DEPA Assignment 3

Troy Zhongyi Zhang
Netid: zhongyiz@uchicago.edu

SakilaSnowflakeDW-DDL:

```
-- -----------------------------------------------------
-- Table `sakila_snowflake`.`fact_rental`
-- Write Fact table fact_rental DDL script here
-- -----------------------------------------------------

CREATE TABLE IF NOT EXISTS `sakila_snowflake`.`fact_rental` (
  `rental_id` INT(10) NOT NULL,
  `rental_last_update` TIMESTAMP NOT NULL DEFAULT CURRENT_TIMESTAMP ON
UPDATE CURRENT_TIMESTAMP,
  `customer_key` INT(8) NOT NULL,
  `staff_key` INT(8) NOT NULL,
  `film_key` INT(8) NOT NULL,
  `store_key` INT(8) NOT NULL,
  `rental_date_key` BIGINT(20) NOT NULL,
  `return_date_key` BIGINT(20) DEFAULT NULL,
  `count_returns` INT(10) NOT NULL,
  `count_rentals` INT(8) NOT NULL,
  `rental_duration` INT(10) NULL DEFAULT NULL,
  `dollar_amount` FLOAT,
  PRIMARY KEY (`rental_id`),
  CONSTRAINT `dim_customer_fact_rental_fk`
    FOREIGN KEY (`customer_key`)
    REFERENCES `sakila_snowflake`.`dim_customer` (`customer_key`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `dim_film_fact_rental_fk`
    FOREIGN KEY (`film_key`)
    REFERENCES `sakila_snowflake`.`dim_film` (`film_key`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `dim_staff_fact_rental_fk`
    FOREIGN KEY (`staff_key`)
    REFERENCES `sakila_snowflake`.`dim_staff` (`staff_key`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `dim_store_fact_rental_fk`
    FOREIGN KEY (`store_key`)
```

```
    REFERENCES `sakila_snowflake`.`dim_store` (`store_key`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `dim_date_fact_rental_fk`
    FOREIGN KEY (`rental_date_key`)
    REFERENCES `sakila_snowflake`.`dim_date` (`date_Id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE,
  CONSTRAINT `dim_date_fact_rental_fk1`
    FOREIGN KEY (`return_date_key`)
    REFERENCES `sakila_snowflake`.`dim_date` (`date_Id`)
    ON DELETE CASCADE
    ON UPDATE CASCADE)
ENGINE = InnoDB
DEFAULT CHARACTER SET = latin1;


CREATE INDEX `dim_store_fact_rental_fk` ON `sakila_snowflake`.`fact_rental` (`store_key`
ASC);
CREATE INDEX `dim_staff_fact_rental_fk` ON `sakila_snowflake`.`fact_rental` (`staff_key`
ASC);
CREATE INDEX `dim_film_fact_rental_fk` ON `sakila_snowflake`.`fact_rental` (`film_key`
ASC);
CREATE INDEX `dim_customer_fact_rental_fk` ON `sakila_snowflake`.`fact_rental`
(`customer_key` ASC);

SET SQL_MODE=@OLD_SQL_MODE;
SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS;
SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS;
```

Action Output

	Time	Action	Response	Duration / Fetch Time
85	01:41:39	CREATE INDEX `dim_actor_dim_film_actor_bridge_fk` ON `sakila_snowflake`.`dim_film_actor_bridge` (`actor_key` ASC)	0 row(s) affected Records: 0 Duplicates: 0 Warnings...	0.0046 sec
86	01:41:39	CREATE TABLE IF NOT EXISTS `sakila_snowflake`.`dim_staff` (`staff_key` INT(8) NOT NULL AUTO_INCREMENT, `st...	0 row(s) affected	0.0040 sec
87	01:41:39	CREATE INDEX `dim_staff_last_update` ON `sakila_snowflake`.`dim_staff` (`staff_last_update` ASC)	0 row(s) affected Records: 0 Duplicates: 0 Warnings...	0.0040 sec
88	01:41:39	CREATE TABLE IF NOT EXISTS `sakila_snowflake`.`dim_store` (`store_key` INT(8) NOT NULL AUTO_INCREMENT, `l...	0 row(s) affected	0.0052 sec
89	01:41:39	CREATE INDEX `store_id` USING BTREE ON `sakila_snowflake`.`dim_store` (`store_id` ASC)	0 row(s) affected Records: 0 Duplicates: 0 Warnings...	0.0057 sec
90	01:41:39	CREATE INDEX `dim_store_last_update` ON `sakila_snowflake`.`dim_store` (`store_last_update` ASC)	0 row(s) affected Records: 0 Duplicates: 0 Warnings...	0.0051 sec
91	01:41:39	CREATE INDEX `dim_location_address_dim_store_fk` ON `sakila_snowflake`.`dim_store` (`location_address_key` ASC)	0 row(s) affected Records: 0 Duplicates: 0 Warnings...	0.0048 sec
92	01:41:39	CREATE TABLE IF NOT EXISTS `sakila_snowflake`.`dim_date` (`date_Id` BIGINT(20) NOT NULL, `date` DATE NOT N...	0 row(s) affected	0.0056 sec
93	01:41:39	CREATE TABLE IF NOT EXISTS `sakila_snowflake`.`numbers` (`number` BIGINT(20) NULL DEFAULT NULL) ENGINE =...	0 row(s) affected	0.0030 sec
94	01:41:39	CREATE TABLE IF NOT EXISTS `sakila_snowflake`.`numbers_small` (`number` INT(11) NULL DEFAULT NULL) ENGINE...	0 row(s) affected	0.0035 sec
95	01:41:39	CREATE TABLE IF NOT EXISTS `sakila_snowflake`.`fact_rental` (`rental_id` INT(10) NOT NULL, `rental_last_update`...	0 row(s) affected	0.015 sec
96	01:41:39	CREATE INDEX `dim_store_fact_rental_fk` ON `sakila_snowflake`.`fact_rental` (`store_key` ASC)	0 row(s) affected Records: 0 Duplicates: 0 Warnings...	0.0083 sec
97	01:41:39	CREATE INDEX `dim_staff_fact_rental_fk` ON `sakila_snowflake`.`fact_rental` (`staff_key` ASC)	0 row(s) affected Records: 0 Duplicates: 0 Warnings...	0.0063 sec
98	01:41:39	CREATE INDEX `dim_film_fact_rental_fk` ON `sakila_snowflake`.`fact_rental` (`film_key` ASC)	0 row(s) affected Records: 0 Duplicates: 0 Warnings...	0.0052 sec
99	01:41:39	CREATE INDEX `dim_customer_fact_rental_fk` ON `sakila_snowflake`.`fact_rental` (`customer_key` ASC)	0 row(s) affected Records: 0 Duplicates: 0 Warnings...	0.0060 sec
100	01:41:39	SET SQL_MODE=@OLD_SQL_MODE	0 row(s) affected	0.000083 sec
101	01:41:39	SET FOREIGN_KEY_CHECKS=@OLD_FOREIGN_KEY_CHECKS	0 row(s) affected	0.000062 sec
102	01:41:39	SET UNIQUE_CHECKS=@OLD_UNIQUE_CHECKS	0 row(s) affected	0.000063 sec

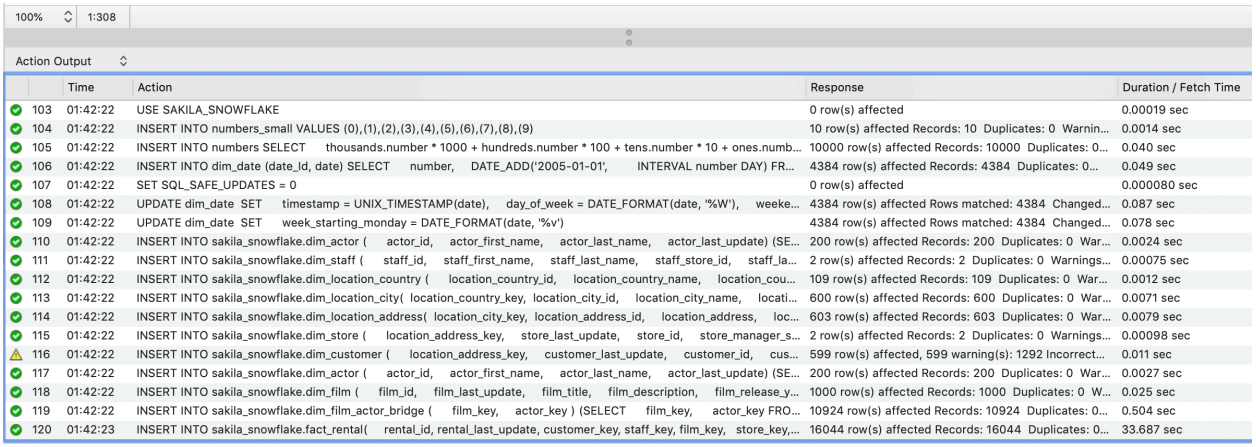
SakilaSnowflakeDW-DML:

```
# The below query might take over 30 seconds to complete and you might get an "Error Code: 2013.
# Lost connection to MySQL server during query" error
# Please follow the instructions below:
#   - In the application menu, select Edit > Preferences > SQL Editor.
#   - Look for the MySQL Session section and increase the DBMS connection read time out value.
#   - Save the settings, quit MySQL Workbench and reopen the connection.

-- -----------------------------------------------------
-- Write Fact table fact_rental DML script here
-- -----------------------------------------------------

INSERT INTO sakila_snowflake.fact_rental(
   rental_id, rental_last_update, customer_key, staff_key, film_key,
        store_key, rental_date_key, return_date_key, count_returns,
        count_rentals, rental_duration, dollar_amount
)
(SELECT
   rental_id,
   r.last_update,
   customer_key,
   staff_key,
   film_key,
   store_key,
   dt.date_id AS rental_date_key,
   dt2.date_id AS return_date_key,
   sum(case when dt.date is not null then 1 else 0 end) as rentalcount,
   sum(case when dt2.date is not null then 1 else 0 end) as returncount,
   sum(film_rental_duration) as rental_duration,
   (film.film_rental_rate * datediff(r.return_date, r.rental_date)) as dollar_amount
FROM sakila.rental r
  join dim_customer cu on cu.customer_id = r.customer_id
  join dim_staff staff on r.staff_id = staff.staff_id
  join sakila.inventory inv on r.inventory_id = inv.inventory_id
  join dim_film film on inv.film_id = film.film_id
  join dim_store store on inv.store_id = store.store_id
  left join dim_date dt on DATE(dt.date) = DATE(r.rental_date)
  left join dim_date dt2 on DATE(dt2.date) = DATE(r.return_date)
GROUP BY
        rental_id,
        r.last_update,
        customer_key,
        staff_key,
```

film_key,
store_key,
dt.date_id,
dt2.date_id);



100% 1:308

Action Output

	Time	Action	Response	Duration / Fetch Time
103	01:42:22	USE SAKILA_SNOWFLAKE	0 row(s) affected	0.00019 sec
104	01:42:22	INSERT INTO numbers_small VALUES (0),(1),(2),(3),(4),(5),(6),(7),(8),(9)	10 row(s) affected Records: 10 Duplicates: 0 Warnin...	0.0014 sec
105	01:42:22	INSERT INTO numbers SELECT thousands.number * 1000 + hundreds.number * 100 + tens.number * 10 + ones.numb...	10000 row(s) affected Records: 10000 Duplicates: 0...	0.040 sec
106	01:42:22	INSERT INTO dim_date (date_Id, date) SELECT number, DATE_ADD('2005-01-01', INTERVAL number DAY) FR...	4384 row(s) affected Records: 4384 Duplicates: 0...	0.049 sec
107	01:42:22	SET SQL_SAFE_UPDATES = 0	0 row(s) affected	0.000080 sec
108	01:42:22	UPDATE dim_date SET timestamp = UNIX_TIMESTAMP(date), day_of_week = DATE_FORMAT(date, '%W'), weeke...	4384 row(s) affected Rows matched: 4384 Changed...	0.087 sec
109	01:42:22	UPDATE dim_date SET week_starting_monday = DATE_FORMAT(date, '%v')	4384 row(s) affected Rows matched: 4384 Changed...	0.078 sec
110	01:42:22	INSERT INTO sakila_snowflake.dim_actor (actor_id, actor_first_name, actor_last_name, actor_last_update) (SE...	200 row(s) affected Records: 200 Duplicates: 0 War...	0.0024 sec
111	01:42:22	INSERT INTO sakila_snowflake.dim_staff (staff_id, staff_first_name, staff_last_name, staff_store_id, staff_la...	2 row(s) affected Records: 2 Duplicates: 0 Warnings...	0.00075 sec
112	01:42:22	INSERT INTO sakila_snowflake.dim_location_country (location_country_id, location_country_name, location_cou...	109 row(s) affected Records: 109 Duplicates: 0 War...	0.0012 sec
113	01:42:22	INSERT INTO sakila_snowflake.dim_location_city(location_country_key, location_city_id, location_city_name, locati...	600 row(s) affected Records: 600 Duplicates: 0 War...	0.0071 sec
114	01:42:22	INSERT INTO sakila_snowflake.dim_location_address(location_city_key, location_address_id, location_address, loc...	603 row(s) affected Records: 603 Duplicates: 0 War...	0.0079 sec
115	01:42:22	INSERT INTO sakila_snowflake.dim_store (location_address_key, store_last_update, store_id, store_manager_s...	2 row(s) affected Records: 2 Duplicates: 0 Warnings...	0.00098 sec
116	01:42:22	INSERT INTO sakila_snowflake.dim_customer (location_address_key, customer_last_update, customer_id, cus...	599 row(s) affected, 599 warning(s): 1292 Incorrect...	0.011 sec
117	01:42:22	INSERT INTO sakila_snowflake.dim_actor (actor_id, actor_first_name, actor_last_name, actor_last_update) (SE...	200 row(s) affected Records: 200 Duplicates: 0 War...	0.0027 sec
118	01:42:22	INSERT INTO sakila_snowflake.dim_film (film_id, film_last_update, film_title, film_description, film_release_y...	1000 row(s) affected Records: 1000 Duplicates: 0 W...	0.025 sec
119	01:42:22	INSERT INTO sakila_snowflake.dim_film_actor_bridge (film_key, actor_key) (SELECT film_key, actor_key FRO...	10924 row(s) affected Records: 10924 Duplicates: 0...	0.504 sec
120	01:42:23	INSERT INTO sakila_snowflake.fact_rental(rental_id, rental_last_update, customer_key, staff_key, film_key, store_key,...	16044 row(s) affected Records: 16044 Duplicates: 0...	33.687 sec

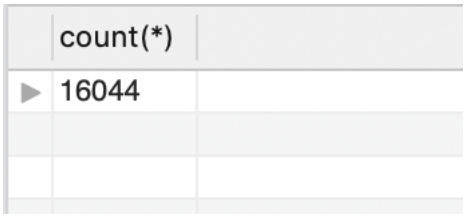
---

To Test my SQL queries result:

SELECT count(*) FROM sakila_snowflake.fact_rental;

1 row(s) returned

	Time	Action	Response	Duration / Fetch Time
116	01:42:22	INSERT INTO sakila_snowflake.dim_customer (location_address_key, customer_last_update, customer_id, cus...	599 row(s) affected, 599 warning(s): 1292 Incorrect...	0.011 sec
117	01:42:22	INSERT INTO sakila_snowflake.dim_actor (actor_id, actor_first_name, actor_last_name, actor_last_update) (SE...	200 row(s) affected Records: 200 Duplicates: 0 War...	0.0027 sec
118	01:42:22	INSERT INTO sakila_snowflake.dim_film (film_id, film_last_update, film_title, film_description, film_release_y...	1000 row(s) affected Records: 1000 Duplicates: 0 W...	0.025 sec
119	01:42:22	INSERT INTO sakila_snowflake.dim_film_actor_bridge (film_key, actor_key) (SELECT film_key, actor_key FRO...	10924 row(s) affected Records: 10924 Duplicates: 0...	0.504 sec
120	01:42:23	INSERT INTO sakila_snowflake.fact_rental(rental_id, rental_last_update, customer_key, staff_key, film_key, store_key,...	16044 row(s) affected Records: 16044 Duplicates: 0...	33.687 sec
121	01:44:24	SELECT * FROM sakila_snowflake.dim_actor LIMIT 0, 1000	400 row(s) returned	0.00051 sec / 0.0000...
122	01:44:33	SELECT count(*) FROM sakila_snowflake.dim_actor LIMIT 0, 1000	1 row(s) returned	0.00085 sec / 0.000...
123	01:44:59	SELECT count(*) FROM sakila_snowflake.fact_rental LIMIT 0, 1000	1 row(s) returned	0.0021 sec / 0.00000...

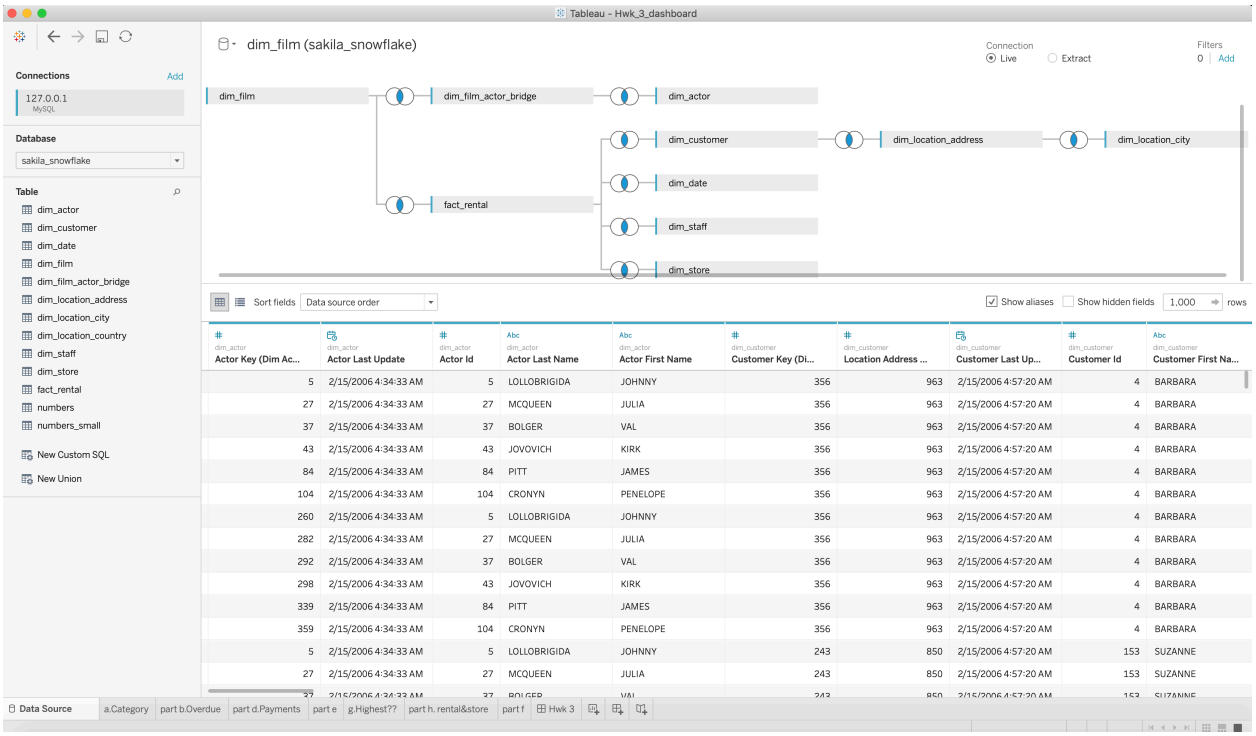


count(*)
16044

I got 16,044 counts for my final result. The total number should be correct.

I will attach the full SQL scripts on Canvas.

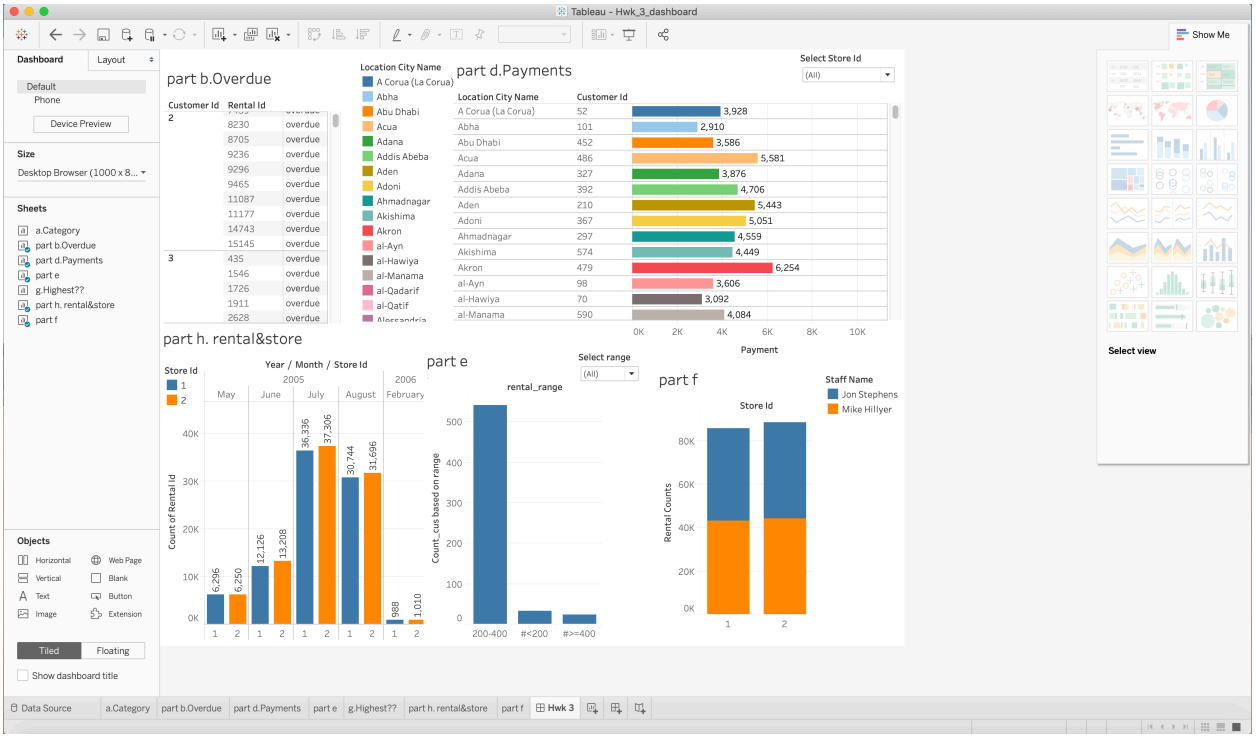
To Show I didn't use any databases other than "sakila_snowflake":



All I have is the sakila_snowflake database. I join ten tables in Tableau to work on the questions from part c.

I will attach my Tableau final dashboard product on Canvas.

The five charts I choose from the question 2 to answer are: Part b, Part d, Part e, Part f, and Part h.



My dashboard preview⬆️

The "**Select Store Id**" filter at the upper-right corner works for **all** the dashboard sections (question parts). I added another filter "**Select range**" for just **part e**.
I will attach my dashboard on Canvas.