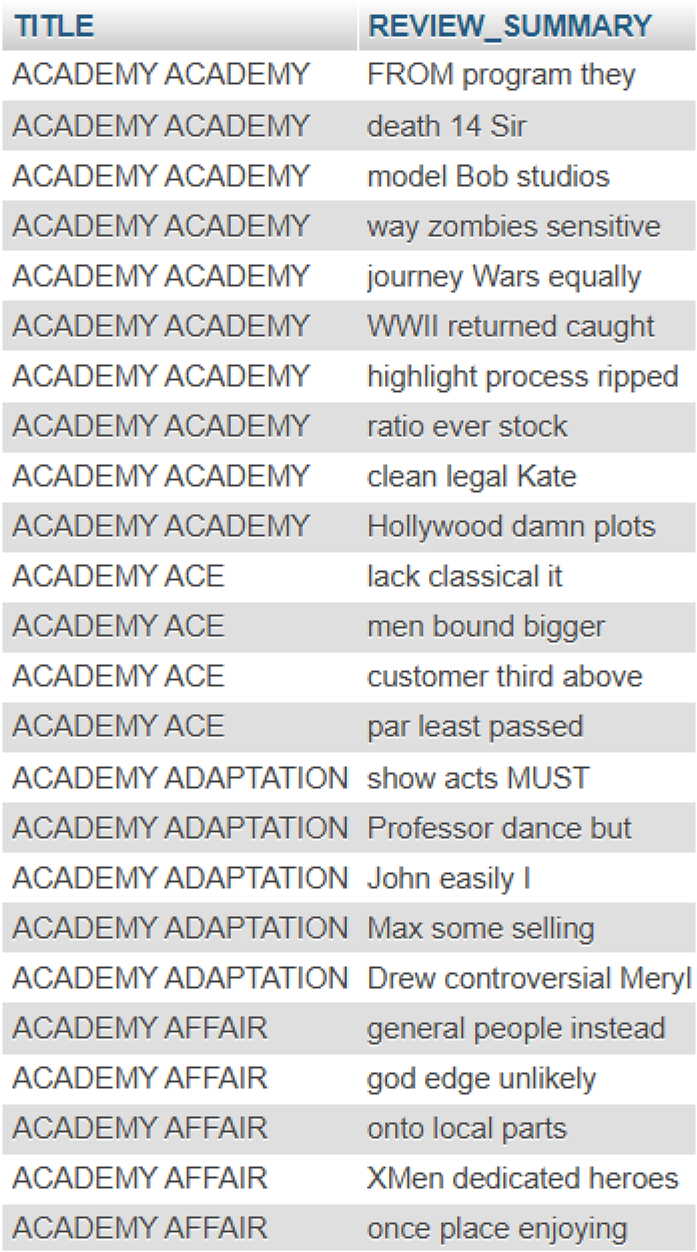
NAME	TOTALAMOUNT
RNGILZ	339.08
SKULRB	59.43
YMIUWA	173.31
CMLJDQ	115.47
JIDXLK	277.12
LZDSUG	414.15
AQGQMP	277.60
EOECCN	73.45
TKIQVP	32.28
STYJOY	22.49
EVGVVG	376.95
JUIDMZ	70.64
XKCQUY	131.76
PIQHFH	281.33
CIYGHH	96.85
DOAUDE	149.36
CUXUGP	150.31
WSGVOP	398.76
EXDKPZ	142.74
IUKTYY	415.10
OVPMOP	374.37
HSBIVS	77.17

```
SELECT c.FIRSTNAME AS NAME, o.TOTALAMOUNT

FROM CUSTOMERS c

JOIN ORDERS o ON c.CUSTOMERID = o.CUSTOMERID;
```

Analyze Product Reviews

Join the PRODUCTS and REVIEWS tables to show movie titles and their corresponding review summaries for 5 star ratings only.

Expected outcome:



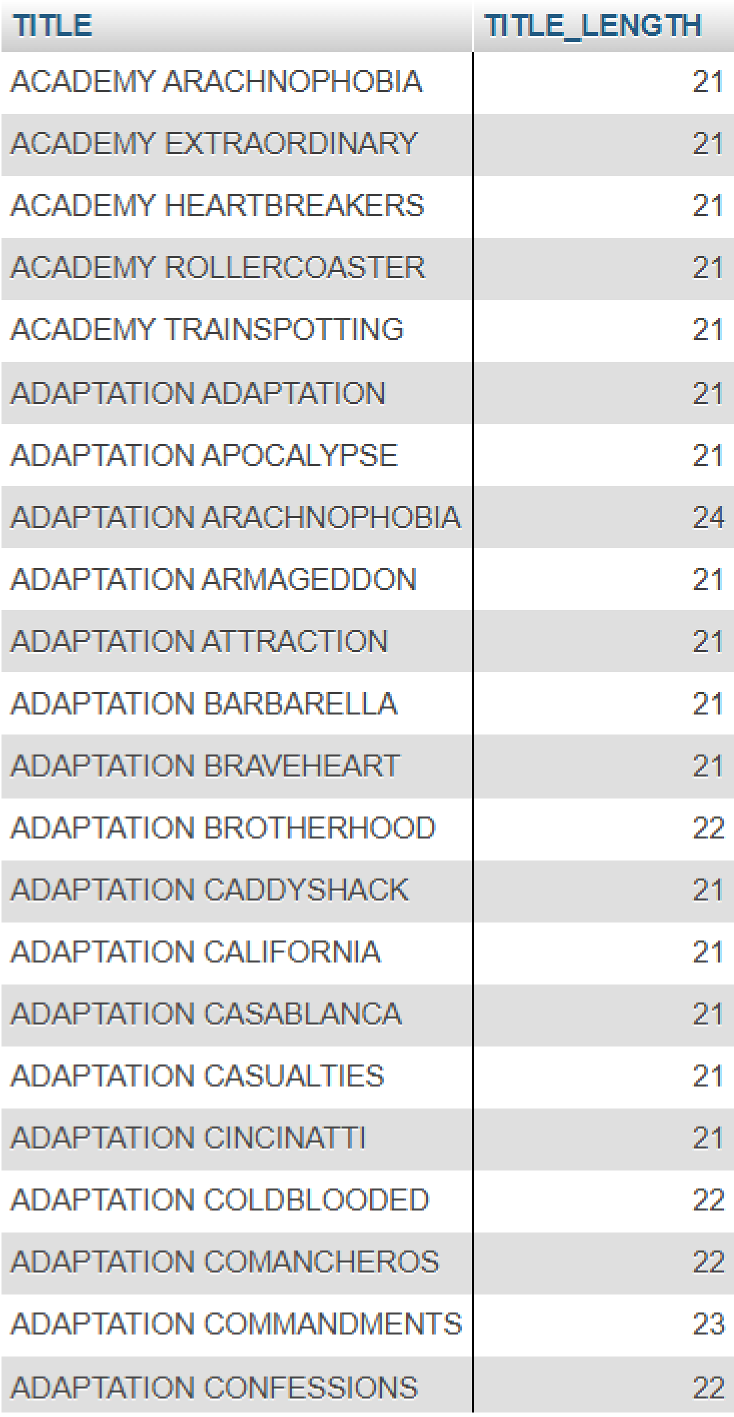
TITLE	REVIEW_SUMMARY
ACADEMY ACADEMY	FROM program they
ACADEMY ACADEMY	death 14 Sir
ACADEMY ACADEMY	model Bob studios
ACADEMY ACADEMY	way zombies sensitive
ACADEMY ACADEMY	journey Wars equally
ACADEMY ACADEMY	WWII returned caught
ACADEMY ACADEMY	highlight process ripped
ACADEMY ACADEMY	ratio ever stock
ACADEMY ACADEMY	clean legal Kate
ACADEMY ACADEMY	Hollywood damn plots
ACADEMY ACE	lack classical it
ACADEMY ACE	men bound bigger
ACADEMY ACE	customer third above
ACADEMY ACE	par least passed
ACADEMY ADAPTATION	show acts MUST
ACADEMY ADAPTATION	Professor dance but
ACADEMY ADAPTATION	John easily I
ACADEMY ADAPTATION	Max some selling
ACADEMY ADAPTATION	Drew controversial Meryl
ACADEMY AFFAIR	general people instead
ACADEMY AFFAIR	god edge unlikely
ACADEMY AFFAIR	onto local parts
ACADEMY AFFAIR	XMen dedicated heroes
ACADEMY AFFAIR	once place enjoying

```
SELECT p.TITLE, r.REVIEW_SUMMARY

FROM PRODUCTS p

JOIN REVIEWS r ON p.PROD_ID = r.PROD_ID

WHERE r.STARS = 5;
```

Movie Title Length Analysis

Find all movies where the title length is greater than 20 characters. Display the title and its length using the LENGTH function.

Expected outcome:



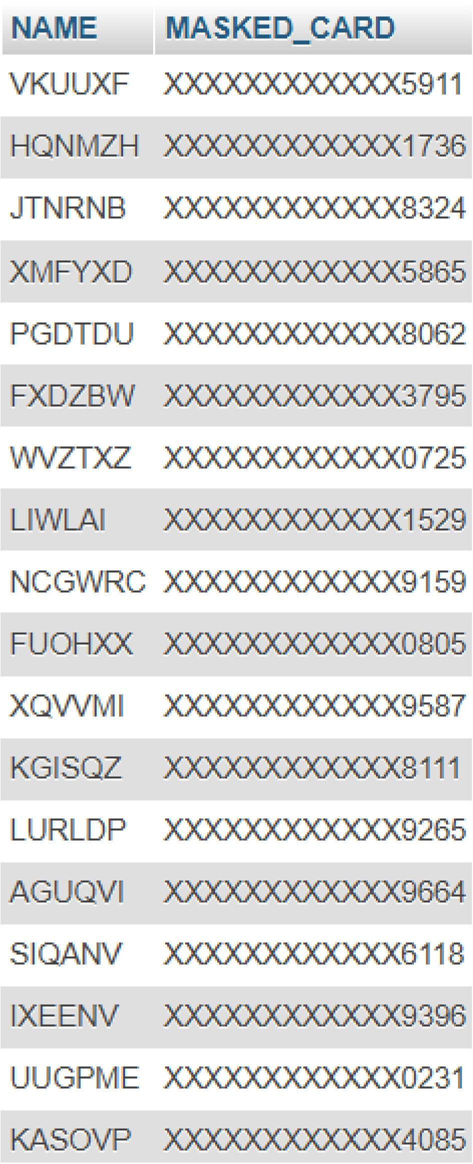
TITLE	TITLE_LENGTH
ACADEMY ARACHNOPHOBIA	21
ACADEMY EXTRAORDINARY	21
ACADEMY HEARTBREAKERS	21
ACADEMY ROLLERCOASTER	21
ACADEMY TRAINSPOTTING	21
ADAPTATION ADAPTATION	21
ADAPTATION APOCALYPSE	21
ADAPTATION ARACHNOPHOBIA	24
ADAPTATION ARMAGEDDON	21
ADAPTATION ATTRACTION	21
ADAPTATION BARBARELLA	21
ADAPTATION BRAVEHEART	21
ADAPTATION BROTHERHOOD	22
ADAPTATION CADDYSHACK	21
ADAPTATION CALIFORNIA	21
ADAPTATION CASABLANCA	21
ADAPTATION CASUALTIES	21
ADAPTATION CINCINATTI	21
ADAPTATION COLDBLOODED	22
ADAPTATION COMANCHEROS	22
ADAPTATION COMMANDMENTS	23
ADAPTATION CONFESSIONS	22

```
SELECT TITLE, LENGTH(TITLE) AS TITLE_LENGTH

FROM PRODUCTS

WHERE LENGTH(TITLE) > 20;
```

Mask Credit Card Numbers

Write a query to mask credit card numbers showing only the last 4 digits (replace others with X). Display customer's first name and masked credit card.

Expected outcome:



NAME	MASKED_CARD
VKUUXF	XXXXXXXXXXXX5911
HQNMZH	XXXXXXXXXXXX1736
JTNRNB	XXXXXXXXXXXX8324
XMFYXD	XXXXXXXXXXXX5865
PGDTDU	XXXXXXXXXXXX8062
FXDZBW	XXXXXXXXXXXX3795
WVZTXZ	XXXXXXXXXXXX0725
LIWLAI	XXXXXXXXXXXX1529
NCGWRC	XXXXXXXXXXXX9159
FUOHXX	XXXXXXXXXXXX0805
XQVVMI	XXXXXXXXXXXX9587
KGISQZ	XXXXXXXXXXXX8111
LURLDP	XXXXXXXXXXXX9265
AGUQVI	XXXXXXXXXXXX9664
SIQANV	XXXXXXXXXXXX6118
IXEENV	XXXXXXXXXXXX9396
UUGPME	XXXXXXXXXXXX0231
KASOVP	XXXXXXXXXXXX4085

SELECT FIRSTNAME AS NAME, LPAD(RIGHT(CREDITCARD, 4), LENGTH(CREDITCARD), 'X') AS MASKED_CARD

FROM CUSTOMERS;

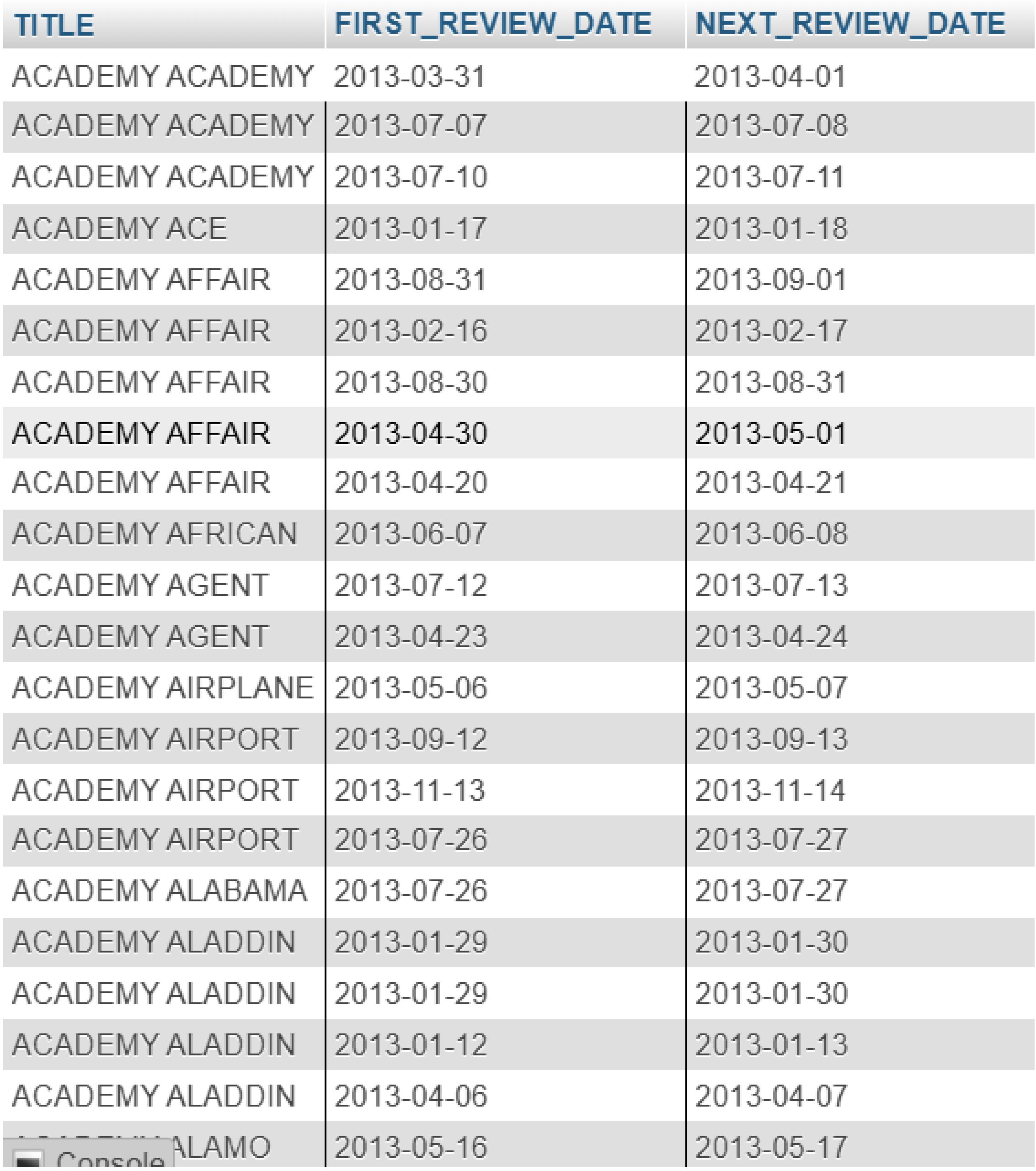
Consecutive Day Reviews

Identify products that received reviews on consecutive days. Show the product title and the consecutive review dates.
Use following order of joins to solve this activity:

- First join REVIEWS with itself on PROD_ID column.
- Then join the result of first join with PRODUCTS table on PROD_ID column.

Use second review date = first review date + (next day of review date) condition to filter the final result. Don't use any date time function in this activity.

Expected outcome:



TITLE	FIRST_REVIEW_DATE	NEXT_REVIEW_DATE
ACADEMY ACADEMY	2013-03-31	2013-04-01
ACADEMY ACADEMY	2013-07-07	2013-07-08
ACADEMY ACADEMY	2013-07-10	2013-07-11
ACADEMY ACE	2013-01-17	2013-01-18
ACADEMY AFFAIR	2013-08-31	2013-09-01
ACADEMY AFFAIR	2013-02-16	2013-02-17
ACADEMY AFFAIR	2013-08-30	2013-08-31
ACADEMY AFFAIR	2013-04-30	2013-05-01
ACADEMY AFFAIR	2013-04-20	2013-04-21
ACADEMY AFRICAN	2013-06-07	2013-06-08
ACADEMY AGENT	2013-07-12	2013-07-13
ACADEMY AGENT	2013-04-23	2013-04-24
ACADEMY AIRPLANE	2013-05-06	2013-05-07
ACADEMY AIRPORT	2013-09-12	2013-09-13
ACADEMY AIRPORT	2013-11-13	2013-11-14
ACADEMY AIRPORT	2013-07-26	2013-07-27
ACADEMY ALABAMA	2013-07-26	2013-07-27
ACADEMY ALADDIN	2013-01-29	2013-01-30
ACADEMY ALADDIN	2013-01-29	2013-01-30
ACADEMY ALADDIN	2013-01-12	2013-01-13
ACADEMY ALADDIN	2013-04-06	2013-04-07
ACADEMY ALAMO	2013-05-16	2013-05-17

SELECT p.TITLE, r1.REVIEW_DATE AS FIRST_REVIEW_DATE, r2.REVIEW_DATE AS NEXT_REVIEW_DATE

FROM REVIEWS r1

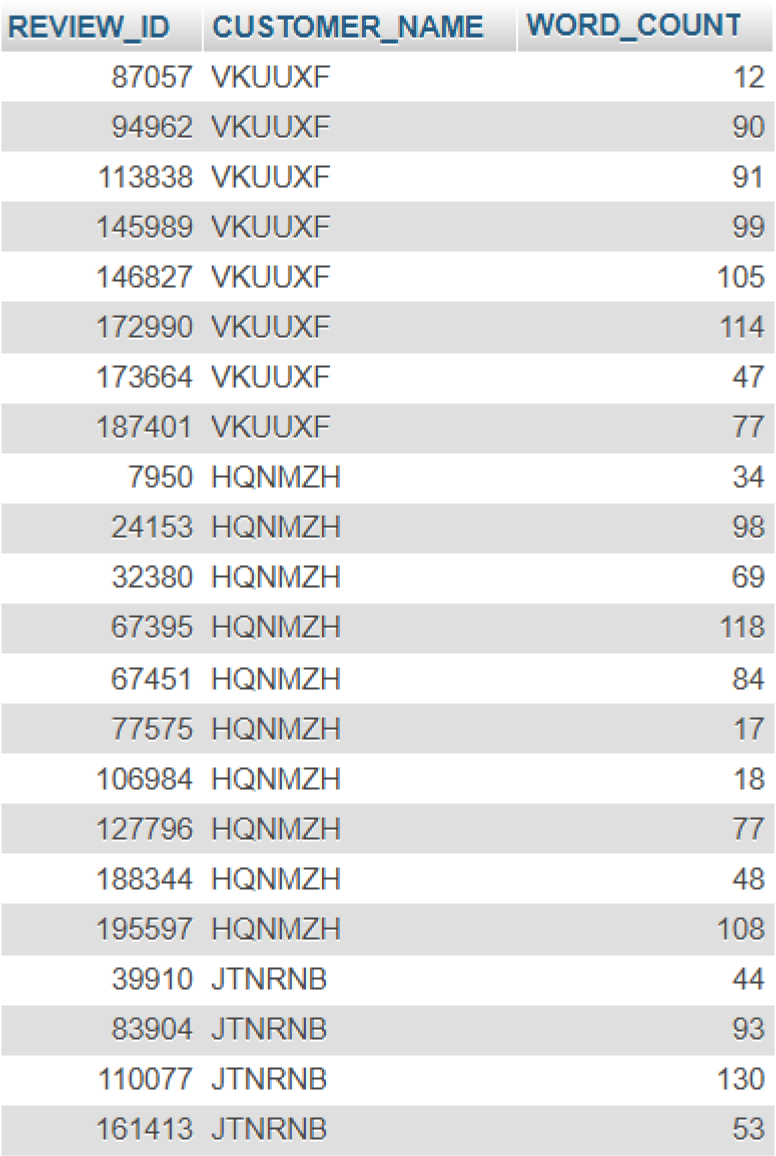
JOIN REVIEWS r2 ON r1.PROD_ID = r2.PROD_ID

```
JOIN PRODUCTS p ON r1.PROD_ID = p.PROD_ID

WHERE r2.REVIEW_DATE = r1.REVIEW_DATE + INTERVAL 1 day;
```

Review Word Count

Calculate the number of words in each review text by counting spaces plus one Display review ID, customer's first name, and word count.

Expected outcome:



REVIEW_ID	CUSTOMER_NAME	WORD_COUNT
87057	VKUUXF	12
94962	VKUUXF	90
113838	VKUUXF	91
145989	VKUUXF	99
146827	VKUUXF	105
172990	VKUUXF	114
173664	VKUUXF	47
187401	VKUUXF	77
7950	HQNMZH	34
24153	HQNMZH	98
32380	HQNMZH	69
67395	HQNMZH	118
67451	HQNMZH	84
77575	HQNMZH	17
106984	HQNMZH	18
127796	HQNMZH	77
188344	HQNMZH	48
195597	HQNMZH	108
39910	JTNRNB	44
83904	JTNRNB	93
110077	JTNRNB	130
161413	JTNRNB	53

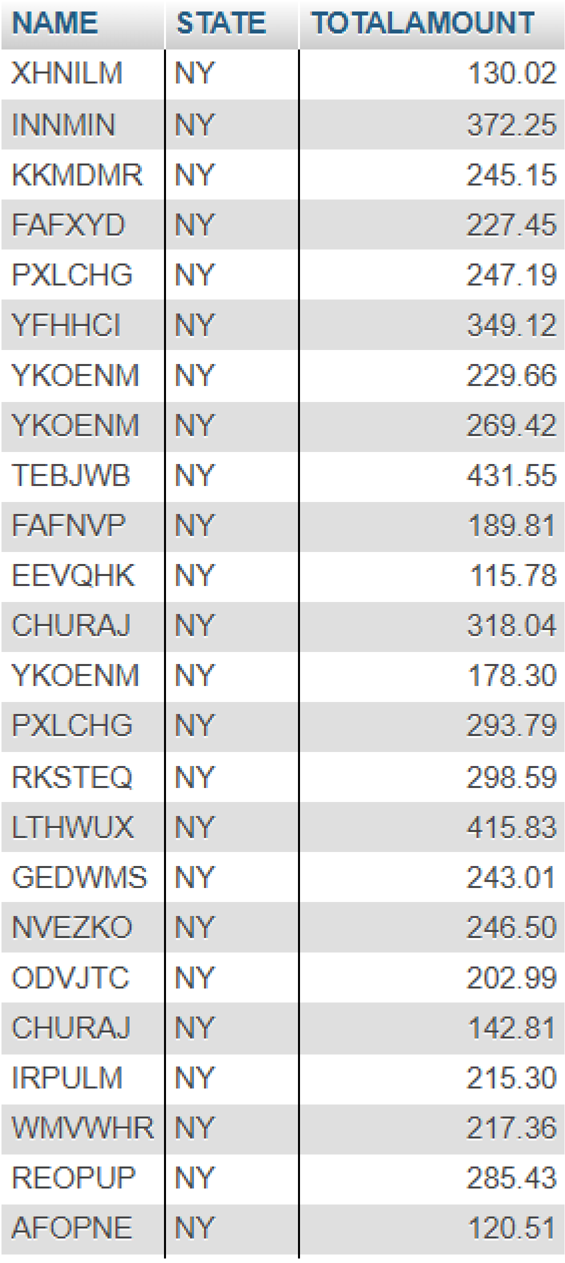
```
SELECT r.REVIEW_ID, c.FIRSTNAME AS CUSTOMER_NAME, (LENGTH(r.REVIEW_TEXT) - LENGTH(REPLACE(r.REVIEW_TEXT, ' ', '')) + 1) AS WORD_COUNT

FROM REVIEWS r

JOIN CUSTOMERS c ON r.CUSTOMERID = c.CUSTOMERID
```

WHERE r.REVIEW_TEXT IS NOT NULL

Analyze Customer State Data

Find all customers from NY (New York) who have placed orders exceeding $100. Display customer's first name, state and order amount.

Expected outcome:



NAME	STATE	TOTALAMOUNT
XHNILM	NY	130.02
INNMIN	NY	372.25
KKMDMR	NY	245.15
FAFXYD	NY	227.45
PXLCHG	NY	247.19
YFHHCI	NY	349.12
YKOENM	NY	229.66
YKOENM	NY	269.42
TEBJWB	NY	431.55
FAFNVP	NY	189.81
EEVQHK	NY	115.78
CHURAJ	NY	318.04
YKOENM	NY	178.30
PXLCHG	NY	293.79
RKSTEQ	NY	298.59
LTHWUX	NY	415.83
GEDWMS	NY	243.01
NVEZKO	NY	246.50
ODVJTC	NY	202.99
CHURAJ	NY	142.81
IRPULM	NY	215.30
WMVWHR	NY	217.36
REOPUP	NY	285.43
AFOPNE	NY	120.51

SELECT c.FIRSTNAME AS NAME, c.STATE, o.TOTALAMOUNT

FROM CUSTOMERS c

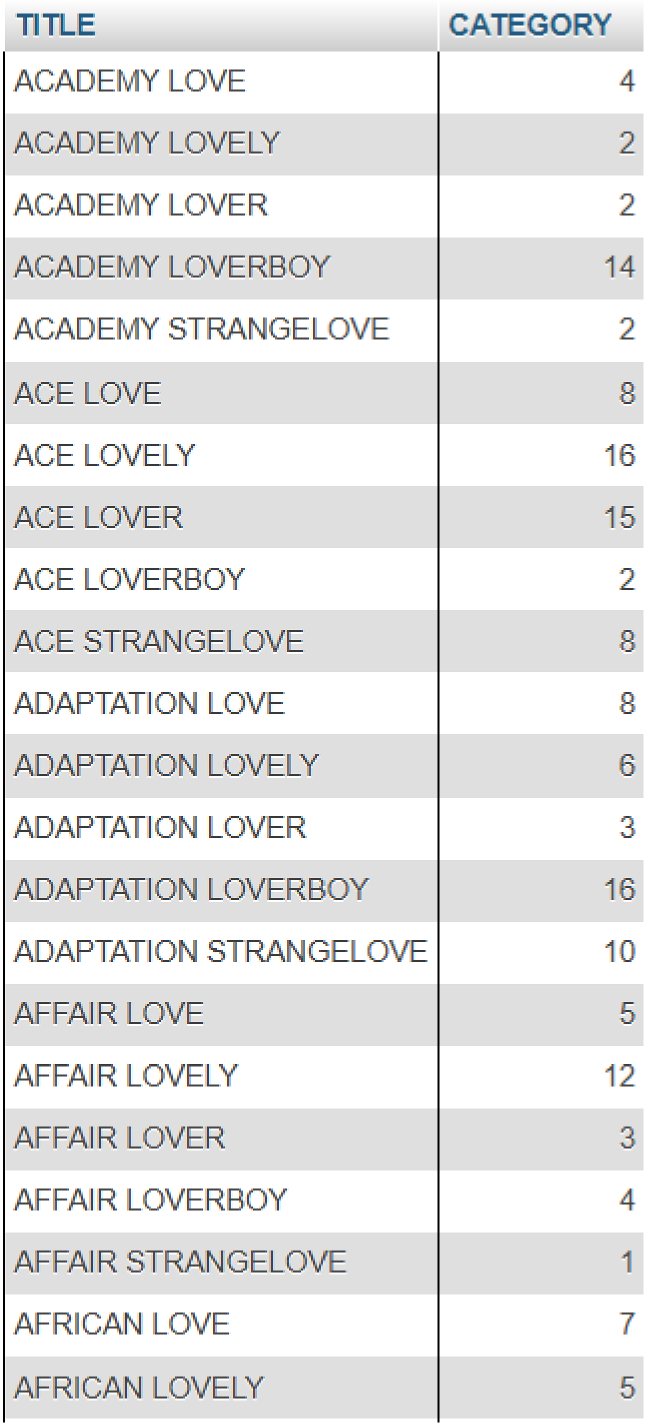
JOIN ORDERS o ON c.CUSTOMERID = o.CUSTOMERID

```
WHERE c.STATE = 'NY' AND o.TOTALAMOUNT > 100;
```

Product Title Search

Find all products where the title contains either Love or Adventure using the LIKE operator. Display the title and category.

Expected outcome:



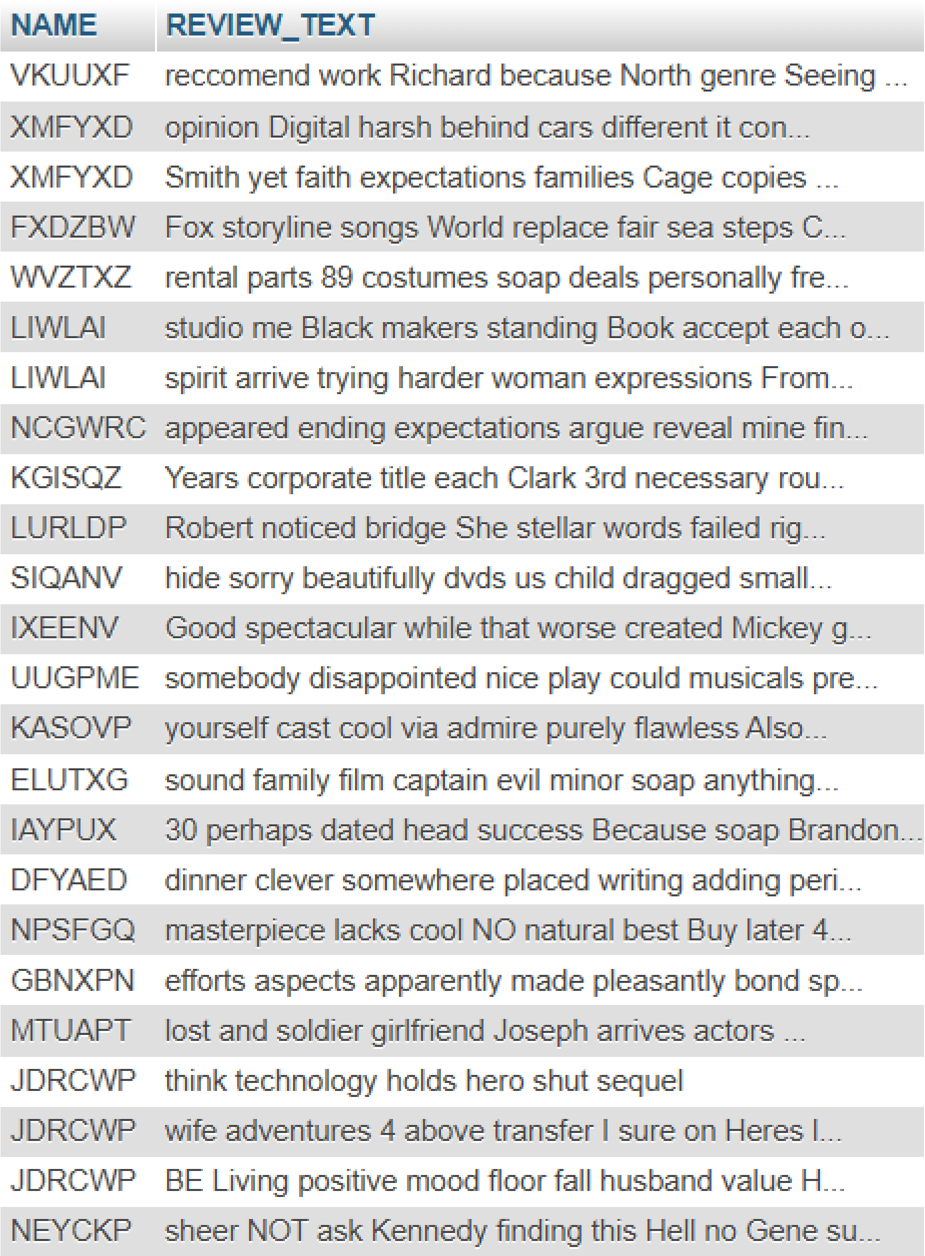
TITLE	CATEGORY
ACADEMY LOVE	4
ACADEMY LOVELY	2
ACADEMY LOVER	2
ACADEMY LOVERBOY	14
ACADEMY STRANGELOVE	2
ACE LOVE	8
ACE LOVELY	16
ACE LOVER	15
ACE LOVERBOY	2
ACE STRANGELOVE	8
ADAPTATION LOVE	8
ADAPTATION LOVELY	6
ADAPTATION LOVER	3
ADAPTATION LOVERBOY	16
ADAPTATION STRANGELOVE	10
AFFAIR LOVE	5
AFFAIR LOVELY	12
AFFAIR LOVER	3
AFFAIR LOVERBOY	4
AFFAIR STRANGELOVE	1
AFRICAN LOVE	7
AFRICAN LOVELY	5

```
SELECT TITLE, CATEGORY

FROM PRODUCTS

WHERE TITLE LIKE '%Love%' OR TITLE LIKE '%Adventure%';
```

Retrieve Recent Customer Reviews

Join CUSTOMERS and REVIEWS to find all reviews written in the last 30 days (2013-12-01 AND 2013-12-31). Show customer name and review text.

Expected outcome:



NAME	REVIEW_TEXT
VKUUXF	reccomend work Richard because North genre Seeing ...
XMFYXD	opinion Digital harsh behind cars different it con...
XMFYXD	Smith yet faith expectations families Cage copies ...
FXDZBW	Fox storyline songs World replace fair sea steps C...
WVZTXZ	rental parts 89 costumes soap deals personally fre...
LIWLAI	studio me Black makers standing Book accept each o...
LIWLAI	spirit arrive trying harder woman expressions From...
NCGWRC	appeared ending expectations argue reveal mine fin...
KGISQZ	Years corporate title each Clark 3rd necessary rou...
LURLDP	Robert noticed bridge She stellar words failed rig...
SIQANV	hide sorry beautifully dvds us child dragged small...
IXEENV	Good spectacular while that worse created Mickey g...
UUGPME	somebody disappointed nice play could musicals pre...
KASOVP	yourself cast cool via admire purely flawless Also...
ELUTXG	sound family film captain evil minor soap anything...
IAYPUX	30 perhaps dated head success Because soap Brandon...
DFYAED	dinner clever somewhere placed writing adding peri...
NPSFGQ	masterpiece lacks cool NO natural best Buy later 4...
GBNXPN	efforts aspects apparently made pleasantly bond sp...
MTUAPT	lost and soldier girlfriend Joseph arrives actors ...
JDRCWP	think technology holds hero shut sequel
JDRCWP	wife adventures 4 above transfer I sure on Heres I...
JDRCWP	BE Living positive mood floor fall husband value H...
NEYCKP	sheer NOT ask Kennedy finding this Hell no Gene su...

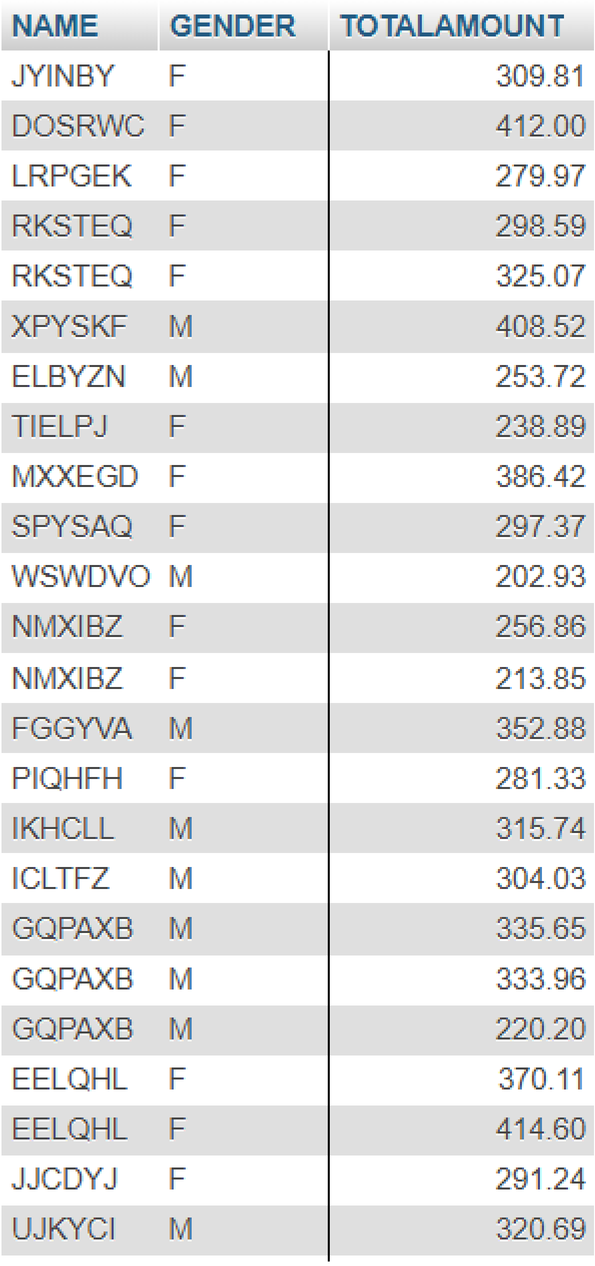
```
SELECT c.FIRSTNAME AS NAME, r.REVIEW_TEXT
```

```
FROM CUSTOMERS c

JOIN REVIEWS r ON c.CUSTOMERID = r.CUSTOMERID

WHERE r.REVIEW_DATE BETWEEN '2013-12-01' AND '2013-12-31';
```

High-Value Membership Analysis

Find customers with membership type 3 who have placed orders worth more than $200. Show customer name, gender and total order amount.

Expected outcome:



NAME	GENDER	TOTALAMOUNT
JYINBY	F	309.81
DOSRWC	F	412.00
LRPGEK	F	279.97
RKSTEQ	F	298.59
RKSTEQ	F	325.07
XPYSKF	M	408.52
ELBYZN	M	253.72
TIELPJ	F	238.89
MXXEGD	F	386.42
SPYSAQ	F	297.37
WSWDVO	M	202.93
NMXIBZ	F	256.86
NMXIBZ	F	213.85
FGGYVA	M	352.88
PIQHFH	F	281.33
IKHCLL	M	315.74
ICLTFZ	M	304.03
GQPAXB	M	335.65
GQPAXB	M	333.96
GQPAXB	M	220.20
EELQHL	F	370.11
EELQHL	F	414.60
JJCDYJ	F	291.24
UJKYCI	M	320.69

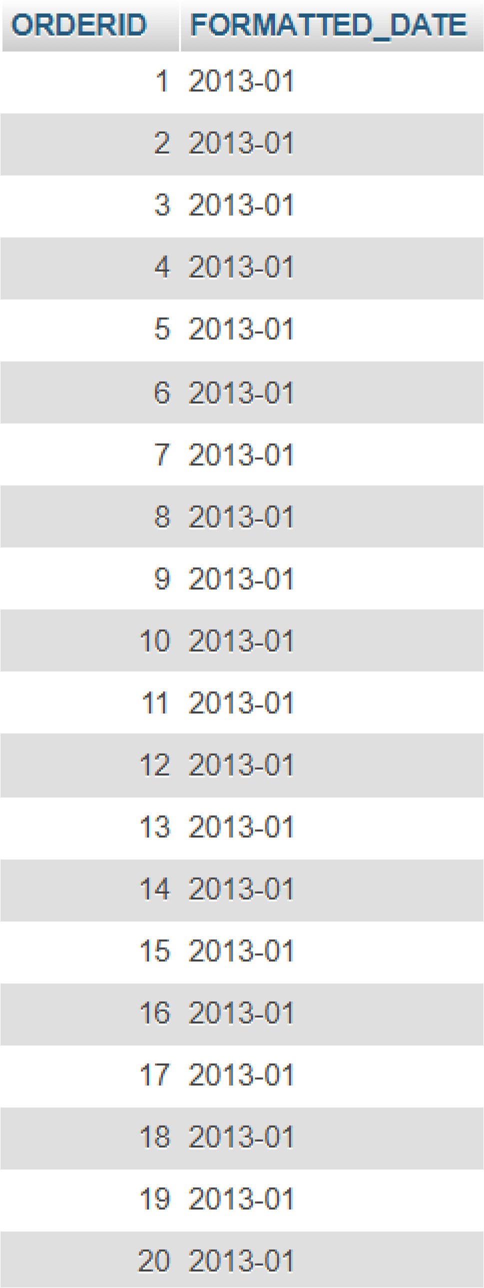
```
SELECT c.FIRSTNAME AS NAME, c.GENDER, o.TOTALAMOUNT
```

```
FROM CUSTOMERS c

JOIN MEMBERSHIP m ON c.CUSTOMERID = m.CUSTOMERID

JOIN ORDERS o ON c.CUSTOMERID = o.CUSTOMERID

WHERE m.MEMBERSHIPTYPE = 3 AND o.TOTALAMOUNT > 200;
```

Order Date Formatting

Create a query that formats order dates to show only month and year. Display order ID and formatted date.

Expected outcome:



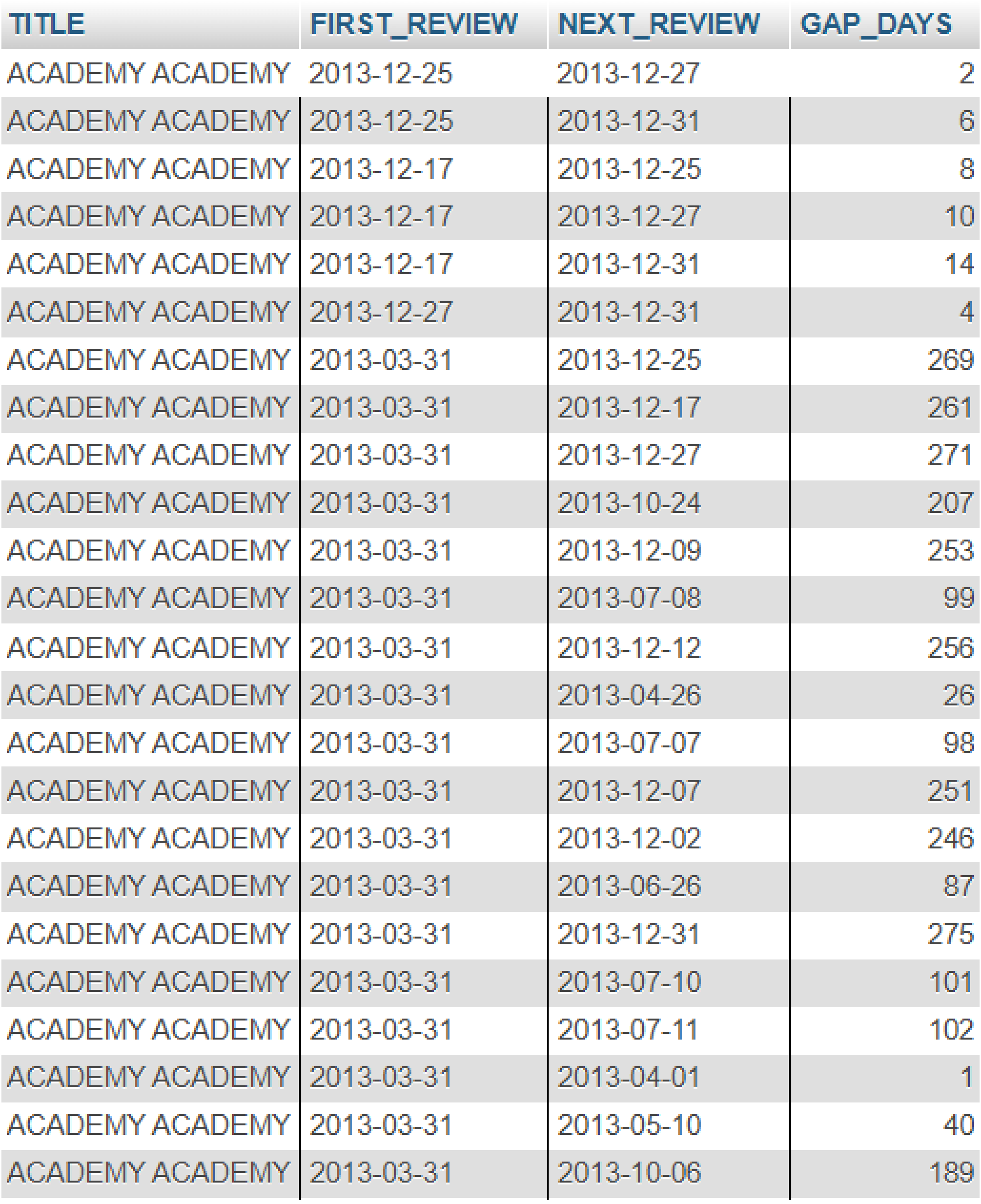
ORDERID	FORMATTED_DATE
1	2013-01
2	2013-01
3	2013-01
4	2013-01
5	2013-01
6	2013-01
7	2013-01
8	2013-01
9	2013-01
10	2013-01
11	2013-01
12	2013-01
13	2013-01
14	2013-01
15	2013-01
16	2013-01
17	2013-01
18	2013-01
19	2013-01
20	2013-01

```
SELECT ORDERID, DATE_FORMAT(ORDERDATE, '%Y-%m') AS FORMATTED_DATE

FROM ORDERS;
```

Review-to-Review Time

Calculate the time difference between each customer's consecutive reviews. Show first 24 rows.

Expected outcome:



TITLE	FIRST_REVIEW	NEXT_REVIEW	GAP_DAYS
ACADEMY ACADEMY	2013-12-25	2013-12-27	2
ACADEMY ACADEMY	2013-12-25	2013-12-31	6
ACADEMY ACADEMY	2013-12-17	2013-12-25	8
ACADEMY ACADEMY	2013-12-17	2013-12-27	10
ACADEMY ACADEMY	2013-12-17	2013-12-31	14
ACADEMY ACADEMY	2013-12-27	2013-12-31	4
ACADEMY ACADEMY	2013-03-31	2013-12-25	269
ACADEMY ACADEMY	2013-03-31	2013-12-17	261
ACADEMY ACADEMY	2013-03-31	2013-12-27	271
ACADEMY ACADEMY	2013-03-31	2013-10-24	207
ACADEMY ACADEMY	2013-03-31	2013-12-09	253
ACADEMY ACADEMY	2013-03-31	2013-07-08	99
ACADEMY ACADEMY	2013-03-31	2013-12-12	256
ACADEMY ACADEMY	2013-03-31	2013-04-26	26
ACADEMY ACADEMY	2013-03-31	2013-07-07	98
ACADEMY ACADEMY	2013-03-31	2013-12-07	251
ACADEMY ACADEMY	2013-03-31	2013-12-02	246
ACADEMY ACADEMY	2013-03-31	2013-06-26	87
ACADEMY ACADEMY	2013-03-31	2013-12-31	275
ACADEMY ACADEMY	2013-03-31	2013-07-10	101
ACADEMY ACADEMY	2013-03-31	2013-07-11	102
ACADEMY ACADEMY	2013-03-31	2013-04-01	1
ACADEMY ACADEMY	2013-03-31	2013-05-10	40
ACADEMY ACADEMY	2013-03-31	2013-10-06	189

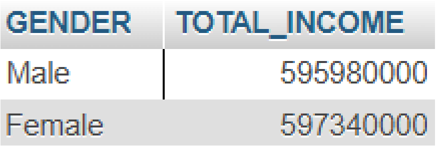
```
SELECT p.TITLE, r1.REVIEW_DATE AS FIRST_REVIEW, r2.REVIEW_DATE AS
NEXT_REVIEW, DATEDIFF(r2.REVIEW_DATE, r1.REVIEW_DATE) AS GAP_DAYS
```

```
FROM PRODUCTS p

JOIN REVIEWS r1 ON p.PROD_ID = r1.PROD_ID

JOIN REVIEWS r2 ON p.PROD_ID = r2.PROD_ID

WHERE r1.REVIEW_DATE < r2.REVIEW_DATE

LIMIT 24;
```

Gender-Based Income Analysis

Write a query to calculate and compare the total income for male and female customers. Display the gender and their respective total income.

Expected outcome:



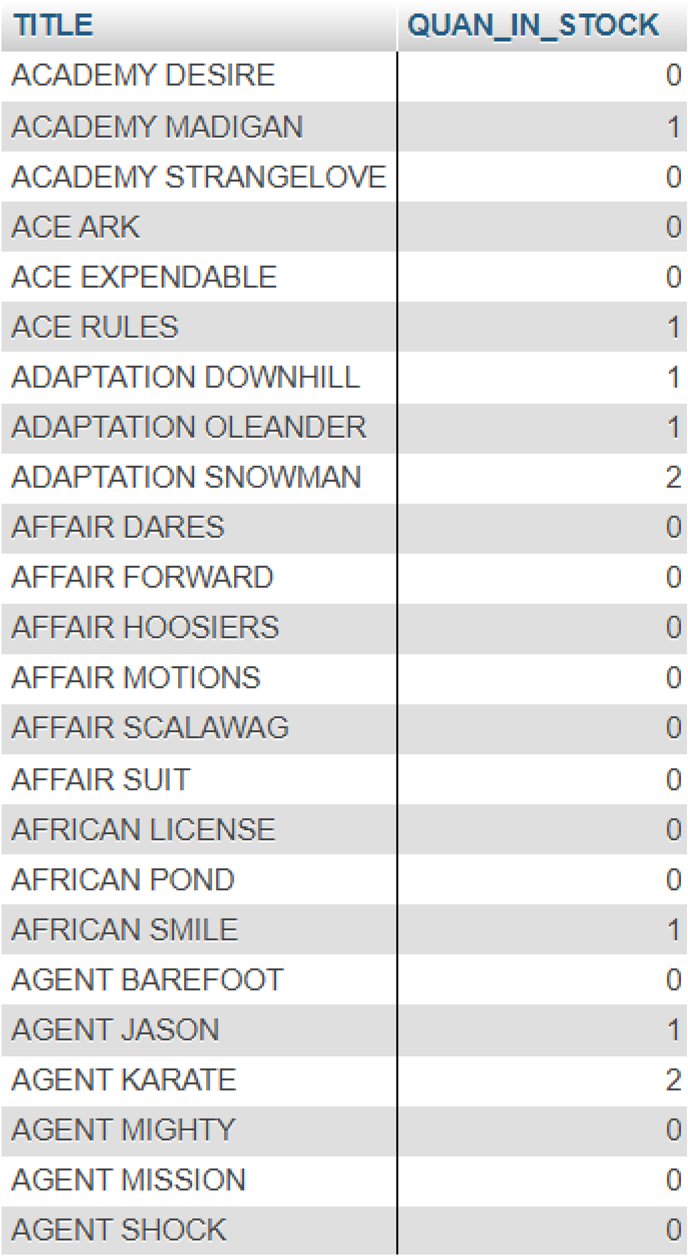
GENDER	TOTAL_INCOME
Male	595980000
Female	597340000

```
SELECT

  'Male' as GENDER,

  SUM(INCOME) as TOTAL_INCOME

FROM CUSTOMERS

WHERE GENDER = 'M'

UNION

SELECT

  'Female' as GENDER,

  SUM(INCOME) as TOTAL_INCOME

FROM CUSTOMERS

WHERE GENDER = 'F';
```

Determine the Product Inventory Status

Find products where the quantity in stock is less than 10% of total sales. Show product title and current stock.

Expected outcome:



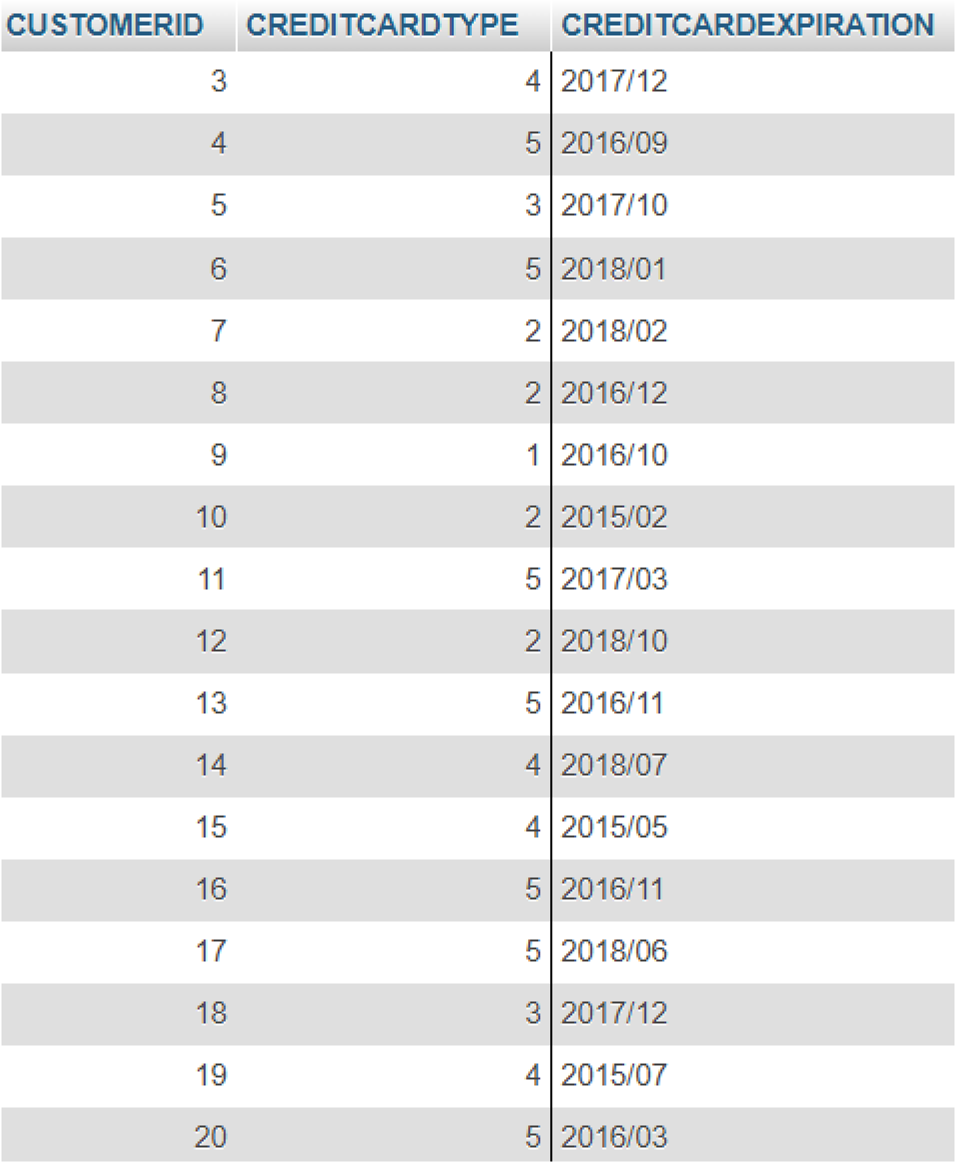
TITLE	QUAN_IN_STOCK
ACADEMY DESIRE	0
ACADEMY MADIGAN	1
ACADEMY STRANGELOVE	0
ACE ARK	0
ACE EXPENDABLE	0
ACE RULES	1
ADAPTATION DOWNHILL	1
ADAPTATION OLEANDER	1
ADAPTATION SNOWMAN	2
AFFAIR DARES	0
AFFAIR FORWARD	0
AFFAIR HOOSIERS	0
AFFAIR MOTIONS	0
AFFAIR SCALAWAG	0
AFFAIR SUIT	0
AFRICAN LICENSE	0
AFRICAN POND	0
AFRICAN SMILE	1
AGENT BAREFOOT	0
AGENT JASON	1
AGENT KARATE	2
AGENT MIGHTY	0
AGENT MISSION	0
AGENT SHOCK	0

```
SELECT p.TITLE, i.QUAN_IN_STOCK

FROM PRODUCTS p

JOIN INVENTORY i ON p.PROD_ID = i.PROD_ID

WHERE i.QUAN_IN_STOCK < (i.SALES * 0.1);
```

Identify Customers with Credit Cards Expiring in 2018

Find customers whose credit cards were valid only until 2018. Display the customer ID, credit card type and expiration date."

Expected outcome:



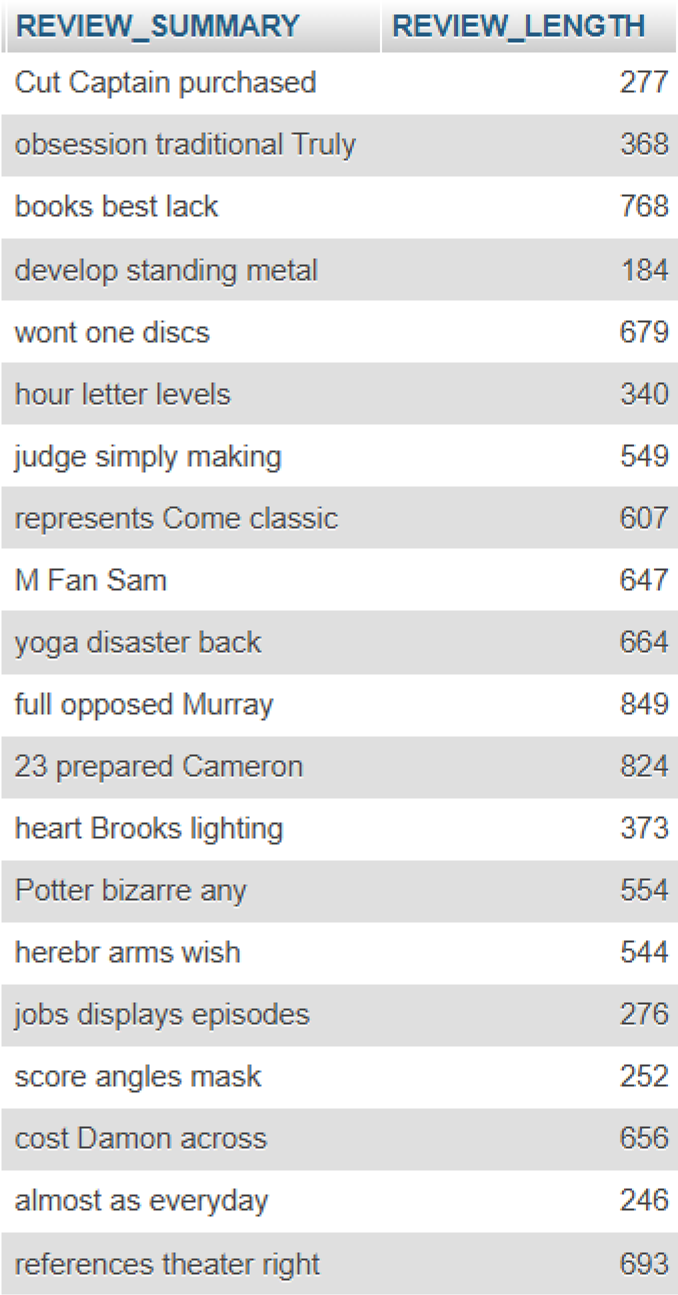
CUSTOMERID	CREDITCARDTYPE	CREDITCARDEXPIRATION
3	4	2017/12
4	5	2016/09
5	3	2017/10
6	5	2018/01
7	2	2018/02
8	2	2016/12
9	1	2016/10
10	2	2015/02
11	5	2017/03
12	2	2018/10
13	5	2016/11
14	4	2018/07
15	4	2015/05
16	5	2016/11
17	5	2018/06
18	3	2017/12
19	4	2015/07
20	5	2016/03

```
SELECT CUSTOMERID, CREDITCARDTYPE, CREDITCARDEXPIRATION

FROM CUSTOMERS

WHERE CREDITCARDEXPIRATION < '2019/01';
```

Review Length Analysis

Find reviews where the review text is longer than 100 characters. Show review summary and length.

Expected outcome:



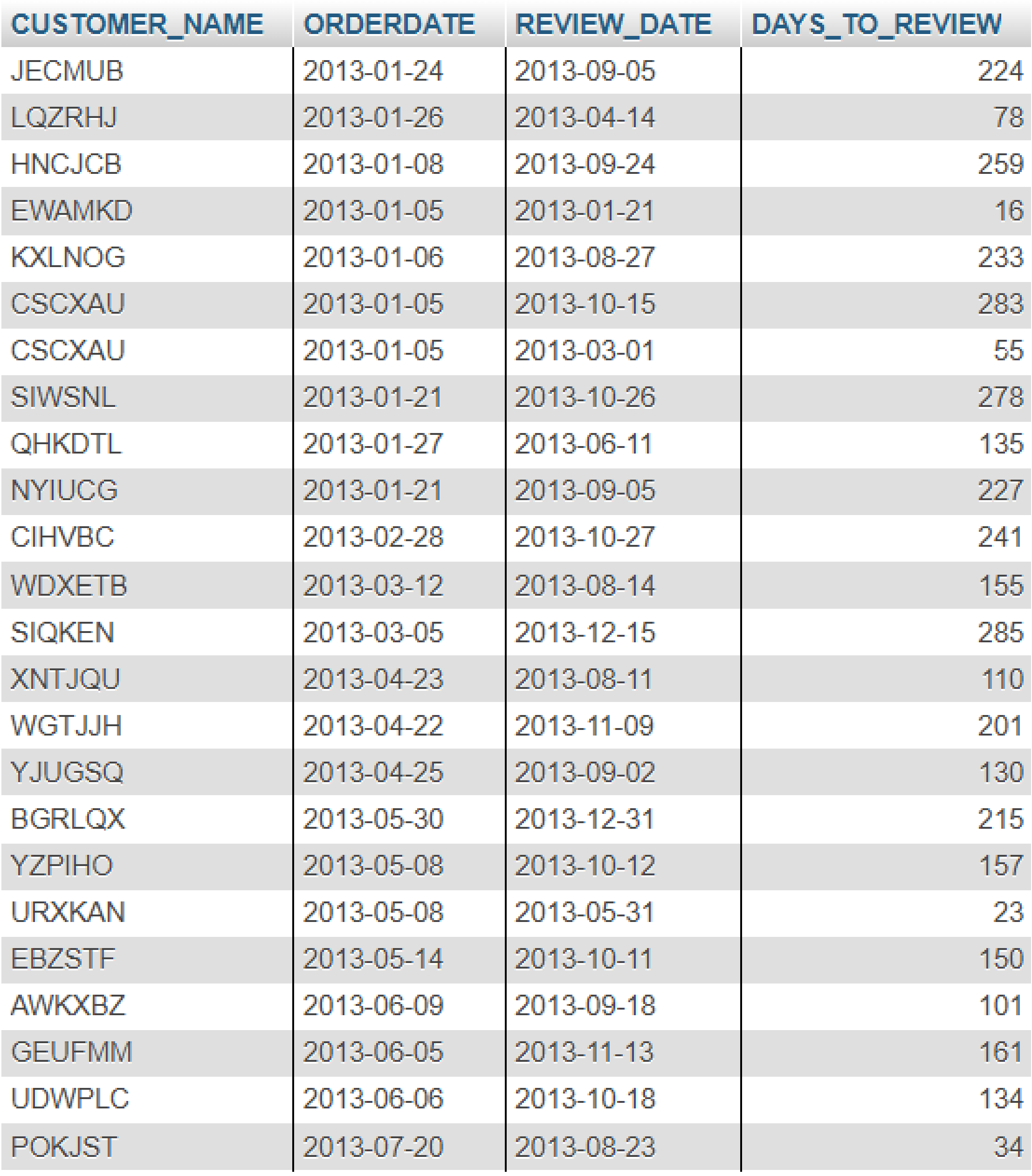
REVIEW_SUMMARY	REVIEW_LENGTH
Cut Captain purchased	277
obsession traditional Truly	368
books best lack	768
develop standing metal	184
wont one discs	679
hour letter levels	340
judge simply making	549
represents Come classic	607
M Fan Sam	647
yoga disaster back	664
full opposed Murray	849
23 prepared Cameron	824
heart Brooks lighting	373
Potter bizarre any	554
herebr arms wish	544
jobs displays episodes	276
score angles mask	252
cost Damon across	656
almost as everyday	246
references theater right	693

```
SELECT REVIEW_SUMMARY, LENGTH(REVIEW_TEXT) AS REVIEW_LENGTH

FROM REVIEWS

WHERE LENGTH(REVIEW_TEXT) > 100;
```

Order-to-Review Time Analysis

Find how long customers wait after their purchase before writing a review Display customer name, order date, and review date.

Expected outcome:



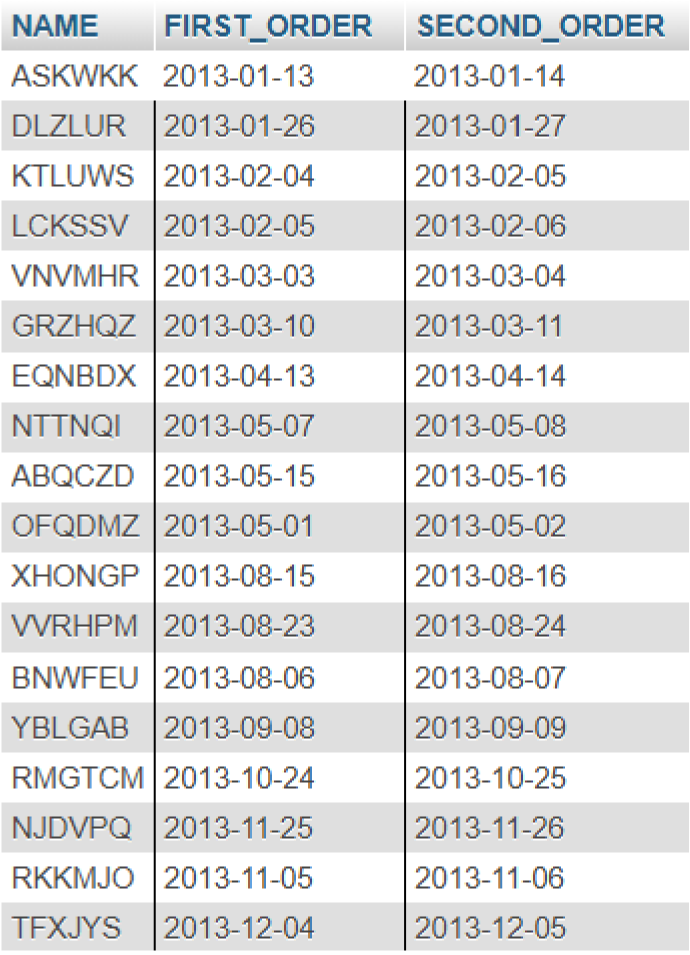
CUSTOMER_NAME	ORDERDATE	REVIEW_DATE	DAYS_TO_REVIEW
JECMUB	2013-01-24	2013-09-05	224
LQZRHJ	2013-01-26	2013-04-14	78
HNCJCB	2013-01-08	2013-09-24	259
EWAMKD	2013-01-05	2013-01-21	16
KXLNOG	2013-01-06	2013-08-27	233
CSCXAU	2013-01-05	2013-10-15	283
CSCXAU	2013-01-05	2013-03-01	55
SIWSNL	2013-01-21	2013-10-26	278
QHKDTL	2013-01-27	2013-06-11	135
NYIUCG	2013-01-21	2013-09-05	227
CIHVBC	2013-02-28	2013-10-27	241
WDXETB	2013-03-12	2013-08-14	155
SIQKEN	2013-03-05	2013-12-15	285
XNTJQU	2013-04-23	2013-08-11	110
WGTJJH	2013-04-22	2013-11-09	201
YJUGSQ	2013-04-25	2013-09-02	130
BGRLQX	2013-05-30	2013-12-31	215
YZPIHO	2013-05-08	2013-10-12	157
URXKAN	2013-05-08	2013-05-31	23
EBZSTF	2013-05-14	2013-10-11	150
AWKXBZ	2013-06-09	2013-09-18	101
GEUFMM	2013-06-05	2013-11-13	161
UDWPLC	2013-06-06	2013-10-18	134
POKJST	2013-07-20	2013-08-23	34

```
SELECT c.FIRSTNAME AS CUSTOMER_NAME, o.ORDERDATE, r.REVIEW_DATE,
DATEDIFF(r.REVIEW_DATE, o.ORDERDATE) AS DAYS_TO_REVIEW

FROM CUSTOMERS c

JOIN ORDERS o ON c.CUSTOMERID = o.CUSTOMERID

JOIN ORDERLINES ol ON o.ORDERID = ol.ORDERID

JOIN REVIEWS r ON c.CUSTOMERID = r.CUSTOMERID AND ol.PROD_ID = r.PROD_ID

WHERE r.REVIEW_DATE >= o.ORDERDATE;
```

Analyze Customer Order Frequency

Find customers who have placed orders on consecutive days. Show customer name and order dates.

Expected outcome:



NAME	FIRST_ORDER	SECOND_ORDER
ASKWKK	2013-01-13	2013-01-14
DLZLUR	2013-01-26	2013-01-27
KTLUWS	2013-02-04	2013-02-05
LCKSSV	2013-02-05	2013-02-06
VNVMHR	2013-03-03	2013-03-04
GRZHQZ	2013-03-10	2013-03-11
EQNBDX	2013-04-13	2013-04-14
NTTNQI	2013-05-07	2013-05-08
ABQCZD	2013-05-15	2013-05-16
OFQDMZ	2013-05-01	2013-05-02
XHONGP	2013-08-15	2013-08-16
VVRHPM	2013-08-23	2013-08-24
BNWFEU	2013-08-06	2013-08-07
YBLGAB	2013-09-08	2013-09-09
RMGTCM	2013-10-24	2013-10-25
NJDVPQ	2013-11-25	2013-11-26
RKKMJO	2013-11-05	2013-11-06
TFXJYS	2013-12-04	2013-12-05

```
SELECT c.FIRSTNAME AS NAME, o1.ORDERDATE AS FIRST_ORDER, o2.ORDERDATE AS SECOND_ORDER

FROM CUSTOMERS c

JOIN ORDERS o1 ON c.CUSTOMERID = o1.CUSTOMERID

JOIN ORDERS o2 ON c.CUSTOMERID = o2.CUSTOMERID

WHERE o2.ORDERDATE = o1.ORDERDATE + 1;
```