CS-591 Tools and techniques for Data Science

Assignment # 03 : GIT & R

Name: Muhammad kawish sarfraz **Roll # MSDS25038**

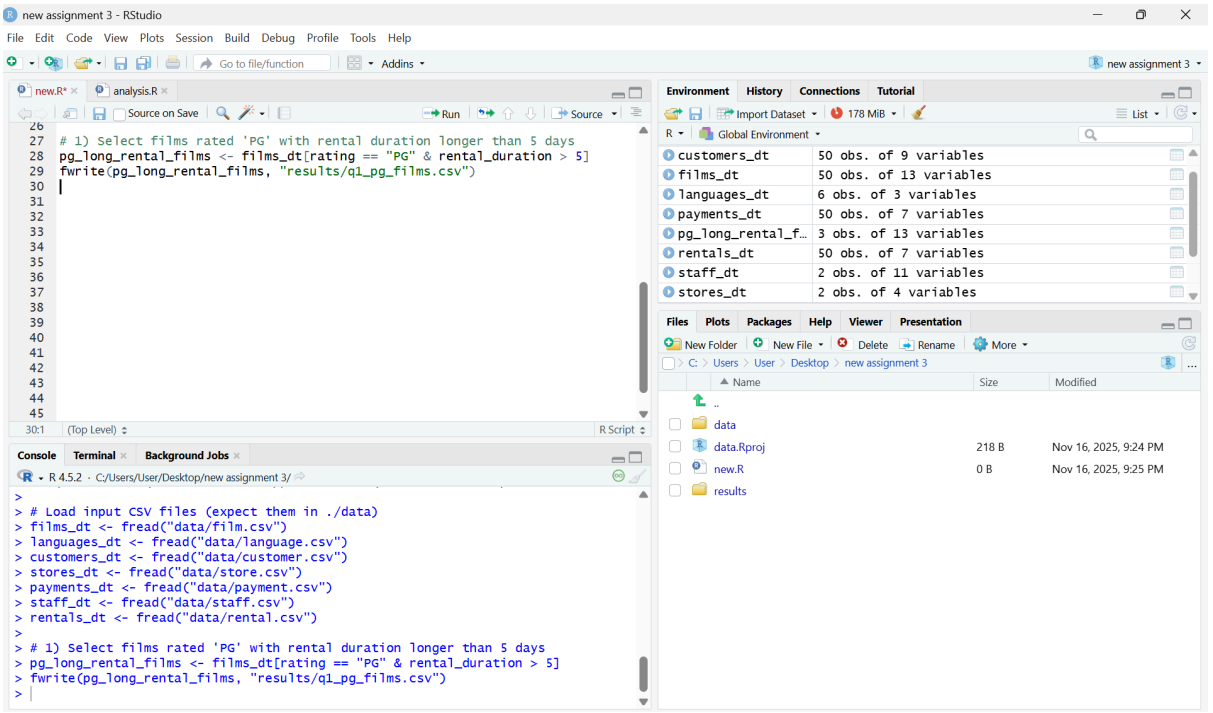
Use tables of 'Sakila' schema which were used during SQL lectures to complete following tasks. Use the Git and data.table package of R instead.

1. Write a query to display all films that have a rating of PG and a rental duration greater than 5 days. [10 Marks]

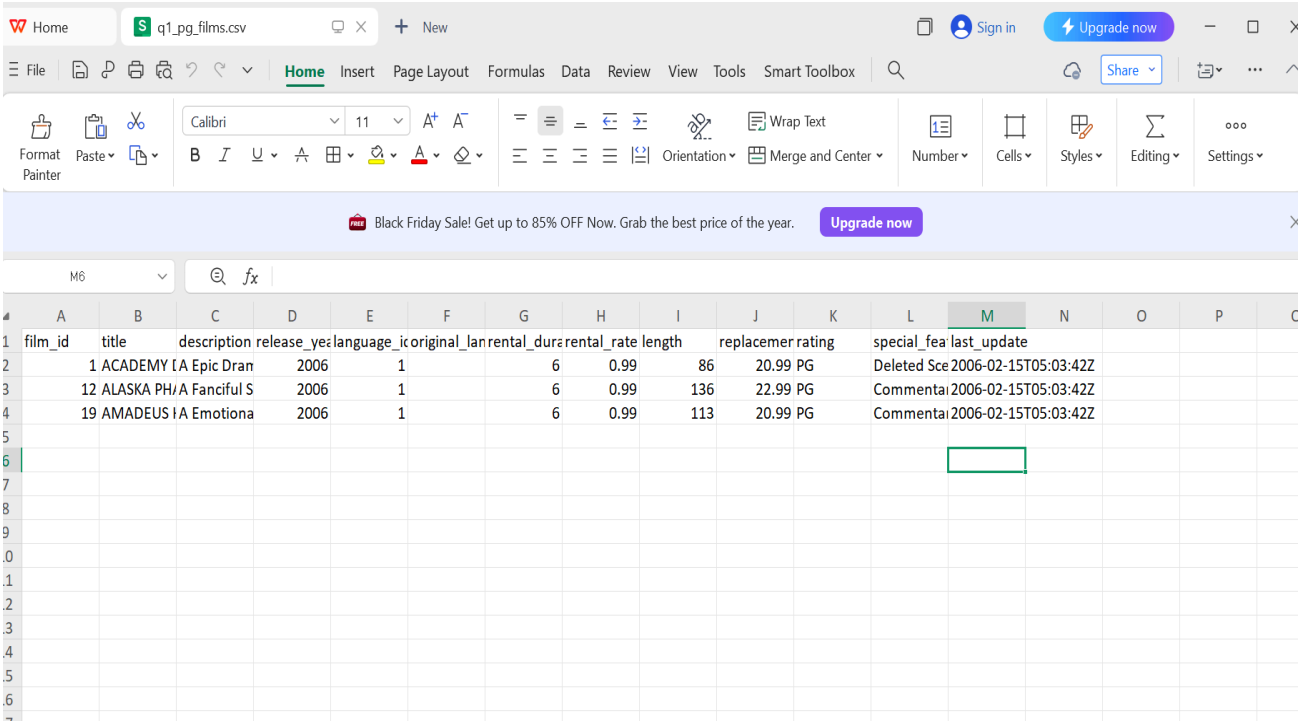
Code:

```
pg_long_rental_films <- films_dt[rating == "PG" &
rental_duration > 5]
fwrite(pg_long_rental_films, "results/q1_pg_films.csv")
```

Screenshot of running code:



Result (opened from result folder):



Explanation:

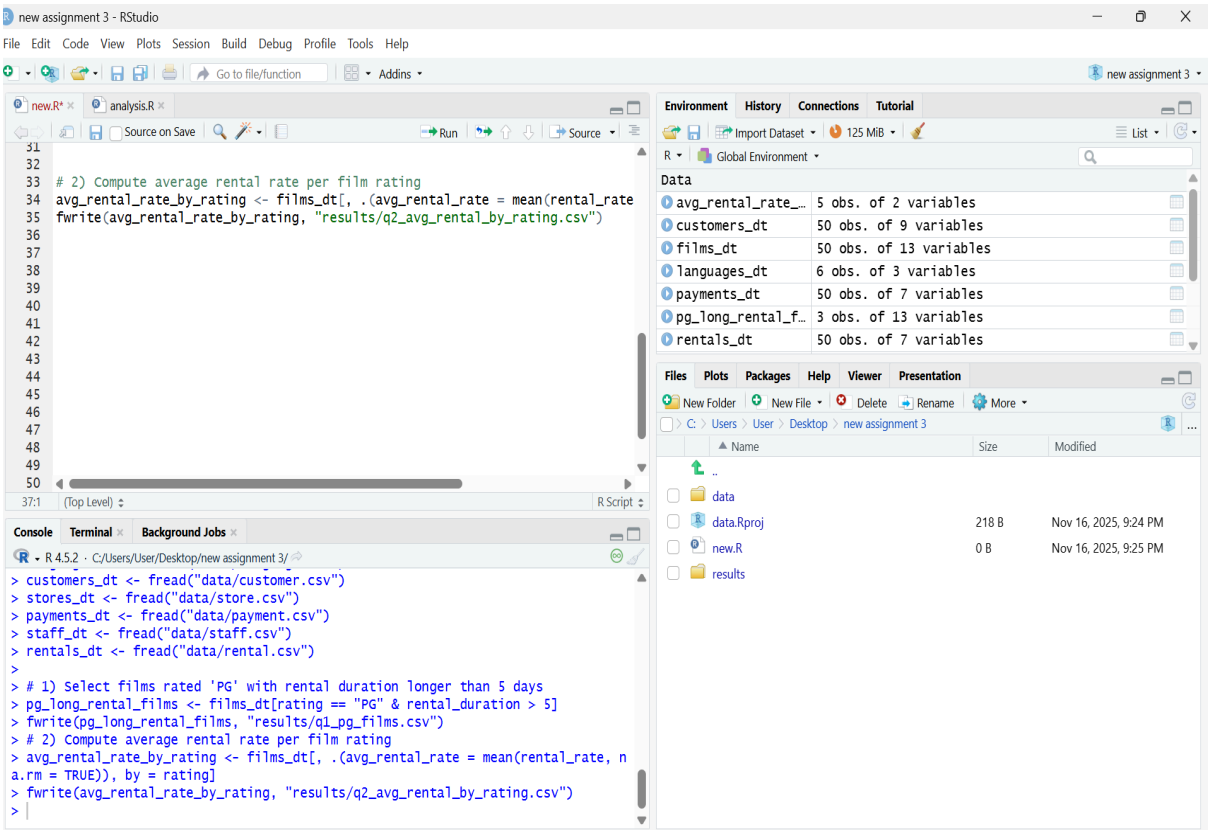
This query filters the film table to show only those movies that are rated PG and rented for more than 5 days. It basically applies two conditions together using AND. The result helps us identify longer-duration PG movies.

2. Write a query to display the average rental rate of films, grouped by their rating. [10 Marks]

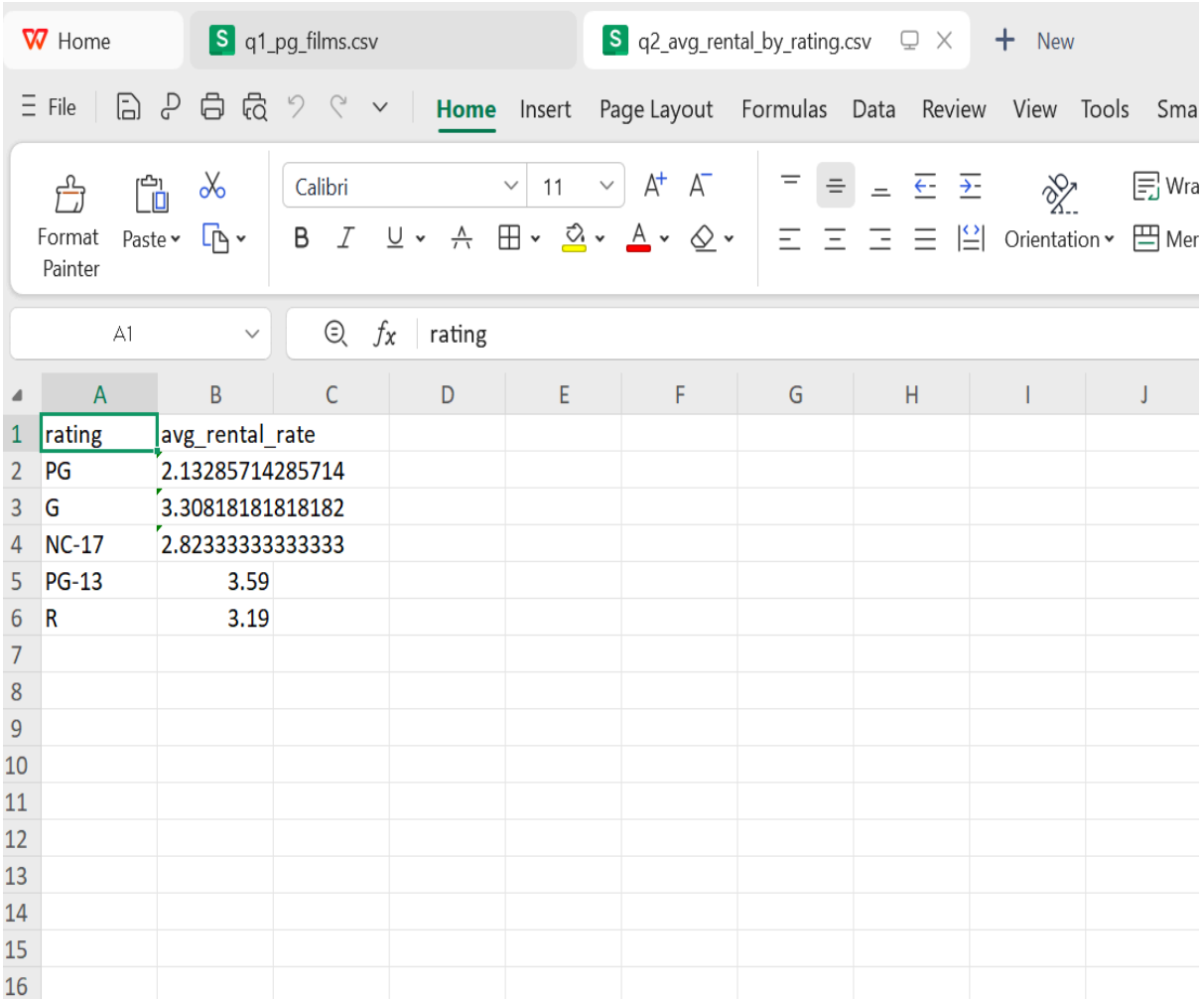
Code:

```
avg_rental_rate_by_rating <- films_dt[, .(avg_rental_rate
= mean(rental_rate, na.rm = TRUE)), by = rating]
fwrite(avg_rental_rate_by_rating,
"results/q2_avg_rental_by_rating.csv")
```

Screenshot of running code:



Result (opened from result folder):



	A	B	C	D	E	F	G	H	I	J
1	rating	avg_rental_rate								
2	PG	2.13285714285714								
3	G	3.30818181818182								
4	NC-17	2.82333333333333								
5	PG-13	3.59								
6	R	3.19								
7										
8										
9										
10										
11										
12										
13										
14										
15										
16										

Explanation:

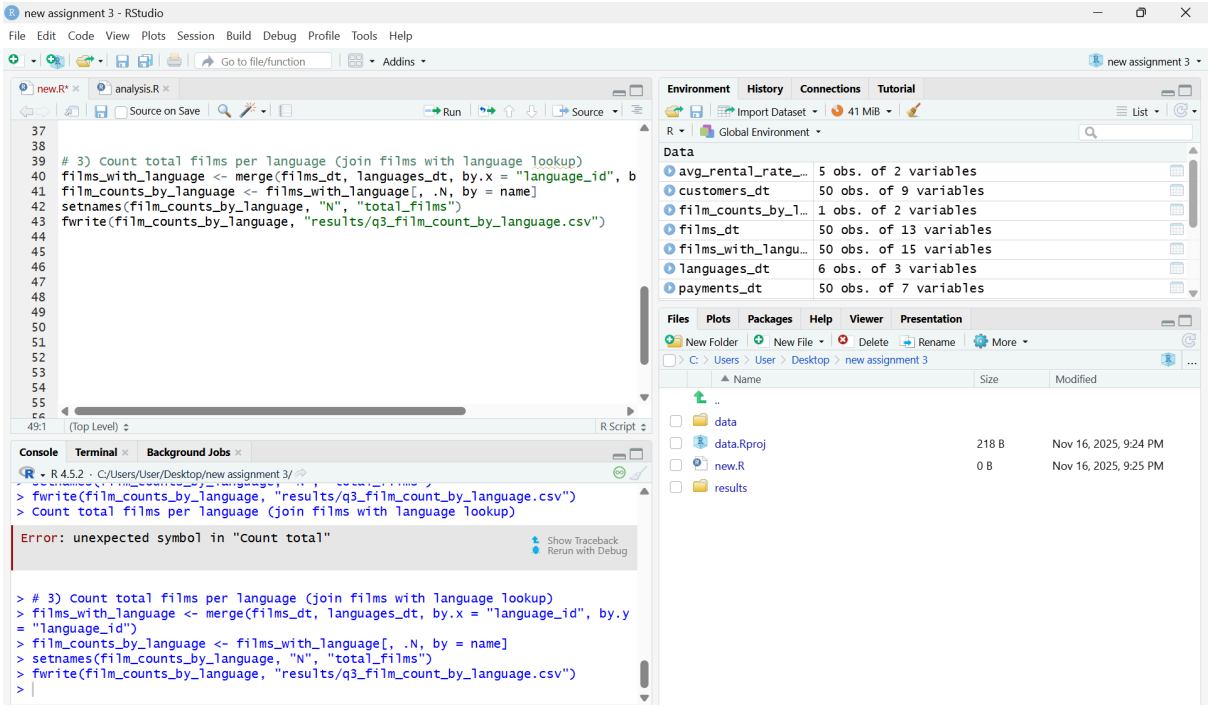
Here I grouped all films by their rating and calculated the average rental rate for each group. This helps us compare which ratings have higher or lower rental prices. It's similar to doing a summary for each category.

3. Write a query to count the total number of films in each language. [10 Marks]

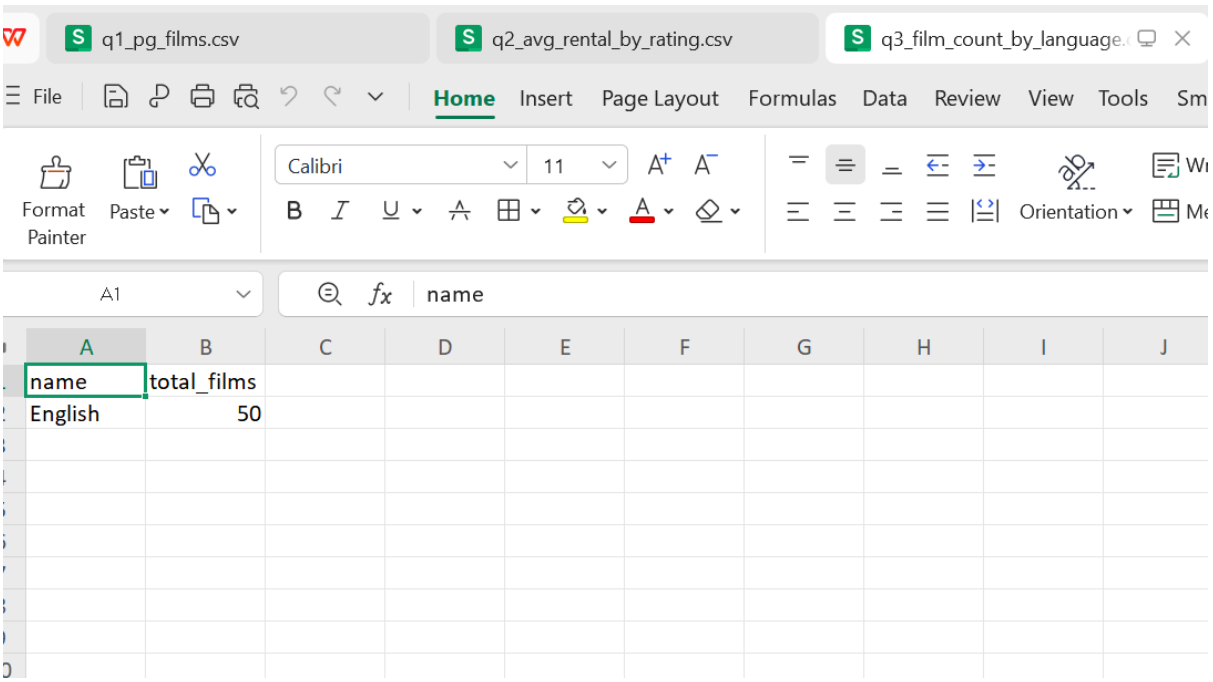
Code:

```
films_with_language <- merge(films_dt, languages_dt, by.x
= "language_id", by.y = "language_id")
film_counts_by_language <- films_with_language[, .N, by =
name]
setnames(film_counts_by_language, "N", "total_films")
fwrite(film_counts_by_language,
"results/q3_film_count_by_language.csv")
```

Screenshot of running code:



Result (opened from result folder):



Explanation:

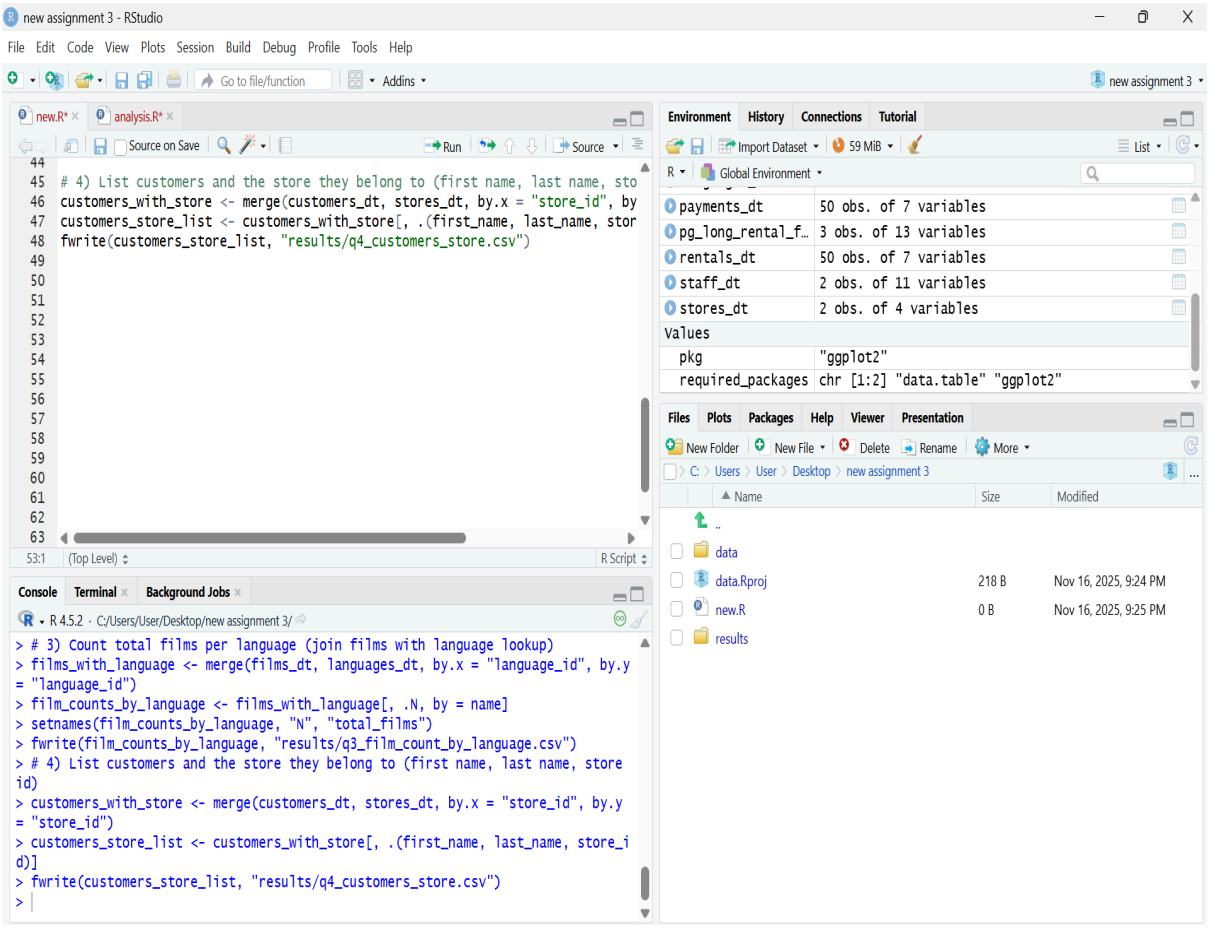
This query joins the films with the language table and then counts how many films exist for each language. It basically tells us which languages have more movies available. Grouping makes it easy to summarize by language.

4. List the customers' names and the store they belong to. [10 Marks]

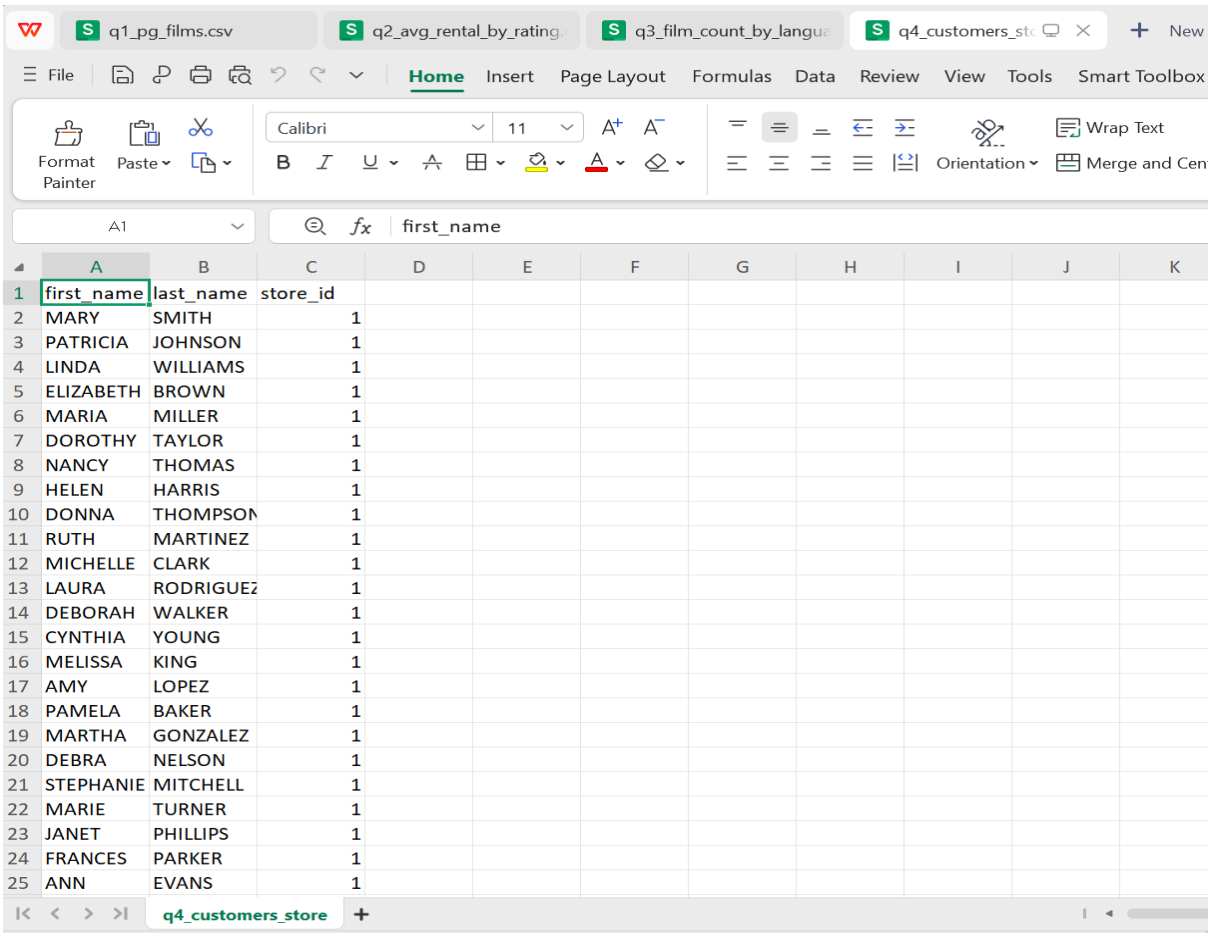
Code:

```
customers_with_store <- merge(customers_dt, stores_dt,
by.x = "store_id", by.y = "store_id")
customers_store_list <- customers_with_store[,
.(first_name, last_name, store_id)]
fwrite(customers_store_list,
"results/q4_customers_store.csv")
```

Screenshot of running code:



Result (opened from result folder):



first_name	last_name	store_id
MARY	SMITH	1
PATRICIA	JOHNSON	1
LINDA	WILLIAMS	1
ELIZABETH	BROWN	1
MARIA	MILLER	1
DOROTHY	TAYLOR	1
NANCY	THOMAS	1
HELEN	HARRIS	1
DONNA	THOMPSON	1
RUTH	MARTINEZ	1
MICHELLE	CLARK	1
LAURA	RODRIGUEZ	1
DEBORAH	WALKER	1
CYNTHIA	YOUNG	1
MELISSA	KING	1
AMY	LOPEZ	1
PAMELA	BAKER	1
MARTHA	GONZALEZ	1
DEBRA	NELSON	1
STEPHANIE	MITCHELL	1
MARIE	TURNER	1
JANET	PHILLIPS	1
FRANCES	PARKER	1
ANN	EVANS	1

Explanation:

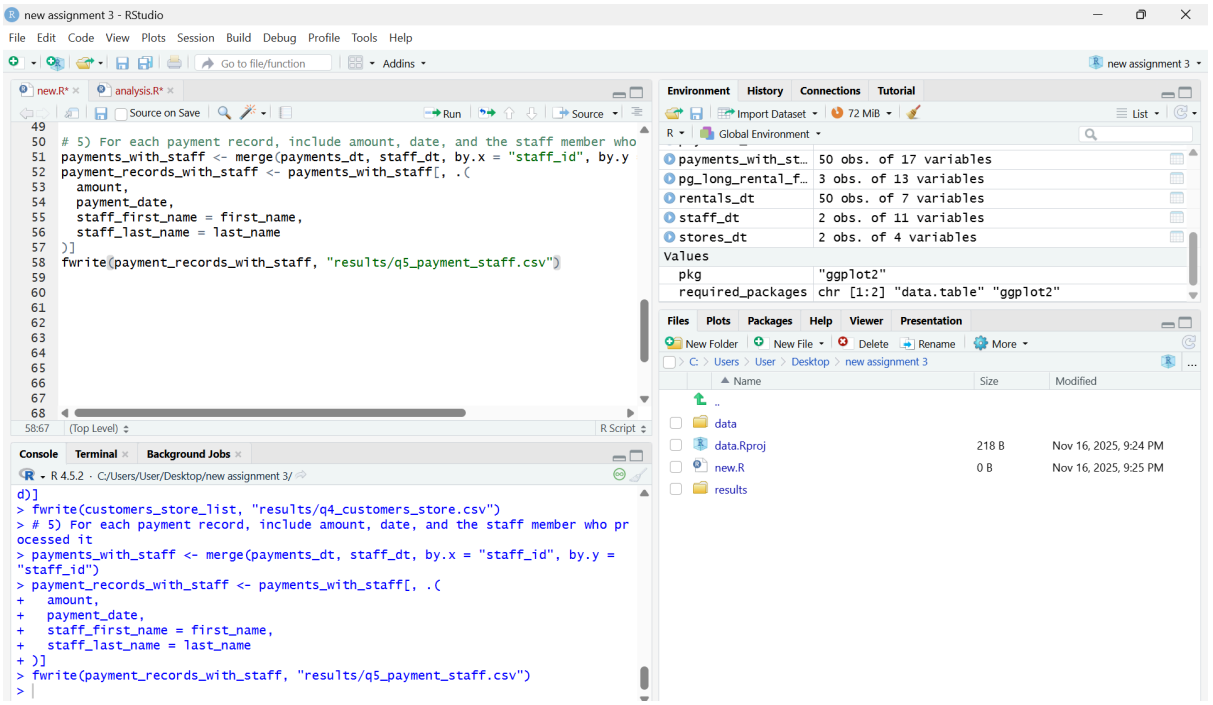
This merges customer information with store details using their store_id. The result shows each customer along with the store they are registered in. It helps understand how customers are distributed across stores.

5. Display the payment amount, payment date, and the staff member who processed each payment. [10 Marks]

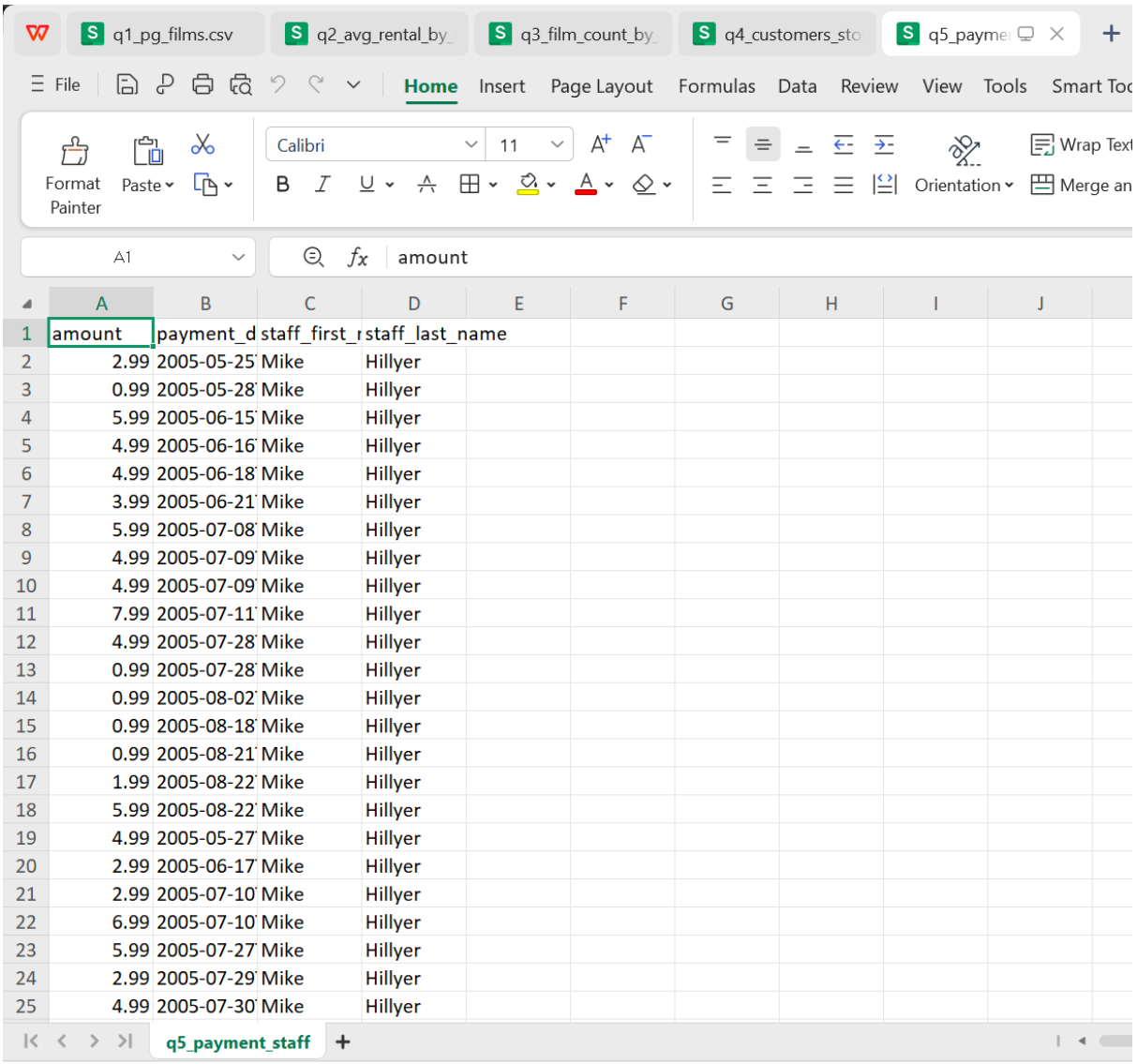
Code:

```
payments_with_staff <- merge(payments_dt, staff_dt, by.x =
"staff_id", by.y = "staff_id")
payment_records_with_staff <- payments_with_staff[, .(
  amount,
  payment_date,
  staff_first_name = first_name,
  staff_last_name = last_name
)]
fwrite(payment_records_with_staff,
"results/q5_payment_staff.csv")
```

Screenshot of running code:



Result (opened from result folder):



amount	payment_d	staff_first_r	staff_last_name
2.99	2005-05-25	Mike	Hillyer
0.99	2005-05-28	Mike	Hillyer
5.99	2005-06-15	Mike	Hillyer
4.99	2005-06-16	Mike	Hillyer
4.99	2005-06-18	Mike	Hillyer
3.99	2005-06-21	Mike	Hillyer
5.99	2005-07-08	Mike	Hillyer
4.99	2005-07-09	Mike	Hillyer
4.99	2005-07-09	Mike	Hillyer
7.99	2005-07-11	Mike	Hillyer
4.99	2005-07-28	Mike	Hillyer
0.99	2005-07-28	Mike	Hillyer
0.99	2005-08-02	Mike	Hillyer
0.99	2005-08-18	Mike	Hillyer
0.99	2005-08-21	Mike	Hillyer
1.99	2005-08-22	Mike	Hillyer
5.99	2005-08-22	Mike	Hillyer
4.99	2005-05-27	Mike	Hillyer
2.99	2005-06-17	Mike	Hillyer
2.99	2005-07-10	Mike	Hillyer
6.99	2005-07-10	Mike	Hillyer
5.99	2005-07-27	Mike	Hillyer
2.99	2005-07-29	Mike	Hillyer
4.99	2005-07-30	Mike	Hillyer

Explanation:

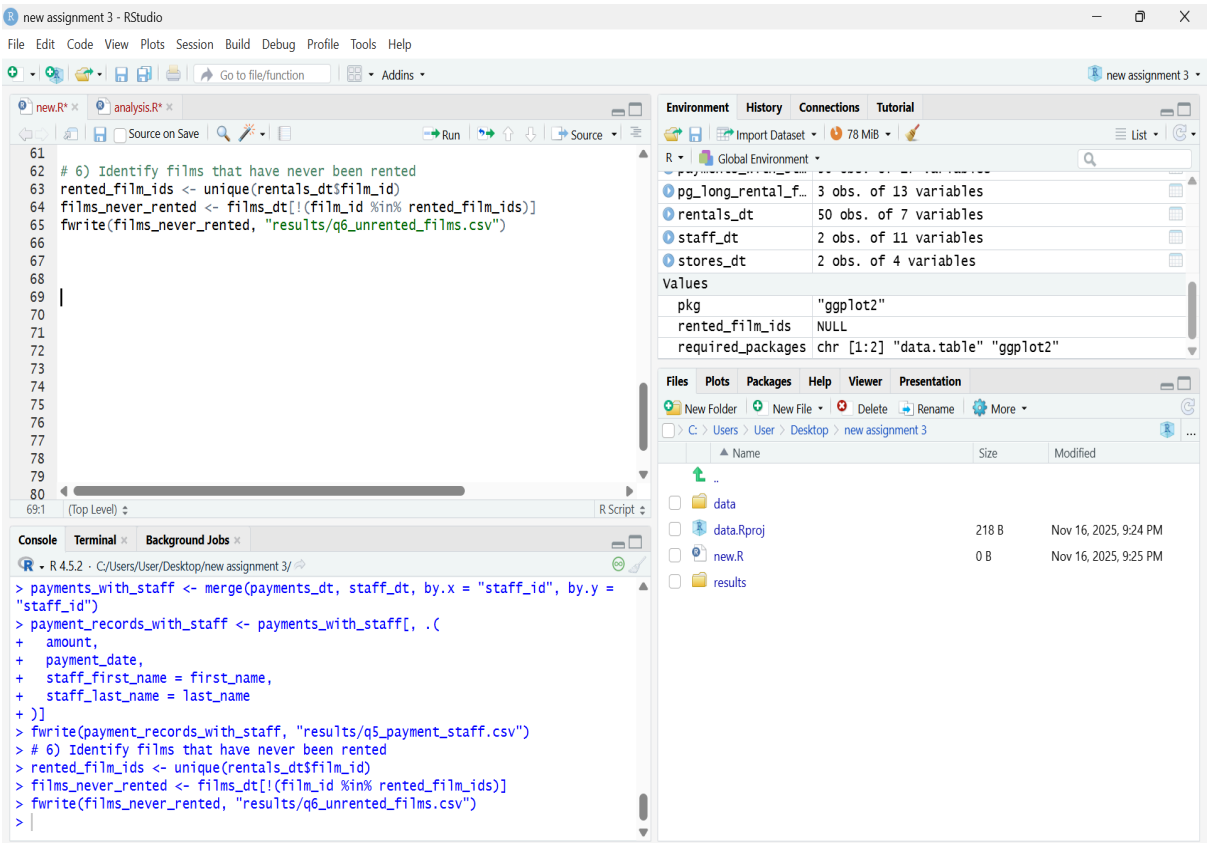
This joins the payment table with the staff table to show who handled each payment. The output includes the amount, the payment date, and the staff member's name. It helps track staff activity and payment processing.

6. Find the films that are not rented. [10 Marks]

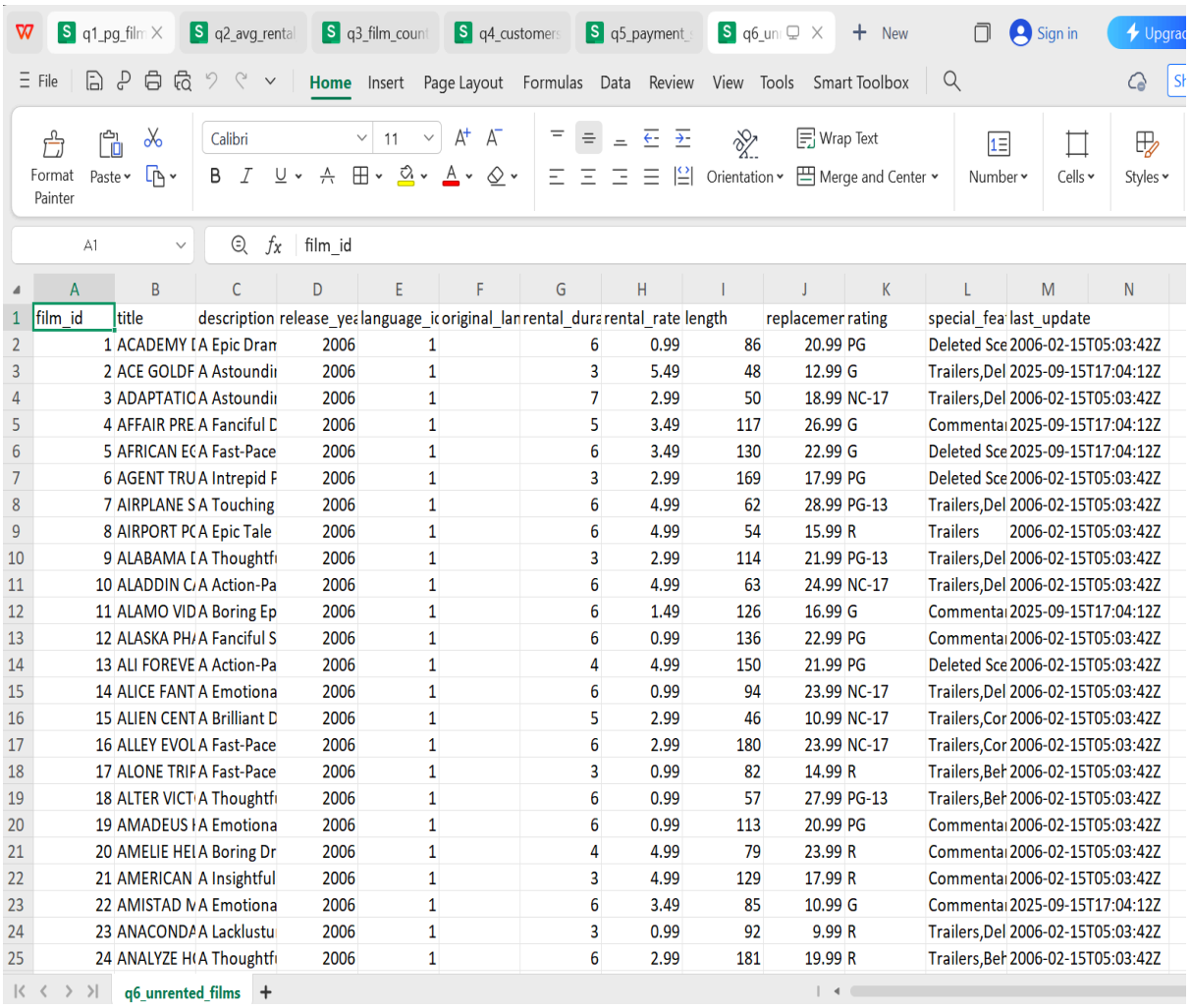
Code:

```
rented_film_ids <- unique(rentals_dt$film_id)
films_never_rented <- films_dt[!(film_id %in%
rented_film_ids)]
fwrite(films_never_rented,
"results/q6_unrented_films.csv")
```

Screenshot of running code:



Result (opened from result folder):



film_id	title	description	release_yea	language_ic	original_lan	rental_dura	rental_rate	length	replacemer	rating	special_fea	last_update
1	ACADEMY I	A Epic Dran	2006	1		6	0.99	86	20.99	PG	Deleted Sce	2006-02-15T05:03:42Z
2	ACE GOLDF	A Astoundi	2006	1		3	5.49	48	12.99	G	Trailers,Del	2025-09-15T17:04:12Z
3	ADAPTATIC	A Astoundi	2006	1		7	2.99	50	18.99	NC-17	Trailers,Del	2006-02-15T05:03:42Z
4	AFFAIR PRE	A Fanciful D	2006	1		5	3.49	117	26.99	G	Commenta	2025-09-15T17:04:12Z
5	AFRICAN EC	A Fast-Pace	2006	1		6	3.49	130	22.99	G	Deleted Sce	2025-09-15T17:04:12Z
6	AGENT TRU	A Intrepid F	2006	1		3	2.99	169	17.99	PG	Deleted Sce	2006-02-15T05:03:42Z
7	AIRPLANE S	A Touching	2006	1		6	4.99	62	28.99	PG-13	Trailers,Del	2006-02-15T05:03:42Z
8	AIRPORT PC	A Epic Tale	2006	1		6	4.99	54	15.99	R	Trailers	2006-02-15T05:03:42Z
9	ALABAMA I	A Thoughtfi	2006	1		3	2.99	114	21.99	PG-13	Trailers,Del	2006-02-15T05:03:42Z
10	ALADDIN C	A Action-Pa	2006	1		6	4.99	63	24.99	NC-17	Trailers,Del	2006-02-15T05:03:42Z
11	ALAMO VID	A Boring Ep	2006	1		6	1.49	126	16.99	G	Commenta	2025-09-15T17:04:12Z
12	ALASKA PH	A Fanciful S	2006	1		6	0.99	136	22.99	PG	Commenta	2006-02-15T05:03:42Z
13	ALI FOREVE	A Action-Pa	2006	1		4	4.99	150	21.99	PG	Deleted Sce	2006-02-15T05:03:42Z
14	ALICE FANT	A Emotiona	2006	1		6	0.99	94	23.99	NC-17	Trailers,Del	2006-02-15T05:03:42Z
15	ALIEN CENT	A Brilliant D	2006	1		5	2.99	46	10.99	NC-17	Trailers,Cor	2006-02-15T05:03:42Z
16	ALLEY EVOL	A Fast-Pace	2006	1		6	2.99	180	23.99	NC-17	Trailers,Cor	2006-02-15T05:03:42Z
17	ALONE TRIF	A Fast-Pace	2006	1		3	0.99	82	14.99	R	Trailers,Bel	2006-02-15T05:03:42Z
18	ALTER VICT	A Thoughtfi	2006	1		6	0.99	57	27.99	PG-13	Trailers,Bel	2006-02-15T05:03:42Z
19	AMADEUS H	A Emotiona	2006	1		6	0.99	113	20.99	PG	Commenta	2006-02-15T05:03:42Z
20	AMELIE HEI	A Boring Dr	2006	1		4	4.99	79	23.99	R	Commenta	2006-02-15T05:03:42Z
21	AMERICAN	A Insightful	2006	1		3	4.99	129	17.99	R	Commenta	2006-02-15T05:03:42Z
22	AMISTAD M	A Emotiona	2006	1		6	3.49	85	10.99	G	Commenta	2025-09-15T17:04:12Z
23	ANACONDA	A Lacklustu	2006	1		3	0.99	92	9.99	R	Trailers,Del	2006-02-15T05:03:42Z
24	ANALYZE H	A Thoughtfi	2006	1		6	2.99	181	19.99	R	Trailers,Bel	2006-02-15T05:03:42Z

Explanation:

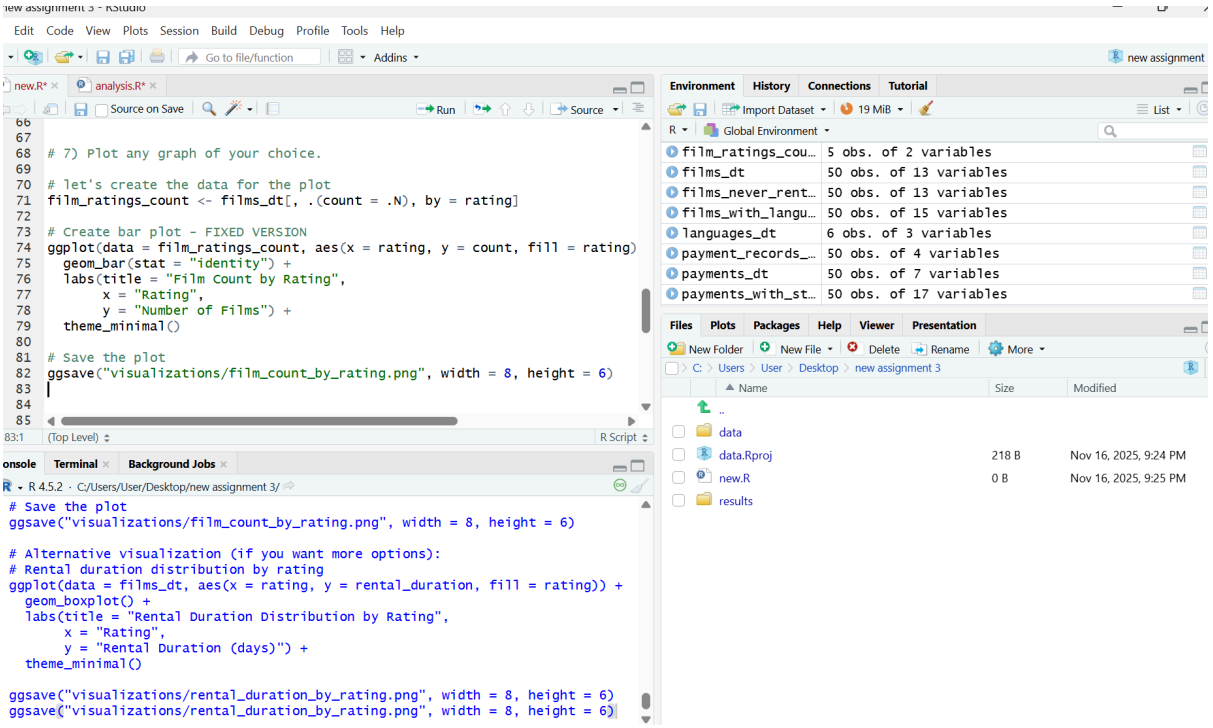
This query finds all films whose film_id does not appear in the rentals table. It basically checks which movies have zero rental records. This helps identify films that are not being watched or rented.

7. Plot any graph of your choice. [10 Marks]

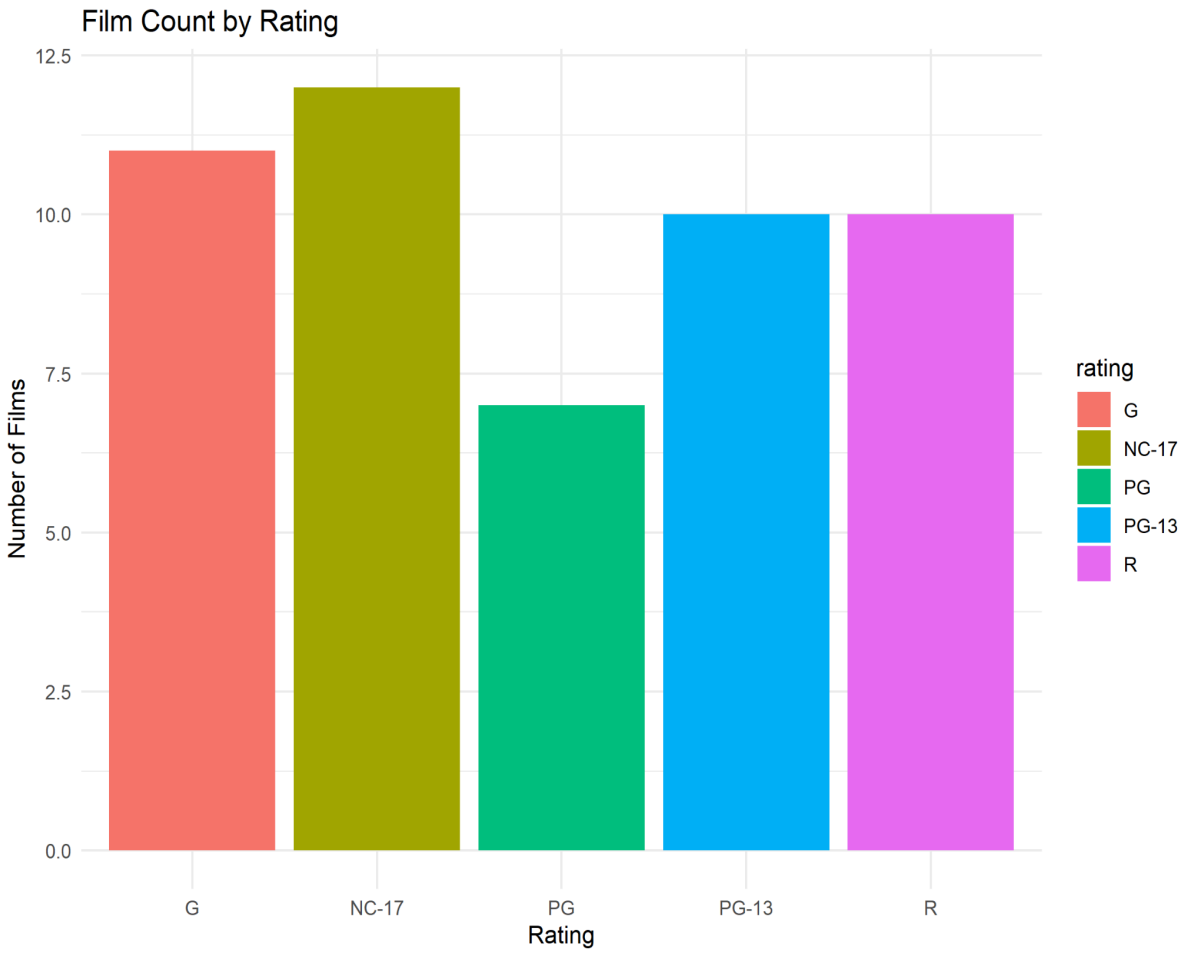
Code:

```
# let's create the data for the plot
film_ratings_count <- films_dt[, .(count = .N), by =
rating]

# Create bar plot - FIXED VERSION
ggplot(data = film_ratings_count, aes(x = rating, y =
count, fill = rating)) +
  geom_bar(stat = "identity") +
  labs(title = "Film Count by Rating",
       x = "Rating",
       y = "Number of Films") +
  theme_minimal()

# Save the plot
ggsave("visualizations/film_count_by_rating.png", width =
8, height = 6)
```

Screenshot of running code:



Visual/graph (from visualisation folder):

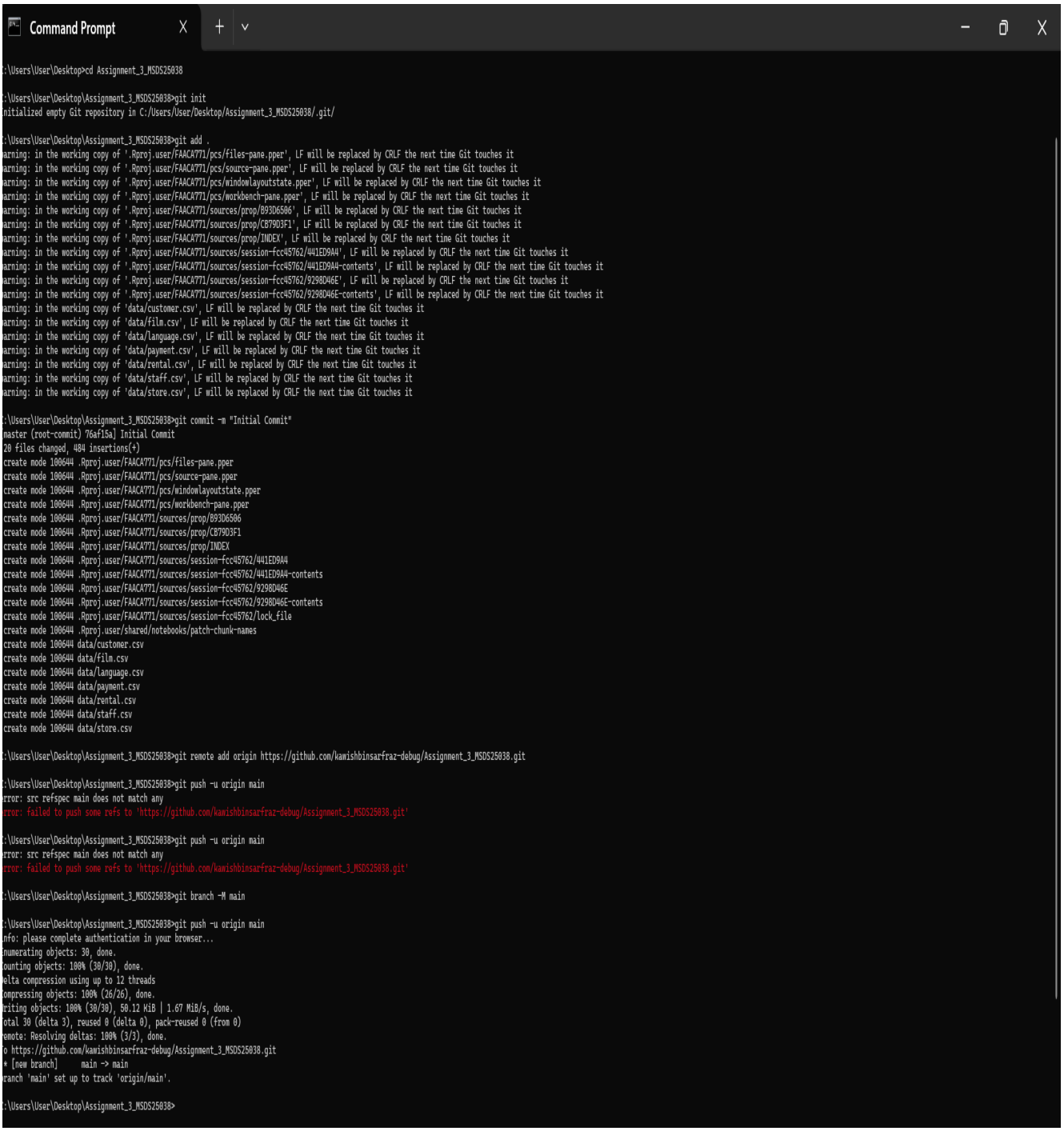


Explanation:

For the plot, I drew a bar graph, counted how many films fall under each rating and visualized it using a bar chart. This helps quickly see which ratings have the highest number of films.

8. Use of Git in the entire assignment. [30 Marks]

Initial commit:



Committing R script and its results:

```
C:\Users\User\Desktop\Assignment_3_MSDS25038>git status
On branch main
Your branch is up to date with 'origin/main'.

Untracked files:
  (use "git add <file>..." to include in what will be committed)
        data.Rproj
        new.R
        results/
        visualizations/

nothing added to commit but untracked files present (use "git add" to track)

C:\Users\User\Desktop\Assignment_3_MSDS25038>git add .

C:\Users\User\Desktop\Assignment_3_MSDS25038>git commit -m "Adding R script file and output result files and visualizations"
[main 97bff7b] Adding R script file and output result files and visualizations
 10 files changed, 178 insertions(+)
 create mode 100644 data.Rproj
 create mode 100644 new.R
 create mode 100644 results/q1_pg_films.csv
 create mode 100644 results/q2_avg_rental_by_rating.csv
 create mode 100644 results/q3_film_count_by_language.csv
 create mode 100644 results/q4_customers_store.csv
 create mode 100644 results/q5_payment_staff.csv
 create mode 100644 results/q6_unrented_films.csv
 create mode 100644 visualizations/film_count_by_rating.png
 create mode 100644 visualizations/rental_duration_by_rating.png

C:\Users\User\Desktop\Assignment_3_MSDS25038>git push
Enumerating objects: 15, done.
Counting objects: 100% (15/15), done.
Delta compression using up to 12 threads
Compressing objects: 100% (12/12), done.
Writing objects: 100% (14/14), 46.18 KiB | 2.72 MiB/s, done.
Total 14 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/kawishbinsarfraz-debug/Assignment_3_MSDS25038.git
   76af15a..97bff7b  main -> main

C:\Users\User\Desktop\Assignment_3_MSDS25038>
```

Final commit:

In the final commit I added this report in my local repo and then committed these changes into the Github repository.

Github repo:

https://github.com/kawishbinsarfraz-debug/Assignment_3_MSDS25038