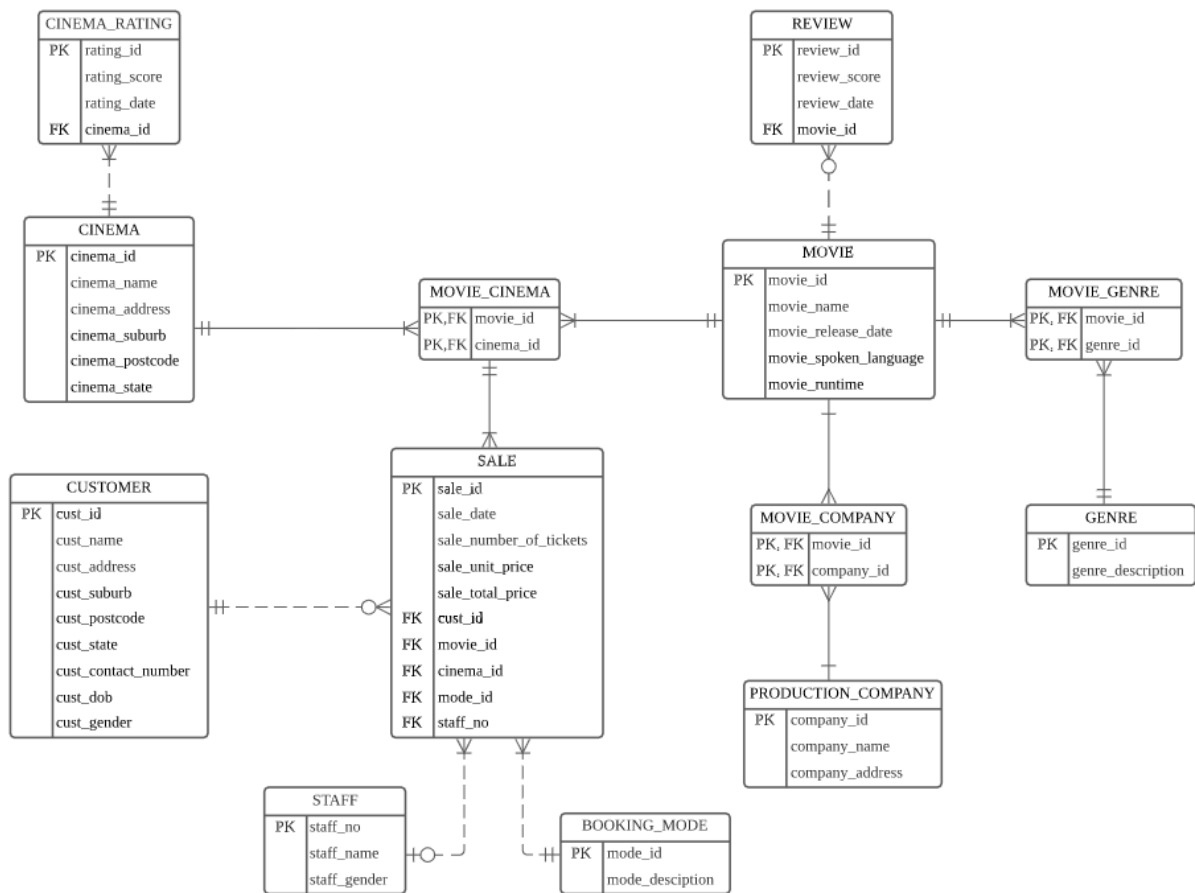


Task C.1

a) E/R Diagram



b) Data Cleaning

BEFORE

-- Find Duplicates

```

SELECT movie_id, COUNT(*)
FROM MonCinema.movie
GROUP BY movie_id
HAVING COUNT(*) > 1;
    
```

	MOVIE_ID	COUNT(*)
1	317	4

```

SELECT company_id, COUNT(*)
FROM MonCinema.production_company
GROUP BY company_id
HAVING COUNT(*) > 1;
    
```

	COMPANY_ID	COUNT(*)
1	18880	4

-- Invalid FK

```
SELECT * FROM MonCinema.cinema_rating WHERE cinema_id NOT IN (SELECT  
cinema_id FROM MonCinema.cinema);
```

	RATING_ID	RATING_SCORE	RATING_DATE	CINEMA_ID
1	700	2	08/07/2017	75

```
SELECT * FROM MonCinema.movie_cinema WHERE cinema_id NOT IN (SELECT  
cinema_id FROM MonCinema.cinema);
```

	MOVIE_ID	CINEMA_ID
1	1234	987

```
SELECT * FROM MonCinema.movie_cinema WHERE movie_id NOT IN (SELECT movie_id  
FROM MonCinema.movie);
```

	MOVIE_ID	CINEMA_ID
1	1234	987

```
SELECT * FROM MonCinema.movie_company WHERE movie_id NOT IN (SELECT  
movie_id FROM MonCinema.movie);
```

	MOVIE_ID	COMPANY_ID
1	1233	18

-- Number of Tickets * Unit Price not equal to Total Price

```
SELECT * FROM MonCinema.sale WHERE (SALE_NUMBER_OF_TICKETS *  
SALE_UNIT_PRICE) != SALE_TOTAL_PRICE;
```

	SALE_ID	SALE_DATE	SALE_NUMBER_OF_TICKETS	SALE_UNIT_PRICE	SALE_TOTAL_PRICE	CUST_ID	MOVIE_ID	CINEMA_ID	MODE_ID	STAFF_NO
1	2500	22/06/2021	1	55	11	131	379	54	1	(null)

-- Mode ID is 1 but no staff

```
SELECT * FROM MonCinema.sale WHERE mode_id = 1 AND staff_no IS NULL;
```

	SALE_ID	SALE_DATE	SALE_NUMBER_OF_TICKETS	SALE_UNIT_PRICE	SALE_TOTAL_PRICE	CUST_ID	MOVIE_ID	CINEMA_ID	MODE_ID	STAFF_NO
1	2500	22/06/2021	1	55	11	131	379	54	1	(null)

-- Invalid Date

```
SELECT * FROM MonCinema.sale WHERE to_date(sale_date, 'DD/MM/YYYY') >  
sysdate;
```

	SALE_ID	SALE_DATE	SALE_NUMBER_OF_TICKETS	SALE_UNIT_PRICE	SALE_TOTAL_PRICE	CUST_ID	MOVIE_ID	CINEMA_ID	MODE_ID	STAFF_NO
1	2501	08/05/2050	1	11	11	97	225	31	3	(null)

-- Null PK

```
SELECT * FROM MonCinema.genre WHERE genre_id IS NULL;
```

	GENRE_ID	GENRE_DESCRIPTION
1	(null)	Undefined

-- Invalid Review Score

```
SELECT * FROM MonCinema.review WHERE review_score < 1 OR review_score > 5;
```

	REVIEW_ID	REVIEW_SCORE	REVIEW_DATE	MOVIE_ID
1	1869	-9	19/08/2016	98122

DATA CLEANING

```
DROP TABLE booking_mode;
CREATE TABLE booking_mode AS
    SELECT * FROM MonCinema.BOOKING_MODE;
```

```
DROP TABLE cinema;
CREATE TABLE cinema AS
    SELECT * FROM MonCinema.CINEMA;
```

```
DROP TABLE cinema_rating;
CREATE TABLE cinema_rating AS
    SELECT DISTINCT * FROM MonCinema.CINEMA_RATING
    WHERE cinema_id IN (SELECT cinema_id FROM MonCinema.CINEMA);
```

```
DROP TABLE customer;
CREATE TABLE customer AS
    SELECT DISTINCT * FROM MonCinema.CUSTOMER;
```

```
DROP TABLE genre;
CREATE TABLE genre AS
    SELECT DISTINCT * FROM MonCinema.GENRE
    WHERE genre_id IS NOT NULL;
```

```
DROP TABLE movie;
CREATE TABLE movie AS
    SELECT DISTINCT * FROM MonCinema.MOVIE;
```

```
DROP TABLE movie_cinema;
CREATE TABLE movie_cinema AS
    SELECT DISTINCT * FROM MonCinema.MOVIE_CINEMA
    WHERE cinema_id IN (SELECT cinema_id FROM MonCinema.CINEMA)
    AND movie_id IN (SELECT movie_id FROM MonCinema.MOVIE);
```

```
DROP TABLE movie_company;
CREATE TABLE movie_company AS
    SELECT DISTINCT * FROM MonCinema.MOVIE_COMPANY
    WHERE movie_id IN (SELECT movie_id FROM MonCinema.MOVIE);
```

```
DROP TABLE movie_genre;
CREATE TABLE movie_genre AS
    SELECT * FROM MonCinema.MOVIE_GENRE;
```

```
DROP TABLE production_company;
CREATE TABLE PRODUCTION_COMPANY AS
    SELECT DISTINCT * FROM MonCinema.PRODUCTION_COMPANY;
```

```
DROP TABLE review;
CREATE TABLE REVIEW AS
    SELECT * FROM MonCinema.REVIEW
    WHERE review_score >= 1 AND review_score <= 5;
```

```
DROP TABLE sale;
CREATE TABLE SALE AS
    SELECT * FROM MonCinema.SALE
    WHERE (SALE_NUMBER_OF_TICKETS * SALE_UNIT_PRICE) = SALE_TOTAL_PRICE
    AND TO_DATE(sale_date, 'DD/MM/YYYY') <= SYSDATE;
```

```
DROP TABLE staff;
CREATE TABLE STAFF AS
    SELECT * FROM MonCinema.STAFF;
```

AFTER

```
SELECT movie_id, COUNT(*)
FROM movie
GROUP BY movie_id
HAVING COUNT(*) > 1;
```

MOVIE_ID	COUNT(*)

```
SELECT company_id, COUNT(*)
FROM production_company
GROUP BY company_id
HAVING COUNT(*) > 1;
```

COMPANY...	COUNT(*)

```
SELECT * FROM cinema_rating WHERE cinema_id NOT IN (SELECT cinema_id FROM
MonCinema.cinema);
```

RATING_ID	RATING_...	RATING_...	CINEMA_ID

```
SELECT * FROM movie_cinema WHERE cinema_id NOT IN (SELECT cinema_id FROM
MonCinema.cinema);
```

MOVIE_ID	CINEMA_ID

```
SELECT * FROM movie_cinema WHERE movie_id NOT IN (SELECT movie_id FROM MonCinema.movie);
```

MOVIE_ID	CINEMA_ID

```
SELECT * FROM movie_company WHERE movie_id NOT IN (SELECT movie_id FROM MonCinema.movie);
```

MOVI...	COMPANY...

```
SELECT * FROM sale WHERE (SALE_NUMBER_OF_TICKETS * SALE_UNIT_PRICE) != SALE_TOTAL_PRICE;
```

SALE_ID	SALE_DATE	SALE_NU...	SALE_UNI...	SALE_TO...	CUST_ID	MOVIE_ID	CINEMA_ID	MODE_ID	STAFF_NO

```
SELECT * FROM sale WHERE mode_id = 1 AND staff_no IS NULL;
```

SALE_ID	SALE_DATE	SALE_NU...	SALE_UNI...	SALE_TO...	CUST_ID	MOVIE_ID	CINEMA_ID	MODE_ID	STAFF_NO

```
SELECT * FROM sale WHERE TO_DATE(sale_date, 'DD/MM/YYYY') > sysdate;
```

SALE_ID	SALE_DATE	SALE_NU...	SALE_UNI...	SALE_TO...	CUST_ID	MOVIE_ID	CINEMA_ID	MODE_ID	STAFF_NO

```
SELECT * FROM review WHERE review_score < 1 OR review_score > 5;
```

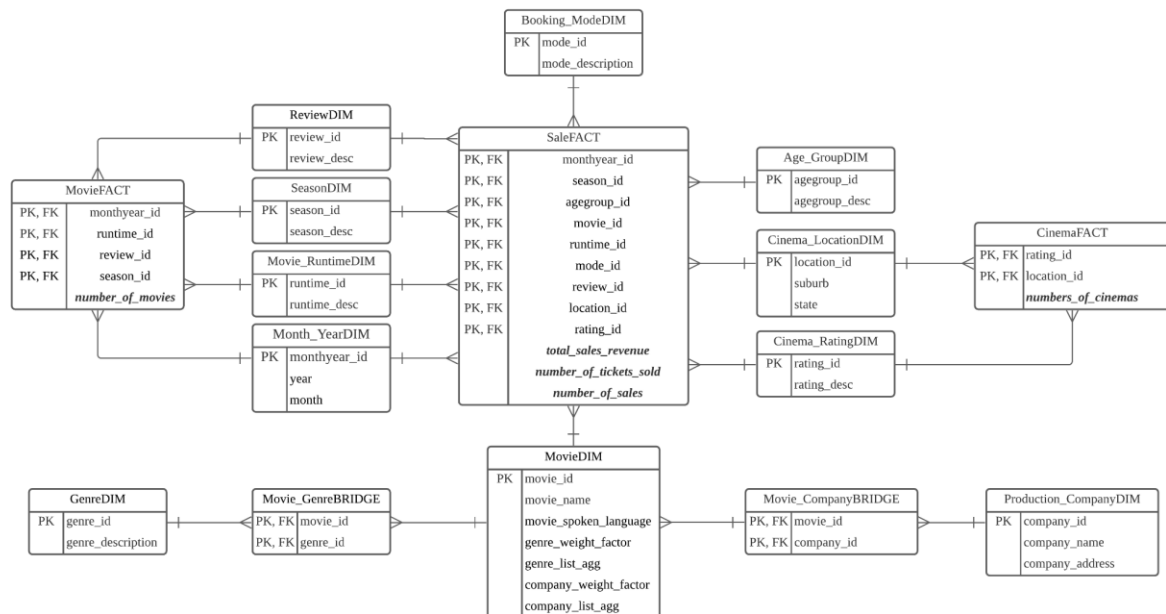
REVIEW_ID	REVIEW_...	REVIEW_...	MOVIE_ID

```
SELECT * FROM genre WHERE genre_id IS NULL;
```

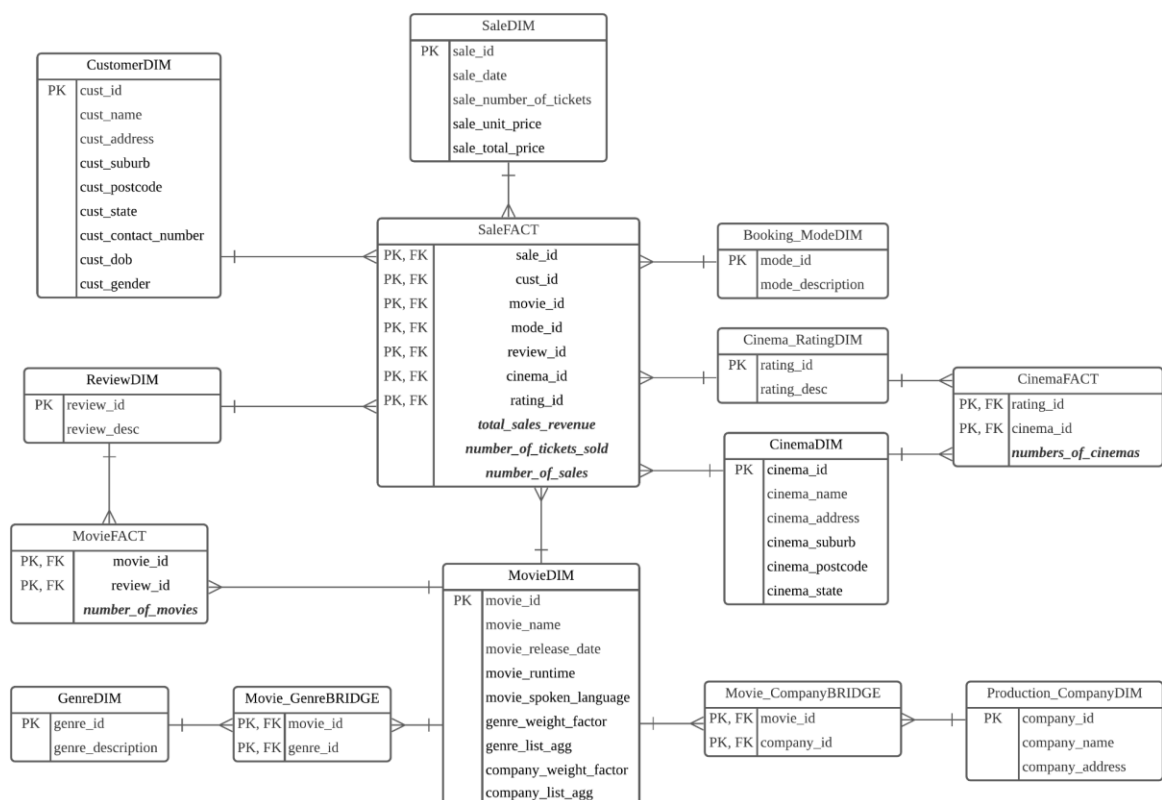
GENRE_ID	GENRE_D...

c) Star Schema Diagrams

Star Schema Level 2 (Version-1)



Star Schema Level 0 (Version-2)



d) SCD Choice

Slowly Changing Dimensions (SCD) was not used in any of the dimensions because during our data exploration, we did not discover any data that changes slowly over time. Hence, there wasn't any implementation of SCD in our case study.

e) Difference Between Star Schemas

The main difference between the Level 2 Star Schema and Level 0 Star Schema is that the Level 0 Star Schema is more detailed compared to the Level 2 Star Schema and the fact measures aggregated by count should be 1 for all rows in the Level 0 Star Schema.

Some of the modifications made to the Level 0 Star Schema from the Level 2 Star Schema include using the customer's date of birth instead of using the customer age group and the runtime id was changed to the movie id as some movies might have the same runtime. Besides, season and month_year were replaced by sale date and movie release date, to ensure there is no aggregation. Cinema location was also replaced by cinema id as one location might have many cinemas.

Task C.2

a) Star Schema Version 1 SQL

```
-- DIMENSIONS
DROP TABLE age_groupDIM;
CREATE TABLE age_groupDIM(
    agegroup_id VARCHAR(20),
    agegroup_desc VARCHAR(20)
);

INSERT INTO age_groupDIM VALUES ('Child', '0-16');
INSERT INTO age_groupDIM VALUES ('Young Adult', '17-30');
INSERT INTO age_groupDIM VALUES ('Middle-Aged Adult', '31-45');
INSERT INTO age_groupDIM VALUES ('Old-Aged Adult', '>45');

DROP TABLE reviewDIM;
CREATE TABLE reviewDIM(
    review_id NUMBER,
    review_desc VARCHAR(20)
);

INSERT INTO reviewDIM VALUES (1, 'Poor');
INSERT INTO reviewDIM VALUES (2, 'Not Good');
INSERT INTO reviewDIM VALUES (3, 'Average');
INSERT INTO reviewDIM VALUES (4, 'Good');
INSERT INTO reviewDIM VALUES (5, 'Excellent');

DROP TABLE seasonDIM;
CREATE TABLE seasonDIM(
    season_id varchar2(10),
    season_desc varchar2(20)
);

INSERT INTO seasonDIM VALUES ('Summer', 'Dec to Feb');
INSERT INTO seasonDIM VALUES ('Autumn', 'Mar to May');
INSERT INTO seasonDIM VALUES ('Winter', 'Jun to Aug');
INSERT INTO seasonDIM VALUES ('Spring', 'Sep to Nov');

DROP TABLE movie_runtimeDIM;
CREATE TABLE movie_runtimeDIM(
    runtime_id VARCHAR(10),
    runtime_desc VARCHAR(40)
);

INSERT INTO movie_runtimeDIM VALUES ('Short', 'Less than 50 minutes');
INSERT INTO movie_runtimeDIM VALUES ('Medium', 'Between 50 and 100
minutes');
INSERT INTO movie_runtimeDIM VALUES ('Long', 'Longer than 100 minutes');
```



```

DROP TABLE cinema_ratingDIM;
CREATE TABLE cinema_ratingDIM(
    rating_id NUMBER,
    rating_desc VARCHAR(20)
);

INSERT INTO cinema_ratingDIM VALUES (1, 'Poor');
INSERT INTO cinema_ratingDIM VALUES (2, 'Not Good');
INSERT INTO cinema_ratingDIM VALUES (3, 'Average');
INSERT INTO cinema_ratingDIM VALUES (4, 'Good');
INSERT INTO cinema_ratingDIM VALUES (5, 'Excellent');

DROP TABLE cinema_locationDIM;
CREATE TABLE cinema_locationDIM AS
    SELECT DISTINCT cinema_suburb || ',' || cinema_state AS location_id,
        cinema_suburb AS suburb,
        cinema_state AS state
    FROM cinema;

DROP TABLE month_yearDIM;
CREATE TABLE month_yearDIM AS
    SELECT DISTINCT TO_CHAR(sale_date, 'MMYYYY') AS monthyear_id,
        TO_CHAR(sale_date, 'Mon') AS Month,
        TO_CHAR(sale_date, 'YYYY') AS Year
    FROM
        (SELECT sale_date FROM sale UNION SELECT movie_release_date
        FROM movie ORDER BY sale_date DESC);

DROP TABLE movieDIM;
CREATE TABLE movieDIM AS
    SELECT g.movie_id, g.movie_name, g.movie_spoken_language,
        g.genre_weight_factor, g.genre_list_agg,
        c.company_weight_factor, c.company_list_agg
    FROM
        (SELECT m.movie_id, m.movie_name, m.movie_spoken_language,
            ROUND(1.0/COUNT(mg.genre_id),4) AS genre_weight_factor,
            LISTAGG (mg.genre_id, '_') WITHIN GROUP
            (ORDER BY mg.genre_id) AS genre_list_agg
        FROM movie m, movie_genre mg
        WHERE m.movie_id = mg.movie_id
        GROUP BY m.movie_id, m.movie_name, m.movie_spoken_language) g,
        (SELECT m.movie_id, m.movie_name, m.movie_spoken_language,
            ROUND(1.0/COUNT(mc.company_id),4) AS company_weight_factor,
            LISTAGG (mc.company_id, '_') WITHIN GROUP
            (ORDER BY mc.company_id) AS company_list_agg
        FROM movie m, movie_company mc
        WHERE m.movie_id = mc.movie_id
        GROUP BY m.movie_id, m.movie_name, m.movie_spoken_language) c
    WHERE g.movie_id = c.movie_id;

```

```

DROP TABLE movie_genreBRIDGE;
CREATE TABLE movie_genreBRIDGE AS
    SELECT * FROM movie_genre;

DROP TABLE genreDIM;
CREATE TABLE genreDIM AS
    SELECT * FROM genre;

DROP TABLE movie_companyBRIDGE;
CREATE TABLE movie_companyBRIDGE AS
    SELECT * FROM movie_company;

DROP TABLE production_companyDIM;
CREATE TABLE production_companyDIM AS
    SELECT * FROM production_company;

DROP TABLE booking_modeDIM;
CREATE TABLE booking_modeDIM AS
    SELECT * FROM booking_mode;

-- FACTS
DROP TABLE cinemaFACT_v1;
CREATE TABLE cinemaFACT_v1 AS
    SELECT c.cinema_suburb || ',' || c.cinema_state AS location_id,
           r.avg_rating_id AS rating_id, COUNT(*) AS number_of_cinemas
    FROM cinema c, (SELECT cinema_id, ROUND(AVG(rating_score)) AS
                     avg_rating_id FROM cinema_rating GROUP BY cinema_id) r
    WHERE c.cinema_id = r.cinema_id
    GROUP BY c.cinema_suburb || ',' || c.cinema_state, r.avg_rating_id;

DROP TABLE movie_tempFACT;
CREATE TABLE movie_tempFACT AS
    SELECT TO_CHAR(m.movie_release_date, 'MMYYYY') AS monthyear_id,
           m.movie_runtime, r.avg_review_id AS review_id
    FROM movie m, (SELECT movie_id, ROUND(AVG(review_score)) AS
                     avg_review_id FROM review GROUP BY movie_id) r
    WHERE m.movie_id = r.movie_id
    GROUP BY m.movie_release_date, m.movie_runtime, r.avg_review_id;

ALTER TABLE movie_tempFACT
    ADD(
        runtime_id VARCHAR(10),
        season_id VARCHAR(10)
    );

UPDATE movie_tempFACT
    SET runtime_id =
        CASE WHEN Movie_Runtime < 50 THEN 'Short'
             WHEN Movie_Runtime >= 50 AND Movie_Runtime <= 100 THEN 'Medium'
             ELSE 'Long'
        END

```

```

        END;
UPDATE movie_tempFACT
    SET season_id =
        CASE WHEN SUBSTR(monthyear_id, 1, 2) BETWEEN '03' AND '05'
            THEN 'Autumn'
            WHEN SUBSTR(monthyear_id, 1, 2) BETWEEN '06' AND '08'
            THEN 'Winter'
            WHEN SUBSTR(monthyear_id, 1, 2) BETWEEN '09' AND '11'
            THEN 'Spring'
            ELSE 'Summer'
        END;

DROP TABLE movieFACT_v1;
CREATE TABLE movieFACT_v1 AS
    SELECT monthyear_id, runtime_id, review_id, season_id,
        COUNT(*) AS number_of_movies
    FROM movie_tempFACT
    GROUP BY monthyear_id, runtime_id, review_id, season_id;

DROP TABLE sale_tempFACT;
CREATE TABLE sale_tempFACT AS
    SELECT TO_CHAR(s.sale_date, 'MMYYYY') AS monthyear_id, s.sale_date,
        c.cust_dob, m.movie_id, m.movie_runtime, s.mode_id,
        r.avg_review_id AS review_id,
        d.cinema_suburb || ',' || d.cinema_state AS location_id,
        t.avg_rating_id AS rating_id,
        s.sale_number_of_tickets, s.sale_total_price
    FROM sale s, customer c, movie m, (SELECT movie_id,
        ROUND(AVG(review_score)) AS avg_review_id FROM review
        GROUP BY movie_id) r, cinema d, (SELECT cinema_id,
        ROUND(AVG(rating_score)) AS avg_rating_id FROM cinema_rating
        GROUP BY cinema_id) t
    WHERE s.cust_id = c.cust_id AND s.movie_id = m.movie_id
        AND s.cinema_id = d.cinema_id AND s.movie_id = r.movie_id
        AND s.cinema_id = t.cinema_id;

ALTER TABLE sale_tempFACT
    ADD (season_id VARCHAR(10),
        agegroup_id VARCHAR(20),
        runtime_id VARCHAR(10));

UPDATE sale_tempFACT
    SET season_id =
        CASE WHEN SUBSTR(monthyear_id, 1, 2) BETWEEN '03' AND '05'
            THEN 'Autumn'
            WHEN SUBSTR(monthyear_id, 1, 2) BETWEEN '06' AND '08'
            THEN 'Winter'
            WHEN SUBSTR(monthyear_id, 1, 2) BETWEEN '09' AND '11'
            THEN 'Spring'
            ELSE 'Summer'
        END;

```

```

UPDATE sale_tempFACT
  SET agegroup_id =
    CASE WHEN FLOOR((SYSDATE - cust_dob)/365) BETWEEN 0 AND 16
      THEN 'Child'
      WHEN FLOOR((SYSDATE - cust_dob)/365) BETWEEN 17 AND 30
      THEN 'Young Adult'
      WHEN FLOOR((SYSDATE - cust_dob)/365) BETWEEN 31 AND 45
      THEN 'Middle-Aged Adult'
      ELSE 'Old-Aged Adult'
    END;

UPDATE sale_tempFACT
  SET runtime_id =
    CASE WHEN Movie_Runtime < 50 THEN 'Short'
      WHEN Movie_Runtime >= 50 AND Movie_Runtime <= 100 THEN 'Medium'
      ELSE 'Long'
    END;

DROP TABLE saleFACT_v1;
CREATE TABLE saleFACT_v1 AS
  SELECT monthyear_id, season_id, agegroup_id, movie_id, runtime_id,
    mode_id, review_id, location_id, rating_id,
    SUM(sale_total_price) AS total_sales_revenue,
    SUM(sale_number_of_tickets) AS number_of_tickets_sold,
    COUNT(*) AS number_of_sales
  FROM sale_tempFACT
  GROUP BY monthyear_id, season_id, agegroup_id, movie_id, runtime_id,
    mode_id, review_id, location_id, rating_id;

```

b) Star Schema Version 2 SQL

```
-- DIMENSIONS
DROP TABLE customerDIM_2;
CREATE TABLE customerDIM_2 AS
    SELECT * FROM customer;

DROP TABLE reviewDIM_2;
CREATE TABLE reviewDIM_2 AS
    SELECT * FROM reviewDIM;

DROP TABLE movieDIM_2;
CREATE TABLE movieDIM_2 AS
    SELECT g.movie_id, g.movie_name, g.movie_release_date, g.movie_runtime,
           g.movie_spoken_language, g.genre_weight_factor,
           g.genre_list_agg, c.company_weight_factor, c.company_list_agg
    FROM
        (SELECT m.movie_id, m.movie_name, m.movie_release_date,
                 movie_runtime, m.movie_spoken_language,
                 ROUND(1.0/COUNT(mg.genre_id),4) AS
genre_weight_factor,
                 LISTAGG (mg.genre_id, '_') WITHIN GROUP
                 (ORDER BY mg.genre_id) AS genre_list_agg
        FROM movie m, movie_genre mg
        WHERE m.movie_id = mg.movie_id
        GROUP BY m.movie_id, m.movie_name, m.movie_release_date,
                 movie_runtime, m.movie_spoken_language) g,
        (SELECT m.movie_id, m.movie_name, m.movie_release_date,
                 movie_runtime, m.movie_spoken_language,
                 ROUND(1.0/COUNT(mc.company_id),4) AS
company_weight_factor,
                 LISTAGG (mc.company_id, '_') WITHIN GROUP
                 (ORDER BY mc.company_id) AS company_list_agg
        FROM movie m, movie_company mc
        WHERE m.movie_id = mc.movie_id
        GROUP BY m.movie_id, m.movie_name, m.movie_release_date,
                 movie_runtime, m.movie_spoken_language) c
    WHERE g.movie_id = c.movie_id;

DROP TABLE movie_genreBRIDGE_2;
CREATE TABLE movie_genreBRIDGE_2 AS
    SELECT * FROM movie_genreBRIDGE;

DROP TABLE genreDIM_2;
CREATE TABLE genreDIM_2 AS
    SELECT * FROM genreDIM;

DROP TABLE movie_companyBRIDGE_2;
CREATE TABLE movie_companyBRIDGE_2 AS
    SELECT * FROM movie_companyBRIDGE;

DROP TABLE production_companyDIM_2;
CREATE TABLE production_companyDIM_2 AS
    SELECT * FROM production_companyDIM;
```

```

DROP TABLE booking_modeDIM_2;
CREATE TABLE booking_modeDIM_2 AS
    SELECT * FROM booking_modeDIM;

DROP TABLE cinema_ratingDIM_2;
CREATE TABLE cinema_ratingDIM_2 AS
    SELECT * FROM cinema_ratingDIM;

DROP TABLE saleDIM_2;
CREATE TABLE saleDIM_2 AS
    SELECT sale_id, sale_date, sale_number_of_tickets, sale_unit_price,
    sale_total_price FROM sale;

DROP TABLE cinemaDIM_2;
CREATE TABLE cinemaDIM_2 AS
    SELECT * FROM cinema;

-- FACTS
DROP TABLE cinemaFACT_v2;
CREATE TABLE cinemaFACT_v2 AS
    SELECT c.cinema_id, r.avg_rating_id AS rating_id,
        COUNT(*) AS number_of_cinemas
    FROM cinema c, (SELECT cinema_id, ROUND(AVG(rating_score))
        AS avg_rating_id FROM cinema_rating GROUP BY cinema_id) r
    WHERE c.cinema_id = r.cinema_id
    GROUP BY c.cinema_id, r.avg_rating_id;

DROP TABLE movieFACT_v2;
CREATE TABLE movieFACT_v2 AS
    SELECT m.movie_id, r.avg_review_id AS review_id,
        COUNT(*) AS number_of_movies
    FROM movie m, (SELECT movie_id, ROUND(AVG(review_score))
        AS avg_review_id FROM review GROUP BY movie_id) r
    WHERE m.movie_id = r.movie_id
    GROUP BY m.movie_id, r.avg_review_id;

DROP TABLE saleFACT_v2;
CREATE TABLE saleFACT_v2 AS
    SELECT s.sale_id, s.cust_id, s.movie_id, s.mode_id, v.review_id,
        s.cinema_id, r.rating_id,
        SUM(s.sale_total_price) AS total_sales_revenue,
        COUNT(s.sale_id) AS number_of_sales,
        SUM(s.sale_number_of_tickets) AS number_of_tickets_sold
    FROM sale s,
        (SELECT cinema_id, ROUND(AVG(rating_score)) AS rating_id
        FROM cinema_rating GROUP BY cinema_id) r,
        (SELECT movie_id, ROUND(AVG(review_score)) AS review_id
        FROM review GROUP BY movie_id) v
    WHERE s.cinema_id = r.cinema_id AND s.movie_id = v.movie_id
    GROUP BY s.sale_id, s.cust_id, s.movie_id, s.mode_id, v.review_id,
        s.cinema_id, r.rating_id;

```

c) Screenshots

Data Cleaning Tables

booking_mode

MODE_ID	MODE_DESCRIPTION
1	1 In-person
2	2 Phone
3	3 Mobile App
4	4 Website

cinema

CINEMA_ID	CINEMA_NAME	CINEMA_ADDRESS	CINEMA_SUBURB	CINEMA_POSTCODE	CINEMA_STATE
1	1 Astor Theatre St Kilda	1 Chapel Street	St Kilda	3182	VIC
2	2 Chinatown Cinema Melbourne	Level 1 - 200 Bourke Street	Melbourne	3000	VIC
3	3 Cinema Nova Melbourne	380 Lygon Street	Melbourne	3053	VIC
4	4 Hoyts Docklands	The District Docklands	Dockland	3008	VIC
5	5 Hoyts Melbourne Central	Melbourne Central Shopping Centre - Cnr Swan...	Melbourne	3000	VIC
6	6 Hoyts Victoria Gardens	VIC Gardens - Corner Burnley and VIC Streets	Richmond	3121	VIC
7	7 IMAX Melbourne	Rathdowne Street	Carlton Gardens	3053	VIC
8	8 Lido Cinemas Hawthorn	1/675 Glenferrie Road	Hawthorn	3122	VIC
9	9 Moonlight Cinema Melbourne	Royal Botanic Gardens - Birdwood Avenue	South Yarra	3141	VIC
10	10 Openair Melbourne (Disney+ Drive-In)	South Beach Reserve	St Kilda	3182	VIC
11	11 Openair Yarra Park	Paddock 20 Yarra Park - Wellington Parade	East Melbourne	3002	VIC
12	12 Palace Balwyn	231 Whitehorse Road	Balwyn	3103	VIC
13	13 Palace Cinema Como	Corner Toorak Road and Chapel Street	South Yarra	3141	VIC
14	14 Palace Kino Cinemas	45 Collins Street	Melbourne	3000	VIC
15	15 Palace Westgarth	89 High Street	Northcote	3070	VIC
16	16 Rooftop Cinema	Level 6 - 252 Swanston Street	Melbourne	3000	VIC
17	17 Village Crown	8 Whiteman Street	Southbank	3006	VIC

cinema_rating

RATING_ID	RATING_SCORE	RATING_DATE	CINEMA_ID
1	3	4 04/02/2010	27
2	4	3 06/02/2010	14
3	6	3 13/02/2010	39
4	19	4 22/04/2010	25
5	24	4 30/05/2010	14
6	25	2 09/06/2010	49
7	27	3 15/06/2010	19
8	36	2 15/07/2010	35
9	59	4 10/10/2010	25
10	71	4 20/11/2010	45
11	72	4 27/11/2010	32
12	80	2 16/12/2010	15
13	85	5 30/12/2010	20
14	92	4 01/02/2011	5
15	99	3 02/03/2011	28
16	103	4 16/03/2011	50
17	108	4 26/03/2011	31

customer

CUST_ID	CUST_NAME	CUST_ADDRESS	CUST_SUBURB	CUST_POSTCODE	CUST_STATE	CUST_CONTACT_NUMBER	CUST_GENDER	CUST_DOB
1	354 Caitlyn Cahill	26 Marley Point Road	DELBURN	3871	VIC	0453296745	F	08/07/2004
2	367 James McCrae	4 Armstrong Street	MEAD	3568	VIC	0453128166	M	30/10/2003
3	380 Amber Loveless	94 Shell Road	CHAPPLE VALE	3239	VIC	0453888362	F	30/07/2003
4	394 Ali Fryer	7 Gaffney Street	CHELSEA	3196	VIC	0490572860	M	11/11/2003
5	401 Anna Snow	42 Wilson Street	YEUNGROON EAST	3525	VIC	0453149623	F	02/10/2003
6	406 Jasmine Stainforth	87 Taltarni Road	GOWAR EAST	3478	VIC	0453287843	F	15/06/1996
7	2 Callum Boyes	60 Walpole Avenue	HEYTESBURY LOWER	3268	VIC	0453145007	M	14/11/2000
8	9 Hugo Miljanovic	59 Bayley Street	EDEN PARK	3757	VIC	0486230732	M	13/07/2002
9	10 Christian Neumayer	74 Edmundsons Road	WALLACE	3352	VIC	0453511154	M	25/03/1997
10	13 Christian Searcy	91 Larissa Court	MILDURA WEST	3500	VIC	0453284792	M	03/06/2004
11	15 Isabella Gilruth	83 Corio Street	ONDIT	3249	VIC	0453937708	F	11/10/1999
12	17 Jackson Kippax	15 Yarra Street	EVANSFORD	3371	VIC	0453910037	M	08/01/1997
13	22 Madeline Mahoney	36 Commercial Street	RIDDELLS CREEK	3431	VIC	0480597913	F	24/04/2001
14	23 Aiden Colbourne	35 Cherokee Road	BARINGHUP	3463	VIC	0453042792	M	17/06/1999
15	30 Aaron Isaacs	57 Rose Street	YARRA JUNCTION	3797	VIC	0499219500	M	30/03/2003
16	37 Adam Boniwell	47 Cherokee Road	DRY DIGGINGS	3461	VIC	0453952634	M	30/06/2001
17	42 Maya Peel	67 Crofts Road	BETE BOLONG NORTH	3888	VIC	0453485221	F	28/11/1998

genre

GENRE_ID	GENRE_DESCRIPTION
1	14 Fantasy
2	12 Adventure
3	80 Crime
4	10751 Family
5	878 Science Fiction
6	9648 Mystery
7	36 History
8	28 Action
9	10402 Music
10	99 Documentary
11	10769 Foreign
12	10770 TV Movie
13	18 Drama
14	35 Comedy
15	27 Horror
16	16 Animation
17	37 Western

movie

MOVIE_ID	MOVIE_NAME	MOVIE_RELEASE_DATE	MOVIE_SPOKEN_LANGUAGE	MOVIE_RUNTIME
1	5072 Severance	19/05/2006	English	96
2	5137 Sky Captain and the World of Tomorrow	17/09/2004	English	107
3	7454 The Bunker	14/09/2001	English	95
4	7942 Run Fatboy Run	06/09/2007	English	100
5	8656 Deep Impact	08/05/1998	English	120
6	8873 Wayne s World 2	10/12/1993	English	95
7	8961 Bad Boys II	18/07/2003	English	147
8	8973 Lord of Illusions	25/08/1995	English	119
9	8983 Felidae	03/11/1994	German	82
10	39183 Once in a Lifetime: The Extraordinary Story of the New York Cosmos	01/01/2006	English	97
11	39231 Man s Favorite Sport?	29/01/1964	English	120
12	45186 Big Bad Mama	19/09/1974	English	83
13	48385 Indestructible Man	24/03/1956	English	72
14	48780 Boat	30/01/2007	English	8
15	6 Judgment Night	15/10/1993	English	110
16	14 American Beauty	15/09/1999	English	122
17	16 Dancer in the Dark	17/05/2000	English	140

movie_cinema

MOVIE_ID	CINEMA_ID
1	293
2	1961
3	2692
4	2976
5	2334
6	292
7	1126
8	3536
9	76
10	41566
11	940
12	3088
13	3396
14	2186
15	2722
16	3556
17	4958

movie_company

MOVIE_ID	COMPANY_ID
1	5
2	16
3	16
4	16
5	21
6	25
7	25
8	26
9	82
10	86
11	107
12	122
13	123
14	141
15	161
16	164
17	165

movie_genre

	MOVIE_ID	GENRE_ID
1	2	18
2	2	80
3	3	18
4	3	35
5	5	35
6	5	80
7	6	28
8	6	53
9	6	80
10	11	12
11	11	28
12	11	878
13	14	18
14	16	18
15	16	80
16	16	10402
17	17	27

production_company

	COMPANY_ID	COMPANY_NAME	COMPANY_ADDRESS
1	985	Filmstiftung NRW	(null)
2	16934	View Askew Productions	(null)
3	1075	Europa Corp	(null)
4	1092	Mockingbird Pictures	(null)
5	5888	Walt Disney	(null)
6	3201	Merlin Films	(null)
7	23310	Karol Film Productions	(null)
8	1239	Aichi Arts Center	(null)
9	1241	Five Star Production	Bangkok Thailand
10	1262	Film Trust S.A.	(null)
11	1264	Alexander Salkind	(null)
12	1296	Crossbow Productions	(null)
13	1315	Charles Chaplin Productions	(null)
14	1314	Hammer Film Productions	London United Kingdom
15	1334	United Productions of America	(null)
16	1343	Classic Films International	(null)
17	779	Lucky Red	(null)

review

	REVIEW_ID	REVIEW_SCORE	REVIEW_DATE	MOVIE_ID
1	1217	4	26/05/2006	2019
2	1218	4	06/02/2000	2020
3	1219	3	13/09/2013	2020
4	1220	4	07/02/2000	2021
5	1221	4	23/09/2005	2021
6	1222	4	26/05/2006	2022
7	1223	2	12/11/2006	2023
8	1224	4	23/09/2005	2023
9	1225	4	10/11/2005	2023
10	1226	3	26/05/2006	2023
11	1227	4	28/02/2011	2028
12	1228	4	01/08/2006	2028
13	1229	4	29/09/1999	2028
14	1230	3	16/08/2001	2028
15	1231	4	23/09/2005	2028
16	1232	4	26/05/2006	2028
17	1233	4	08/02/2000	2033

staff

	STAFF_NO	STAFF_NAME	STAFF_GENDER
1	1	Blake Hovell	M
2	2	Zachary Troup	M
3	3	Seth Canning	M
4	4	Koby Kabu	M
5	5	Eve Goldstein	F
6	6	Olivia De Vis	F
7	7	Evie Clark-Duff	F
8	8	Jayden Shellshear	M
9	9	Grace Wall	F
10	10	Henry Wentcher	M

sale

SALE_ID	SALE_DATE	SALE_NUMBER_OF_TICKETS	SALE_UNIT_PRICE	SALE_TOTAL_PRICE	CUST_ID	MOVIE_ID	CINEMA_ID	MODE_ID	STAFF_NO
1	679 25/12/2018	1	11	11	360	951	19	1	10
2	680 25/12/2018	2	11	22	510	4271	51	4	(null)
3	681 25/12/2018	2	11	22	404	5686	51	3	(null)
4	682 26/12/2018	1	11	11	557	380	38	3	(null)
5	683 26/12/2018	2	11	22	378	150	45	2	(null)
6	684 27/12/2018	5	11	55	488	898	7	1	1
7	685 27/12/2018	1	11	11	344	1411	24	4	(null)
8	686 27/12/2018	1	11	11	119	26131	52	4	(null)
9	687 27/12/2018	1	11	11	498	910	41	3	(null)
10	688 27/12/2018	3	11	33	548	11	19	4	(null)
11	689 27/12/2018	2	11	22	200	3089	47	1	8
12	690 28/12/2018	1	11	11	274	288	37	4	(null)
13	691 28/12/2018	1	11	11	114	1382	50	4	(null)
14	692 29/12/2018	1	11	11	7	2594	58	2	(null)
15	693 29/12/2018	2	11	22	182	1845	21	3	(null)
16	694 29/12/2018	2	11	22	106	1923	32	3	(null)
17	695 30/12/2018	5	11	55	536	968	3	4	(null)

Star Schema Version 1 Dimension Tables

age_groupDIM

AGEGROUP_ID	AGEGROUP_DESC
1 Child	0-16
2 Young Adult	17-30
3 Middle-Aged Adult	31-45
4 Old-Aged Adult	>45

reviewDIM

REVIEW_ID	REVIEW_DESC
1	1 Poor
2	2 Not Good
3	3 Average
4	4 Good
5	5 Excellent

seasonDIM

SEASON_ID	SEASON_DESC
1 Summer	Dec to Feb
2 Autumn	Mar to May
3 Winter	Jun to Aug
4 Spring	Sep to Nov

movie_runtimeDIM

RUNTIME_ID	RUNTIME_DESC
1 Short	Less than 50 minutes
2 Medium	Between 50 and 100 minutes
3 Long	Longer than 100 minutes

cinema_ratingDIM

RATING_ID	RATING_DESC
1	1 Poor
2	2 Not Good
3	3 Average
4	4 Good
5	5 Excellent

cinema_locationDIM

LOCATION_ID	SUBURB	STATE
1 Chadstone,VIC	Chadstone	VIC
2 Melton,VIC	Melton	VIC
3 Werribee,VIC	Werribee	VIC
4 Dandenong South,VIC	Dandenong South	VIC
5 Carlton Gardens,VIC	Carlton Gardens	VIC
6 Coburg ,VIC	Coburg	VIC
7 Epping,VIC	Epping	VIC
8 Altona North,VIC	Altona North	VIC
9 Yarraville,VIC	Yarraville	VIC
10 East Melbourne,VIC	East Melbourne	VIC
11 Mount Waverely,VIC	Mount Waverely	VIC
12 Sunbury,VIC	Sunbury	VIC
13 Brighton,VIC	Brighton	VIC
14 Narre Warren,VIC	Narre Warren	VIC
15 Cheltenham,VIC	Cheltenham	VIC
16 Richmond,VIC	Richmond	VIC
17 Northcote,VIC	Northcote	VIC

month_yearDIM

MONTHYEAR_ID	MONTH	YEAR
1 031920	Mar	1920
2 031926	Mar	1926
3 091927	Sep	1927
4 091932	Sep	1932
5 061936	Jun	1936
6 121939	Dec	1939
7 031942	Mar	1942
8 121945	Dec	1945
9 071950	Jul	1950
10 031952	Mar	1952
11 051961	May	1961
12 101962	Oct	1962
13 061964	Jun	1964
14 021965	Feb	1965
15 021970	Feb	1970
16 101970	Oct	1970
17 071973	Jul	1973

movieDIM

MOVIE_ID	MOVIE_NAME	MOVIE_SPOKEN_LANGUAGE	GENRE_WEIGHT_FACTOR	GENRE_LIST_AGG	COMPANY_WEIGHT_FACTOR	COMPANY_LIST_AGG
1	2 Ariel	Finnish	0.518_80		0.52303_2396	
2	3 Varjoja paratiisissa	Finnish	0.518_35		12303	
3	5 Four Rooms	English	0.535_80		0.514_59	
4	6 Judgment Night	English	0.333328_53_80		0.333333_1644_4248	
5	11 Star Wars	English	0.333312_28_878		0.51_306	
6	14 American Beauty	English	118		0.527_2721	
7	16 Dancer in the Dark	English	0.333318_80_10402		0.04358_76_119_157_201_321_2996_4524_5358_5755_5975	
8	17 The Dark	English	0.333327_53_9648		0.2547_248_2268_2452	
9	18 The Fifth Element	English	0.212_14_28_53_878		0.55_9	
10	19 Metropolis	German	0.518_878		0.54_12372	
11	20 My Life Without Me	English	0.518_10749		0.549_77	
12	21 The Endless Summer	English	199		113723	
13	22 Pirates of the Caribb...	English	0.333312_14_28		0.52_130	
14	24 Kill Bill: Vol. 1	English	0.528_80		0.333314_59_39121	
15	25 Jarhead	English	0.518_10752		0.2533_1522_14440_19934	
16	26 LaLehet Al HaMayim	Hebrew	118		0.515_16	
17	35 The Simpsons Movie	English	0.333316_35_10751		0.518_306	

booking_modeDIM

MODE_ID	MODE_DESCRIPTION
1	1 In-person
2	2 Phone
3	3 Mobile App
4	4 Website

movie_genreBRIDGE

	MOVIE_ID	GENRE_ID
1	2	18
2	2	80
3	3	18
4	3	35
5	5	35
6	5	80
7	6	28
8	6	53
9	6	80
10	11	12
11	11	28
12	11	878
13	14	18
14	16	18
15	16	80
16	16	10402
17	17	27

genreDIM

	GENRE_ID	GENRE_DESCRIPTION
1	14	Fantasy
2	12	Adventure
3	80	Crime
4	10751	Family
5	878	Science Fiction
6	9648	Mystery
7	36	History
8	28	Action
9	10402	Music
10	99	Documentary
11	10769	Foreign
12	10770	TV Movie
13	18	Drama
14	35	Comedy
15	27	Horror
16	16	Animation
17	37	Western

movie_companyBRIDGE

	MOVIE_ID	COMPANY_ID
1	5	14
2	16	76
3	16	157
4	16	11239
5	21	13723
6	25	14440
7	25	19934
8	26	16
9	82	33
10	86	75
11	107	13419
12	122	12
13	123	60
14	141	3334
15	161	172
16	164	11355
17	165	33

production_companyDIM

COMPANY_ID	COMPANY_NAME	COMPANY_ADDRESS
1	985 Filmstiftung NRW	(null)
2	16934 View Askew Productions	(null)
3	1075 Europa Corp	(null)
4	1092 Mockingbird Pictures	(null)
5	5888 Walt Disney	(null)
6	3201 Merlin Films	(null)
7	23310 Karol Film Productions	(null)
8	1239 Aichi Arts Center	(null)
9	1241 Five Star Production	Bangkok Thailand
10	1262 Film Trust S.A.	(null)
11	1264 Alexander Salkind	(null)
12	1296 Crossbow Productions	(null)
13	1315 Charles Chaplin Productions	(null)
14	1314 Hammer Film Productions	London United Kingdom
15	1334 United Productions of America	(null)
16	1343 Classic Films International	(null)
17	779 Lucky Red	(null)

Star Schema Version 1 Fact Tables

cinemaFACT_v1

LOCATION_ID	RATING_ID	NUMBER_OF_CINEMAS
1 Frankston, VIC	4	2
2 Altona North, VIC	4	1
3 Yarraville, VIC	4	1
4 Sunshine, VIC	4	1
5 Chadstone, VIC	4	1
6 Richmond, VIC	4	1
7 Maribyrnong, VIC	4	1
8 Southbank, VIC	4	1
9 Dockland, VIC	4	1
10 Clayton, VIC	4	1
11 Boronia, VIC	4	1
12 Melton, VIC	4	1
13 South Yarra, VIC	4	3
14 Forest Hill, VIC	4	1
15 Melbourne, VIC	4	5
16 Doncaster, VIC	4	1
17 Carlton Gardens, VIC	3	1

movieFACT_v1

MONTHYEAR_ID	RUNTIME_ID	REVIEW_ID	SEASON_ID	NUMBER_OF_MOVIES
1 102001	Medium	4 Spring		2
2 092001	Long	5 Spring		1
3 012002	Long	3 Summer		2
4 021968	Medium	4 Summer		1
5 082001	Long	3 Winter		2
6 121979	Long	4 Summer		1
7 091997	Medium	4 Spring		1
8 092003	Long	5 Spring		1
9 052006	Long	4 Autumn		1
10 011966	Medium	5 Summer		1
11 041999	Medium	4 Autumn		2
12 031988	Medium	5 Autumn		1
13 061966	Medium	4 Winter		1
14 061976	Medium	3 Winter		1
15 081948	Long	4 Winter		1
16 092003	Long	4 Spring		2
17 091998	Medium	3 Spring		1

saleFACT_v1

MONTHYEAR_ID	SEASON_ID	AGEGROUP_ID	MOVIE_ID	RUNTIME_ID	MODE_ID	REVIEW_ID	LOCATION_ID	RATING_ID	TOTAL_SALES_REVENUE	NUMBER_OF_TICKETS_SOLD	NUMBER_OF_SALES
1 122018	Summer	Young Adult	3089 Long		1	4 Airport West, VIC		4	22	2	1
2 012019	Summer	Young Adult	440 Medium		4	4 Brighton, VIC		4	22	2	1
3 012019	Summer	Young Adult	1265 Medium		3	4 South Yarra, VIC		4	11	1	1
4 012019	Summer	Young Adult	8873 Medium		4	5 Burwood East, VIC		4	22	2	1
5 012019	Summer	Young Adult	479 Medium		2	2 Maribyrnong, VIC		4	22	2	1
6 012019	Summer	Young Adult	1266 Long		4	4 Balwyn, VIC		4	11	1	1
7 012019	Summer	Young Adult	1257 Long		1	5 Mount Waverely, VIC		4	11	1	1
8 012019	Summer	Young Adult	207 Long		2	3 Preston, VIC		3	22	2	1
9 012019	Summer	Young Adult	931 Long		3	4 Chadstone, VIC		4	11	1	1
10 022019	Summer	Young Adult	45722 Medium		4	3 Richmond, VIC		4	11	1	1
11 022019	Summer	Young Adult	8665 Long		2	4 Forest Hill, VIC		4	22	2	1
12 022019	Summer	Child	2722 Long		1	2 Elsternwick, VIC		4	22	2	1
13 022019	Summer	Young Adult	95 Long		2	3 Elsternwick, VIC		4	22	2	1
14 022019	Summer	Young Adult	1777 Long		3	4 Altona North, VIC		4	11	1	1
15 022019	Summer	Child	1272 Long		1	3 East Melbourne, VIC		4	22	2	1
16 022019	Summer	Young Adult	588 Long		1	3 Werribee, VIC		4	11	1	1
17 022019	Summer	Young Adult	31696 Medium		1	4 Dockland, VIC		4	11	1	1

Star Schema Version 2 Dimension Tables

customerDIM_2

⚡ CUST_ID ⚡	⚡ CUST_NAME ⚡	⚡ CUST_ADDRESS ⚡	⚡ CUST_SUBURB ⚡	⚡ CUST_POSTCODE ⚡	⚡ CUST_STATE ⚡	⚡ CUST_CONTACT_NUMBER ⚡	⚡ CUST_GENDER ⚡	⚡ CUST_DOB ⚡
1	354 Caitlyn Cahill	26 Marley Point Road	DELBURN	3871	VIC	0453296745	F	08/07/2004
2	367 James McCrae	4 Armstrong Street	MEAD	3568	VIC	0453128166	M	30/10/2003
3	380 Amber Loveless	94 Shell Road	CHAPPLE VALE	3239	VIC	0453888362	F	30/07/2003
4	394 Ali Fryer	7 Gaffney Street	CHELSEA	3196	VIC	0490572860	M	11/11/2003
5	401 Anna Snow	42 Wilson Street	YEUNGROON EAST	3525	VIC	0453149623	F	02/10/2003
6	406 Jasmine Stainforth	87 Taltarni Road	GOWAR EAST	3478	VIC	0453287843	F	15/06/1996
7	2 Callum Boyes	60 Walpole Avenue	HEYTESBURY LOWER	3268	VIC	0453145007	M	14/11/2000
8	9 Hugo Miljanovic	59 Bayley Street	EDEN PARK	3757	VIC	0486230732	M	13/07/2002
9	10 Christian Neumayer	74 Edmundsons Road	WALLACE	3352	VIC	0453511154	M	25/03/1997
10	13 Christian Searcy	91 Larissa Court	MILDURA WEST	3500	VIC	0453284792	M	03/06/2004
11	15 Isabella Gilruth	83 Corio Street	ONDIT	3249	VIC	0453937708	F	11/10/1999
12	17 Jackson Kippax	15 Yarra Street	EVANSFORD	3371	VIC	0453910037	M	08/01/1997
13	22 Madeline Mahoney	36 Commercial Street	RIDDELLS CREEK	3431	VIC	0480597913	F	24/04/2001
14	23 Aiden Colbourne	35 Cherokee Road	BARINGHUP	3463	VIC	0453042792	M	17/06/1999
15	30 Aaron Isaacs	57 Rose Street	YARRA JUNCTION	3797	VIC	0499219500	M	30/03/2003
16	37 Adam Boniwell	47 Cherokee Road	DRY DIGGINGS	3461	VIC	0453952634	M	30/06/2001
17	42 Maya Peel	67 Crofts Road	BETE BOLONG NORTH	3888	VIC	0453485221	F	28/11/1998

reviewDIM_2

⚡ REVIEW_ID ⚡	⚡ REVIEW_DESC ⚡
1	1 Poor
2	2 Not Good
3	3 Average
4	4 Good
5	5 Excellent

booking_modeDIM_2

⚡ MODE_ID ⚡	⚡ MODE_DESCRIPTION ⚡
1	1 In-person
2	2 Phone
3	3 Mobile App
4	4 Website

cinema_ratingDIM_2

⚡ RATING_ID ⚡	⚡ RATING_DESC ⚡
1	1 Poor
2	2 Not Good
3	3 Average
4	4 Good
5	5 Excellent

cinemaDIM_2

⚡ CINEMA_ID ⚡	⚡ CINEMA_NAME ⚡	⚡ CINEMA_ADDRESS ⚡	⚡ CINEMA_SUBURB ⚡	⚡ CINEMA_POSTCODE ⚡	⚡ CINEMA_STATE ⚡
1	1 Astor Theatre St Kilda	1 Chapel Street	St Kilda	3182	VIC
2	2 Chinatown Cinema Melbourne	Level 1 - 200 Bourke Street	Melbourne	3000	VIC
3	3 Cinema Nova Melbourne	380 Lygon Street	Melbourne	3053	VIC
4	4 Hoyts Docklands	The District Docklands	Dockland	3008	VIC
5	5 Hoyts Melbourne Central	Melbourne Central Shopping Centre - Cnr Swan...	Melbourne	3000	VIC
6	6 Hoyts Victoria Gardens	VIC Gardens - Corner Burnley and VIC Streets	Richmond	3121	VIC
7	7 IMAX Melbourne	Rathdowne Street	Carlton Gardens	3053	VIC
8	8 Lido Cinemas Hawthorn	1/675 Glenferrie Road	Hawthorn	3122	VIC
9	9 Moonlight Cinema Melbourne	Royal Botanic Gardens - Birdwood Avenue	South Yarra	3141	VIC
10	10 Openair Melbourne (Disney+ Drive-In)	South Beach Reserve	St Kilda	3182	VIC
11	11 Openair Yarra Park	Paddock 20 Yarra Park - Wellington Parade	East Melbourne	3002	VIC
12	12 Palace Balwyn	231 Whitehorse Road	Balwyn	3103	VIC
13	13 Palace Cinema Como	Corner Toorak Road and Chapel Street	South Yarra	3141	VIC
14	14 Palace Kino Cinemas	45 Collins Street	Melbourne	3000	VIC
15	15 Palace Westgarth	89 High Street	Northcote	3070	VIC
16	16 Rooftop Cinema	Level 6 - 252 Swanston Street	Melbourne	3000	VIC
17	17 Village Crown	8 Whiteman Street	Southbank	3006	VIC

movieDIM_2

MOVIE_ID	MOVIE_NAME	MOVIE_RELEASE_DATE	MOVIE_RUNTIME	MOVIE_SPOKEN_LANGUAGE	GENRE_WEIGHT_FACTOR	GENRE_LIST_AGG	COMPANY_WEIGHT_FACTOR	COMPANY_LIST_AGG
1	2Ariel	21/10/1988	69	Finnish	0.518_80		0.52303_2396	
2	3Varjoja paratiisissa	16/10/1986	76	Finnish	0.518_35		12303	
3	5Four Rooms	09/12/1995	98	English	0.535_80		0.514_59	
4	6Judgment Night	15/10/1993	110	English	0.333328_53_80		0.333333_1644_4248	
5	11Star Wars	25/05/1977	121	English	0.333312_28_878		0.51_306	
6	14American Beauty	15/09/1999	122	English	118		0.527_2721	
7	16Dancer in the Dark	17/05/2000	140	English	0.333318_80_10402		0.04358_76_119_157_201_32	
8	17The Dark	26/01/2006	87	English	0.333327_53_9648		0.2547_248_2268_2452	
9	18The Fifth Element	07/05/1997	126	English	0.212_14_28_53_878		0.55_9	
10	19Metropolis	10/01/1927	153	German	0.518_878		0.54_12372	
11	20My Life Without Me	07/03/2003	106	English	0.518_10749		0.549_77	
12	21The Endless Summer	15/06/1966	95	English	199		113723	
13	22Pirates of the Carib...	09/07/2003	143	English	0.333312_14_28		0.52_130	
14	24Kill Bill: Vol. 1	10/10/2003	111	English	0.528_80		0.333314_59_39121	
15	25Jarhead	04/11/2005	125	English	0.518_10752		0.2533_1522_14440_19934	
16	26LaLehet Al HaMayim	05/02/2004	103	Hebrew	118		0.515_16	
17	35The Simpsons Movie	25/07/2007	87	English	0.333316_35_10751		0.518_306	

production_companyDIM_2

COMPANY_ID	COMPANY_NAME	COMPANY_ADDRESS
1	985Filmstiftung NRW	(null)
2	16934View Askew Productions	(null)
3	1075Europa Corp	(null)
4	1092Mockingbird Pictures	(null)
5	5888Walt Disney	(null)
6	3201Merlin Films	(null)
7	23310Karol Film Productions	(null)
8	1239Aichi Arts Center	(null)
9	1241Five Star Production	Bangkok Thailand
10	1262Film Trust S.A.	(null)
11	1264Alexander Salkind	(null)
12	1296Crossbow Productions	(null)
13	1315Charles Chaplin Productions	(null)
14	1314Hammer Film Productions	London United Kingdom
15	1334United Productions of America	(null)
16	1343Classic Films International	(null)
17	779Lucky Red	(null)

saleDIM_2

SALE_ID	SALE_DATE	SALE_NUMBER_OF_TICKETS	SALE_UNIT_PRICE	SALE_TOTAL_PRICE
1	67925/12/2018	1	11	11
2	68025/12/2018	2	11	22
3	68125/12/2018	2	11	22
4	68226/12/2018	1	11	11
5	68326/12/2018	2	11	22
6	68427/12/2018	5	11	55
7	68527/12/2018	1	11	11
8	68627/12/2018	1	11	11
9	68727/12/2018	1	11	11
10	68827/12/2018	3	11	33
11	68927/12/2018	2	11	22
12	69028/12/2018	1	11	11
13	69128/12/2018	1	11	11
14	69229/12/2018	1	11	11
15	69329/12/2018	2	11	22
16	69429/12/2018	2	11	22
17	69530/12/2018	5	11	55

movie_genreBRIDGE_2

MOVIE_ID	GENRE_ID
1	2
2	2
3	3
4	3
5	5
6	5
7	6
8	6
9	6
10	11
11	11
12	11
13	14
14	16
15	16
16	16
17	17

genreDIM_2

	GENRE_ID	GENRE_DESCRIPTION
1	14	Fantasy
2	12	Adventure
3	80	Crime
4	10751	Family
5	878	Science Fiction
6	9648	Mystery
7	36	History
8	28	Action
9	10402	Music
10	99	Documentary
11	10769	Foreign
12	10770	TV Movie
13	18	Drama
14	35	Comedy
15	27	Horror
16	16	Animation
17	37	Western

movie_companyBRIDGE_2

	MOVIE_ID	COMPANY_ID
1	5	14
2	16	76
3	16	157
4	16	11239
5	21	13723
6	25	14440
7	25	19934
8	26	16
9	82	33
10	86	75
11	107	13419
12	122	12
13	123	60
14	141	3334
15	161	172
16	164	11355
17	165	33

Star Schema Version 2 Fact Tables

cinemaFACT_v2

	CINEMA_ID	RATING_ID	NUMBER_OF_CINEMAS
1	17	4	1
2	30	4	1
3	45	4	1
4	4	4	1
5	10	3	1
6	12	4	1
7	29	4	1
8	31	4	1
9	43	3	1
10	44	4	1
11	51	4	1
12	56	4	1
13	57	3	1
14	1	4	1
15	6	4	1
16	20	3	1
17	32	4	1

movieFACT_v2

	MOVIE_ID	REVIEW_ID	NUMBER_OF_MOVIES
1	2133	4	1
2	2761	4	1
3	2959	4	1
4	3019	4	1
5	166	2	1
6	277	5	1
7	663	3	1
8	762	2	1
9	786	3	1
10	851	4	1
11	948	4	1
12	965	4	1
13	1371	3	1
14	1597	3	1
15	1653	4	1
16	1807	2	1
17	4639	1	1

saleFACT_v2

	SALE_ID	CUST_ID	MOVIE_ID	MODE_ID	REVIEW_ID	CINEMA_ID	RATING_ID	TOTAL_SALES_REVENUE	NUMBER_OF_SALES	NUMBER_OF_TICKETS_SOLD
1	690	274	288	4	3	37	3	11	1	1
2	695	536	968	4	5	3	3	55	1	5
3	710	522	7008	2	3	27	4	33	1	3
4	711	183	3052	4	3	24	4	11	1	1
5	725	329	7348	2	2	10	3	11	1	1
6	736	65	26	3	3	47	4	11	1	1
7	750	377	1266	4	4	12	4	11	1	1
8	761	477	858	2	4	10	3	11	1	1
9	772	491	24	2	4	29	4	44	1	4
10	773	348	4701	3	4	53	4	11	1	1
11	786	395	1127	1	4	10	3	11	1	1
12	787	319	163	4	4	54	4	11	1	1
13	791	486	86332	4	1	6	4	22	1	2
14	797	482	140174	4	4	60	4	55	1	5
15	798	345	329	2	3	6	4	55	1	5
16	811	533	3577	1	4	36	4	11	1	1
17	820	168	70862	3	3	41	4	22	1	2

Task C.3

a) Simple Reports

REPORT 1 (TOP k)

Query question:

What are the top 5 Science Fiction movies that are longer than 100 minutes with the highest sales revenue?

Explanation:

This query will help the management find out which long movies that are under the genre of Science Fiction provided the highest revenue.

Level 2

```
SELECT s.movie_id, m.movie_name,
       SUM(s.total_sales_revenue) AS TOTAL_SALES_REVENUE
FROM saleFACT_v1 s, movieDIM m
WHERE s.movie_id IN (SELECT movie_id FROM movie_genreBRIDGE
                     WHERE genre_id = (SELECT genre_id FROM genreDIM
                                       WHERE genre_description = 'Science Fiction'))
AND s.runtime_id = (SELECT runtime_id FROM movie_runtimeDIM
                   WHERE runtime_desc = 'Longer than 100 minutes')
AND s.movie_id = m.movie_id
GROUP BY s.movie_id, m.movie_name
ORDER BY SUM(s.total_sales_revenue) DESC
FETCH NEXT 5 ROWS ONLY;
```

	MOVIE_ID	MOVIE_NAME	TOTAL_SALES_REVENUE
1	281	Strange Days	209
2	152	Star Trek: The Motion Picture	143
3	2134	The Time Machine	132
4	4723	Southland Tales	121
5	180	Minority Report	110

Level 0

```
SELECT s.movie_id, m.movie_name,
       SUM(s.total_sales_revenue) AS TOTAL_SALES_REVENUE
FROM saleFACT_v2 s, movieDIM_2 m
WHERE s.movie_id IN (SELECT movie_id FROM movie_genreBRIDGE_2
                     WHERE genre_id = (SELECT genre_id FROM genreDIM_2
                                       WHERE genre_description = 'Science Fiction'))
AND m.movie_runtime > 100
AND s.movie_id = m.movie_id
GROUP BY s.movie_id, m.movie_name
ORDER BY SUM(s.total_sales_revenue) DESC
FETCH NEXT 5 ROWS ONLY;
```

	MOVIE_ID	MOVIE_NAME	TOTAL_SALES_REVENUE
1	281	Strange Days	209
2	152	Star Trek: The Motion Picture	143
3	2134	The Time Machine	132
4	4723	Southland Tales	121
5	180	Minority Report	110

REPORT 2 (TOP n%)

Query question:

What are the top 20% English movies with Excellent reviews that have the highest number of tickets sold booked through the mobile app?

Explanation:

This query will help the management find out how many English movies with Excellent reviews were booked through the mobile app.

Level 2

```
SELECT s.movie_id, m.movie_name,
       SUM(s.number_of_tickets_sold) AS NUMBER_OF_TICKETS_SOLD
FROM saleFACT_v1 s, movieDIM m
WHERE s.review_id = (SELECT review_id FROM reviewDIM
                     WHERE review_desc = 'Excellent')
      AND s.mode_id = (SELECT mode_id FROM booking_modeDIM
                       WHERE mode_description = 'Mobile App')
      AND m.movie_spoken_language = 'English'
      AND s.movie_id = m.movie_id
GROUP BY s.movie_id, m.movie_name
ORDER BY SUM(s.number_of_tickets_sold) DESC
FETCH FIRST 20 PERCENT ROWS ONLY;
```

	MOVIE_ID	MOVIE_NAME	NUMBER_OF_TICKETS_SOLD
1	534	Terminator Salvation	9
2	928	Gremlins 2: The New Batch	5
3	2926	The Three Musketeers	5
4	4772	Cul-de-sac	5
5	111759	Don Q Son of Zorro	5
6	38061	Anybody s Son Will Do	5
7	2114	Final Fantasy: The Spirits Within	4
8	2690	Irma la Douce	3
9	2144	One Night at McCool s	3
10	1267	Meet the Robinsons	3

Level 0

```
SELECT s.movie_id, m.movie_name,
       SUM(s.number_of_tickets_sold) AS NUMBER_OF_TICKETS_SOLD
FROM saleFACT_v2 s, movieDIM_2 m
WHERE s.review_id = (SELECT review_id FROM reviewDIM_2
                     WHERE review_desc = 'Excellent')
      AND s.mode_id = (SELECT mode_id FROM booking_modeDIM_2
                       WHERE mode_description = 'Mobile App')
      AND m.movie_spoken_language = 'English'
      AND s.movie_id = m.movie_id
GROUP BY s.movie_id, m.movie_name
ORDER BY SUM(s.number_of_tickets_sold) DESC
FETCH FIRST 20 PERCENT ROWS ONLY;
```

	MOVIE_ID	MOVIE_NAME	NUMBER_OF_TICKETS_SOLD
1	534	Terminator Salvation	9
2	928	Gremlins 2: The New Batch	5
3	2926	The Three Musketeers	5
4	4772	Cul-de-sac	5
5	111759	Don Q Son of Zorro	5
6	38061	Anybody s Son Will Do	5
7	2114	Final Fantasy: The Spirits Within	4
8	2690	Irma la Douce	3
9	2144	One Night at McCool s	3
10	1267	Meet the Robinsons	3

REPORT 3 (SHOW ALL)

Query question:

What is the total number of sales per month in 2020 that was done through the website?

Explanation:

This query will help the management find out how many times the website is used for booking each month in 2020.

Level 2

```
SELECT monthyear_id, mode_id, SUM(number_of_sales) AS NUMBER_OF_SALES
FROM saleFACT_v1
WHERE monthyear_id LIKE '%2020%'
      AND mode_id = (SELECT mode_id FROM booking_modeDIM
                     WHERE mode_description = 'Website')
GROUP BY monthyear_id, mode_id
ORDER BY monthyear_id;
```

	MONTHYEAR_ID	MODE_ID	NUMBER_OF_SALES
1	012020	4	15
2	022020	4	11
3	032020	4	12
4	042020	4	11
5	052020	4	19
6	062020	4	11
7	072020	4	18
8	082020	4	16
9	092020	4	14
10	102020	4	14
11	112020	4	8
12	122020	4	15

Level 0

```
SELECT TO_CHAR(t.sale_date, 'MMYYYY') AS monthyear_id, s.mode_id,
      SUM(s.number_of_sales) AS NUMBER_OF_SALES
FROM saleFACT_v2 s, saleDIM_2 t
WHERE TO_CHAR(t.sale_date, 'YYYY') = '2020'
      AND s.mode_id = (SELECT mode_id FROM booking_modeDIM
                     WHERE mode_description = 'Website')
      AND s.sale_id = t.sale_id
GROUP BY TO_CHAR(t.sale_date, 'MMYYYY'), mode_id
ORDER BY TO_CHAR(t.sale_date, 'MMYYYY');
```

	MONTHYEAR_ID	MODE_ID	NUMBER_OF_SALES
1	012020	4	15
2	022020	4	11
3	032020	4	12
4	042020	4	11
5	052020	4	19
6	062020	4	11
7	072020	4	18
8	082020	4	16
9	092020	4	14
10	102020	4	14
11	112020	4	8
12	122020	4	15

b) Reports with Proper Sub-Totals

REPORT 4 (CUBE)

Level 2

```
SELECT
    DECODE(GROUPING(location_id), 1, 'All Locations', location_id)
        AS Location,
    DECODE(GROUPING(season_id), 1, 'All Seasons', season_id) AS Seasons,
    DECODE(GROUPING(runtime_id), 1, 'All Movie Runtime Categories',
        runtime_id) AS Movie_Runtime_Category,
    SUM(total_sales_revenue) AS TOTAL_SALES_REVENUE
FROM saleFACT_v1
GROUP BY CUBE(location_id, season_id, runtime_id);
```

LOCATION	SEASONS	MOVIE_RUNTIME_CATEGORY	TOTAL_SALES_REVENUE
1 All Locations	All Seasons	All Movie Runtime Categories	46970
2 All Locations	All Seasons	Long	28204
3 All Locations	All Seasons	Short	506
4 All Locations	All Seasons	Medium	18260
5 All Locations	Autumn	All Movie Runtime Categories	13420
6 All Locations	Autumn	Long	8349
7 All Locations	Autumn	Short	77
8 All Locations	Autumn	Medium	4994
9 All Locations	Spring	All Movie Runtime Categories	9801
10 All Locations	Spring	Long	5797
11 All Locations	Spring	Short	110
12 All Locations	Spring	Medium	3894
13 All Locations	Summer	All Movie Runtime Categories	12628
14 All Locations	Summer	Long	7502
15 All Locations	Summer	Short	242

Level 0

```
SELECT
    DECODE(GROUPING(c.cinema_suburb ||', '|| c.cinema_state), 1,
            'All Locations',
            c.cinema_suburb ||', '|| c.cinema_state) AS Location,
    DECODE(GROUPING(sa.season), 1, 'All Seasons', sa.season) AS Seasons,
    DECODE(GROUPING(m.runtime_category), 1, 'All Movie Runtime Categories',
            m.runtime_category) AS Movie_Runtime_Category,
    SUM(total_sales_revenue) AS TOTAL_SALES_REVENUE
FROM
    saleFACT_v2 s,
    (SELECT DISTINCT sale_id,
        CASE
            WHEN TO_CHAR(sale_date, 'MM') BETWEEN '03' AND '05'
            THEN 'Autumn'
            WHEN TO_CHAR(sale_date, 'MM') BETWEEN '06' AND '08'
            THEN 'Winter'
            WHEN TO_CHAR(sale_date, 'MM') BETWEEN '09' AND '11'
            THEN 'Spring'
            ELSE 'Summer'
        END AS season
    FROM saleDIM_2) sa,
    (SELECT DISTINCT movie_id,
        CASE
            WHEN movie_runtime < 50 THEN 'Short'
            WHEN movie_runtime >= 50 AND movie_runtime <= 100 THEN 'Medium'
            ELSE 'Long'
        END AS RUNTIME_CATEGORY
    FROM movieDIM_2) m,
    cinemaDIM_2 c
WHERE s.sale_id = sa.sale_id
AND s.movie_id = m.movie_id
AND s.cinema_id = c.cinema_id
GROUP BY CUBE(c.cinema_suburb ||', '|| c.cinema_state, sa.season,
              m.runtime_category);
```

	LOCATION	SEASONS	MOVIE_RUNTIME_CATEGORY	TOTAL_SALES_REVENUE
1	All Locations	All Seasons	All Movie Runtime Categories	46970
2	All Locations	All Seasons	Long	28204
3	All Locations	All Seasons	Short	506
4	All Locations	All Seasons	Medium	18260
5	All Locations	Autumn	All Movie Runtime Categories	13420
6	All Locations	Autumn	Long	8349
7	All Locations	Autumn	Short	77
8	All Locations	Autumn	Medium	4994
9	All Locations	Spring	All Movie Runtime Categories	9801
10	All Locations	Spring	Long	5797
11	All Locations	Spring	Short	110
12	All Locations	Spring	Medium	3894
13	All Locations	Summer	All Movie Runtime Categories	12628
14	All Locations	Summer	Long	7502
15	All Locations	Summer	Short	242

REPORT 5 (PARTIAL CUBE)

Level 2

```
SELECT
    DECODE(GROUPING(location_id), 1, 'All Locations', location_id)
        AS Location,
    DECODE(GROUPING(season_id), 1, 'All Seasons', season_id) AS Seasons,
    DECODE(GROUPING(runtime_id), 1, 'All Movie Runtime Categories',
        runtime_id) AS Movie_Runtime_Category,
    SUM(total_sales_revenue) AS TOTAL_SALES_REVENUE
FROM saleFACT_v1
GROUP BY CUBE(season_id, runtime_id), location_id;
```

LOCATION	SEASONS	MOVIE_RUNTIME_CATEGORY	TOTAL_SALES_REVENUE
1 Balwyn, VIC	All Seasons	All Movie Runtime Categories	583
2 Balwyn, VIC	All Seasons	Long	407
3 Balwyn, VIC	All Seasons	Medium	176
4 Balwyn, VIC	Autumn	All Movie Runtime Categories	165
5 Balwyn, VIC	Autumn	Long	110
6 Balwyn, VIC	Autumn	Medium	55
7 Balwyn, VIC	Spring	All Movie Runtime Categories	99
8 Balwyn, VIC	Spring	Long	77
9 Balwyn, VIC	Spring	Medium	22
10 Balwyn, VIC	Summer	All Movie Runtime Categories	132
11 Balwyn, VIC	Summer	Long	88
12 Balwyn, VIC	Summer	Medium	44
13 Balwyn, VIC	Winter	All Movie Runtime Categories	187
14 Balwyn, VIC	Winter	Long	132
15 Balwyn, VIC	Winter	Medium	55

Level 0

```
SELECT
    DECODE(GROUPING(c.cinema_suburb || ',' || c.cinema_state), 1,
            'All Locations', c.cinema_suburb || ',' || c.cinema_state)
        AS Location,
    DECODE(GROUPING(sa.season), 1, 'All Seasons', sa.season) AS Seasons,
    DECODE(GROUPING(m.runtime_category), 1, 'All Movie Runtime Categories',
            m.runtime_category) AS Movie_Runtime_Category,
    SUM(total_sales_revenue) AS TOTAL_SALES_REVENUE
FROM
    saleFACT_v2 s,
    (SELECT DISTINCT sale_id,
        CASE
            WHEN TO_CHAR(sale_date, 'MM') BETWEEN '03' AND '05'
            THEN 'Autumn'
            WHEN TO_CHAR(sale_date, 'MM') BETWEEN '06' AND '08'
            THEN 'Winter'
            WHEN TO_CHAR(sale_date, 'MM') BETWEEN '09' AND '11'
            THEN 'Spring'
            ELSE 'Summer'
        END AS season
    FROM saleDIM_2) sa,
    (SELECT DISTINCT movie_id,
        CASE
            WHEN movie_runtime < 50 THEN 'Short'
            WHEN movie_runtime >= 50 AND movie_runtime <= 100 THEN 'Medium'
            ELSE 'Long'
        END AS RUNTIME_CATEGORY
    FROM movieDIM_2) m,
    cinemaDIM_2 c
WHERE s.sale_id = sa.sale_id
AND s.movie_id = m.movie_id
AND s.cinema_id = c.cinema_id
GROUP BY CUBE(sa.season, m.runtime_category),
           c.cinema_suburb || ', ' || c.cinema_state;
```

	LOCATION	SEASONS	MOVIE_RUNTIME_CATEGORY	TOTAL_SALES_REVENUE
1	Balwyn, VIC	All Seasons	All Movie Runtime Categories	583
2	Balwyn, VIC	All Seasons	Long	407
3	Balwyn, VIC	All Seasons	Medium	176
4	Balwyn, VIC	Autumn	All Movie Runtime Categories	165
5	Balwyn, VIC	Autumn	Long	110
6	Balwyn, VIC	Autumn	Medium	55
7	Balwyn, VIC	Spring	All Movie Runtime Categories	99
8	Balwyn, VIC	Spring	Long	77
9	Balwyn, VIC	Spring	Medium	22
10	Balwyn, VIC	Summer	All Movie Runtime Categories	132
11	Balwyn, VIC	Summer	Long	88
12	Balwyn, VIC	Summer	Medium	44
13	Balwyn, VIC	Winter	All Movie Runtime Categories	187
14	Balwyn, VIC	Winter	Long	132
15	Balwyn, VIC	Winter	Medium	55

REPORT 6 (ROLLUP)

Query question:

What are the total number of sales and subtotals of rolling-up aggregate combinations of the specified attributes: age group category, season and booking mode? (Interested in total number of sales of each booking mode)

Explanation:

Management can change their booking mode to phone, mobile app or website to avoid physical contact if more than half of the customers use these booking modes.

Level 2

```
SELECT
    DECODE(GROUPING(mode_description), 1, 'All Booking Modes',
           mode_description) AS BookingModes,
    DECODE(GROUPING(agegroup_id), 1, 'All AgeGroups', agegroup_id) AS
        Agegroups,
    DECODE(GROUPING(season_id), 1, 'All Seasons', season_id) AS Seasons,
    SUM(number_of_sales) AS NUMBER_OF_SALES
FROM saleFACT_v1 s, booking_modeDIM b
WHERE s.mode_id = b.mode_id
GROUP BY ROLLUP(mode_description, agegroup_id, season_id)
ORDER BY mode_description DESC, agegroup_id DESC, season_id DESC;
```

	BOOKINGMODES	AGEGROUPS	SEASONS	NUMBER_OF_SALES
1	All Booking Modes	All AgeGroups	All Seasons	2499
2	Website	All AgeGroups	All Seasons	635
3	Website	Young Adult	All Seasons	578
4	Website	Young Adult	Winter	141
5	Website	Young Adult	Summer	144
6	Website	Young Adult	Spring	126
7	Website	Young Adult	Autumn	167
8	Website	Child	All Seasons	57
9	Website	Child	Winter	11
10	Website	Child	Summer	16
11	Website	Child	Spring	8
12	Website	Child	Autumn	22
13	Phone	All AgeGroups	All Seasons	611
14	Phone	Young Adult	All Seasons	547
15	Phone	Young Adult	Winter	143

Level 0

```
SELECT
    DECODE(GROUPING(b.mode_description), 1, 'All Booking Modes',
           b.mode_description) AS BookingModes,
    DECODE(GROUPING(c.agegroup), 1, 'All AgeGroups', c.agegroup)
      AS Agegroups,
    DECODE(GROUPING(sa.season), 1, 'All Seasons', sa.season) AS Seasons,
    SUM(number_of_sales) AS NUMBER_OF_SALES
FROM
    saleFACT_v2 s,
    (SELECT DISTINCT sale_id,
        CASE
            WHEN TO_CHAR(sale_date, 'MM') BETWEEN '03' AND '05'
            THEN 'Autumn'
            WHEN TO_CHAR(sale_date, 'MM') BETWEEN '06' AND '08'
            THEN 'Winter'
            WHEN TO_CHAR(sale_date, 'MM') BETWEEN '09' AND '11'
            THEN 'Spring'
            ELSE 'Summer'
        END AS season
    FROM saleDIM_2) sa,
    (SELECT cust_id,
        CASE WHEN FLOOR((SYSDATE - cust_dob)/365) BETWEEN 0 AND 16
            THEN 'Child'
            WHEN FLOOR((SYSDATE - cust_dob)/365) BETWEEN 17 AND 30
            THEN 'Young Adult'
            WHEN FLOOR((SYSDATE - cust_dob)/365) BETWEEN 31 AND 45
            THEN 'Middle-Aged Adult'
            ELSE 'Old-Aged Adult'
        END AS agegroup
    FROM customerDIM_2) c,
    booking_modeDIM_2 b
WHERE s.sale_id = sa.sale_id
AND s.cust_id = c.cust_id
AND s.mode_id = b.mode_id
GROUP BY ROLLUP(b.mode_description, c.agegroup, sa.season)
ORDER BY b.mode_description DESC, agegroup DESC, season DESC;
```

BOOKINGMODES	AGEGROUPS	SEASONS	NUMBER_OF_SALES
1 All Booking Modes	All AgeGroups	All Seasons	2499
2 Website	All AgeGroups	All Seasons	635
3 Website	Young Adult	All Seasons	578
4 Website	Young Adult	Winter	141
5 Website	Young Adult	Summer	144
6 Website	Young Adult	Spring	126
7 Website	Young Adult	Autumn	167
8 Website	Child	All Seasons	57
9 Website	Child	Winter	11
10 Website	Child	Summer	16
11 Website	Child	Spring	8
12 Website	Child	Autumn	22
13 Phone	All AgeGroups	All Seasons	611
14 Phone	Young Adult	All Seasons	547
15 Phone	Young Adult	Winter	143

REPORT 7 (PARTIAL ROLLUP)

Level 2

```
SELECT
    DECODE (GROUPING (agegroup_id), 1, 'All AgeGroups', agegroup_id)
        AS Agegroups,
    DECODE (GROUPING (season_id), 1, 'All Seasons', season_id) AS Seasons,
    DECODE (GROUPING (b.mode_description), 1, 'All Booking Modes',
        b.mode_description) AS BookingModes,
    SUM (number_of_sales) AS NUMBER_OF_SALES
FROM saleFACT_v1 s, booking_modeDIM b
WHERE s.mode_id = b.mode_id
GROUP BY ROLLUP (agegroup_id, season_id), b.mode_description;
```

	AGEGROUPS	SEASONS	BOOKINGMODES	NUMBER_OF_SALES
1	Child	Autumn	Phone	19
2	Child	Spring	Phone	18
3	Child	Summer	Phone	13
4	Child	Winter	Phone	14
5	Child	All Seasons	Phone	64
6	Young Adult	Autumn	Phone	149
7	Young Adult	Spring	Phone	116
8	Young Adult	Summer	Phone	139
9	Young Adult	Winter	Phone	143
10	Young Adult	All Seasons	Phone	547
11	All AgeGroups	All Seasons	Phone	611
12	Child	Autumn	Website	22
13	Child	Spring	Website	8
14	Child	Summer	Website	16
15	Child	Winter	Website	11

Level 0

```
SELECT
    DECODE(GROUPING(c.agegroup), 1, 'All AgeGroups', c.agegroup)
        AS Agegroups,
    DECODE(GROUPING(sa.season), 1, 'All Seasons', sa.season) AS Seasons,
    DECODE(GROUPING(b.mode_description), 1, 'All Booking Modes',
b.mode_description)
        AS BookingModes,
    SUM(number_of_sales) AS NUMBER_OF_SALES
FROM
    saleFACT_v2 s,
    (SELECT DISTINCT sale_id,
        CASE
            WHEN TO_CHAR(sale_date, 'MM') BETWEEN '03' AND '05'
            THEN 'Autumn'
            WHEN TO_CHAR(sale_date, 'MM') BETWEEN '06' AND '08'
            THEN 'Winter'
            WHEN TO_CHAR(sale_date, 'MM') BETWEEN '09' AND '11'
            THEN 'Spring'
            ELSE 'Summer'
        END AS season
    FROM saleDIM_2) sa,
    (SELECT cust_id,
        CASE WHEN FLOOR((SYSDATE - cust_dob)/365) BETWEEN 0 AND 16
            THEN 'Child'
            WHEN FLOOR((SYSDATE - cust_dob)/365) BETWEEN 17 AND 30
            THEN 'Young Adult'
            WHEN FLOOR((SYSDATE - cust_dob)/365) BETWEEN 31 AND 45
            THEN 'Middle-Aged Adult'
            ELSE 'Old-Aged Adult'
        END AS agegroup
    FROM customerDIM_2) c,
    booking_modeDIM_2 b
WHERE s.sale_id = sa.sale_id
AND s.cust_id = c.cust_id
AND s.mode_id = b.mode_id
GROUP BY ROLLUP(c.agegroup, sa.season), b.mode_description;
```

AGEGROUPS	SEASONS	BOOKINGMODES	NUMBER_OF_SALES
1 Child	Autumn	Phone	19
2 Child	Spring	Phone	18
3 Child	Summer	Phone	13
4 Child	Winter	Phone	14
5 Child	All Seasons	Phone	64
6 Young Adult	Autumn	Phone	149
7 Young Adult	Spring	Phone	116
8 Young Adult	Summer	Phone	139
9 Young Adult	Winter	Phone	143
10 Young Adult	All Seasons	Phone	547
11 All AgeGroups	All Seasons	Phone	611
12 Child	Autumn	Website	22
13 Child	Spring	Website	8
14 Child	Summer	Website	16
15 Child	Winter	Website	11

c) Reports with Moving and Cumulative Aggregate

REPORT 8

Level 2

```
SELECT m.year, SUM(total_sales_revenue) AS TOTAL_SALES,  
       SUM(SUM(total_sales_revenue)) OVER  
         (ORDER BY m.year ROWS UNBOUNDED PRECEDING) AS CUMULATIVE_TOTAL_SALES  
FROM saleFACT_v1 s, month_yearDIM m, movie_genreBRIDGE g  
WHERE s.monthyear_id = m.monthyear_id  
      AND s.movie_id = g.movie_id  
      AND g.genre_id = (SELECT genre_id FROM genreDIM  
                        WHERE genre_description = 'Animation')  
GROUP BY m.year;
```

YEAR	TOTAL_SALES	CUMULATIVE_TOTAL_SALES
1 2018	231	231
2 2019	440	671
3 2020	209	880
4 2021	143	1023

Level 0

```
SELECT TO_CHAR(d.sale_date, 'YYYY') AS year,  
       SUM(total_sales_revenue) AS TOTAL_SALES,  
       SUM(SUM(total_sales_revenue)) OVER  
         (ORDER BY TO_CHAR(d.sale_date, 'YYYY') ROWS UNBOUNDED PRECEDING)  
         AS CUMULATIVE_TOTAL_SALES  
FROM saleFACT_v2 s, saleDIM_2 d, movie_genreBRIDGE_2 g  
WHERE s.sale_id = d.sale_id  
      AND s.movie_id = g.movie_id  
      AND g.genre_id = (SELECT genre_id FROM genreDIM_2  
                        WHERE genre_description = 'Animation')  
GROUP BY TO_CHAR(d.sale_date, 'YYYY');
```

YEAR	TOTAL_SALES	CUMULATIVE_TOTAL_SALES
1 2018	231	231
2 2019	440	671
3 2020	209	880
4 2021	143	1023

REPORT 9

Query question:

What are the total sales revenue and cumulative total sales revenue each month in Clayton?

Explanation:

To see which month has the highest total sales revenue in Clayton and also to see the cumulative total sales revenue for each month.

Level 2

```
SELECT m.month, SUM(total_sales_revenue) AS TOTAL_SALES,  
       SUM(SUM(total_sales_revenue)) OVER (ORDER BY TO_DATE(m.month, 'Mon')  
       ROWS UNBOUNDED PRECEDING) AS CUMULATIVE_TOTAL_SALES  
FROM saleFACT_v1 s, month_yearDIM m  
WHERE s.monthyear_id = m.monthyear_id  
      AND s.location_id LIKE 'Clayton%'  
GROUP BY m.month;
```

MONTH	TOTAL_SALES	CUMULATIVE_TOTAL_SALES
1 Jan	55	55
2 Feb	143	198
3 Mar	55	253
4 Apr	55	308
5 May	110	418
6 Jun	132	550
7 Jul	176	726
8 Aug	33	759
9 Sep	121	880
10 Oct	55	935
11 Nov	154	1089
12 Dec	33	1122

Level 0

```
SELECT TO_CHAR(TO_DATE(TO_CHAR(d.sale_date, 'MM'), 'MM'), 'Mon') AS Month,  
       SUM(total_sales_revenue) AS TOTAL_SALES,  
       SUM(SUM(total_sales_revenue)) OVER (ORDER BY TO_CHAR(d.sale_date,  
       'MM') ROWS UNBOUNDED PRECEDING) AS CUMULATIVE_TOTAL_SALES  
FROM saleFACT_v2 s, saleDIM_2 d, cinemaDIM_2 c  
WHERE s.sale_id = d.sale_id  
      AND c.cinema_id = s.cinema_id  
      AND s.cinema_id = (SELECT cinema_id FROM cinemaDIM_2  
      WHERE cinema_suburb = 'Clayton')  
GROUP BY TO_CHAR(d.sale_date, 'MM')  
ORDER BY TO_CHAR(d.sale_date, 'MM');
```

MONTH	TOTAL_SALES	CUMULATIVE_TOTAL_SALES
1 Jan	55	55
2 Feb	143	198
3 Mar	55	253
4 Apr	55	308
5 May	110	418
6 Jun	132	550
7 Jul	176	726
8 Aug	33	759
9 Sep	121	880
10 Oct	55	935
11 Nov	154	1089
12 Dec	33	1122

REPORT 10

Query question:

What are the total sales revenue and moving average total sales revenue of 3 monthly in Clayton?

Explanation:

To observe the total sales revenue and moving average total sales revenue every 3 months in Clayton.

Level 2

```
SELECT m.month, SUM(total_sales_revenue) AS TOTAL_SALES,
       ROUND(AVG(SUM(total_sales_revenue)) OVER
         (ORDER BY TO_DATE(m.month, 'Mon') ROWS BETWEEN 2 PRECEDING
          AND CURRENT ROW),2) AS MOVING_TOTAL_SALES
FROM saleFACT_v1 s, month_yearDIM m
WHERE s.monthyear_id = m.monthyear_id AND s.location_id LIKE 'Clayton%'
GROUP BY m.month;
```

MONTH	TOTAL_SALES	MOVING_TOTAL_SALES
1 Jan	55	55
2 Feb	143	99
3 Mar	55	84.33
4 Apr	55	84.33
5 May	110	73.33
6 Jun	132	99
7 Jul	176	139.33
8 Aug	33	113.67
9 Sep	121	110
10 Oct	55	69.67
11 Nov	154	110
12 Dec	33	80.67

Level 0

```
SELECT TO_CHAR(TO_DATE(TO_CHAR(d.sale_date, 'MM'), 'MM'),'Mon') AS Month,
       SUM(total_sales_revenue) AS TOTAL_SALES,
       ROUND(AVG(SUM(total_sales_revenue)) OVER (ORDER BY
         TO_CHAR(d.sale_date, 'MM') ROWS BETWEEN 2 PRECEDING
          AND CURRENT ROW),2) AS MOVING_TOTAL_SALES
FROM saleFACT_v2 s, saleDIM_2 d, cinemaDIM_2 c
WHERE s.sale_id = d.sale_id
      AND c.cinema_id = s.cinema_id
      AND s.cinema_id = (SELECT cinema_id FROM cinemaDIM_2
        WHERE cinema_suburb = 'Clayton')
GROUP BY TO_CHAR(d.sale_date, 'MM');
```

MONTH	TOTAL_SALES	MOVING_TOTAL_SALES
1 Jan	55	55
2 Feb	143	99
3 Mar	55	84.33
4 Apr	55	84.33
5 May	110	73.33
6 Jun	132	99
7 Jul	176	139.33
8 Aug	33	113.67
9 Sep	121	110
10 Oct	55	69.67
11 Nov	154	110
12 Dec	33	80.67

d) Reports with Partitions

REPORT 11 (PARTITION)

Level 2

```
SELECT a.month, a.mode_id, b.genre_id, c.NUMBER_OF_TICKETS_SOLD,
       a.BOOKINGMODE_RANK_BY_MONTH, b.GENRE_RANK_BY_MONTH
FROM
  (SELECT t.month AS Month, s.mode_id, SUM(s.number_of_tickets_sold) AS
    NUMBER_OF_TICKETS_SOLD, RANK() OVER (PARTITION BY t.month ORDER BY
    SUM(s.number_of_tickets_sold) DESC) AS BOOKINGMODE_RANK_BY_MONTH
  FROM saleFACT_v1 s, month_yearDIM t
  WHERE s.monthyear_id = t.monthyear_id
  GROUP BY t.month, s.mode_id
  ORDER BY t.month, s.mode_id) a,

  (SELECT t.month AS Month, g.genre_id, SUM(s.number_of_tickets_sold) AS
    NUMBER_OF_TICKETS_SOLD, RANK() OVER (PARTITION BY t.month ORDER BY
    SUM(s.number_of_tickets_sold) DESC) AS GENRE_RANK_BY_MONTH
  FROM saleFACT_v1 s, month_yearDIM t, movie_genreBRIDGE g
  WHERE s.monthyear_id = t.monthyear_id
  AND s.movie_id = g.movie_id
  GROUP BY t.month, g.genre_id
  ORDER BY t.month, g.genre_id) b,

  (SELECT t.month AS month, s.mode_id, g.genre_id,
    SUM(s.number_of_tickets_sold) AS NUMBER_OF_TICKETS_SOLD
  FROM saleFACT_v1 s, month_yearDIM t, movie_genreBRIDGE g
  WHERE s.monthyear_id = t.monthyear_id
  AND s.movie_id = g.movie_id
  GROUP BY t.month, g.genre_id, s.mode_id
  ORDER BY t.month, s.mode_id, g.genre_id) c

WHERE a.month = b.month
      AND a.month = c.month
      AND a.mode_id = c.mode_id
      AND b.genre_id = c.genre_id
ORDER BY TO_DATE(month, 'Mon'), BOOKINGMODE_RANK_BY_MONTH,
        GENRE_RANK_BY_MONTH;
```

MONTH	MODE_ID	GENRE_ID	NUMBER_OF_TICKETS_SOLD	BOOKINGMODE_RANK_BY_MONTH	GENRE_RANK_BY_MONTH
1 Jan	1	18	63	1	1
2 Jan	1	35	45	1	2
3 Jan	1	53	20	1	3
4 Jan	1	28	11	1	4
5 Jan	1	10749	36	1	5
6 Jan	1	12	17	1	6
7 Jan	1	80	12	1	7
8 Jan	1	878	11	1	8
9 Jan	1	9648	8	1	9
10 Jan	1	27	5	1	10
11 Jan	1	10751	3	1	11
12 Jan	1	14	8	1	12
13 Jan	1	36	5	1	13
14 Jan	1	37	6	1	14
15 Jan	1	10752	1	1	15
16 Jan	1	99	2	1	17
17 Jan	1	10770	1	1	17

Level 0

```
SELECT a.month, a.mode_id, b.genre_id, c.NUMBER_OF_TICKETS_SOLD,
       a.BOOKINGMODE_RANK_BY_MONTH, b.GENRE_RANK_BY_MONTH
FROM
  (SELECT TO_CHAR(d.sale_date, 'Mon') AS Month, s.mode_id,
          SUM(s.number_of_tickets_sold) AS NUMBER_OF_TICKETS_SOLD,
          RANK() OVER (PARTITION BY TO_CHAR(d.sale_date, 'Mon') ORDER BY
                        SUM(s.number_of_tickets_sold) DESC) AS BOOKINGMODE_RANK_BY_MONTH
  FROM saleFACT_v2 s, saleDIM_2 d
  WHERE s.sale_id = d.sale_id
  GROUP BY TO_CHAR(d.sale_date, 'Mon'), s.mode_id
  ORDER BY TO_CHAR(d.sale_date, 'Mon'), s.mode_id) a,

  (SELECT TO_CHAR(d.sale_date, 'Mon') AS Month, g.genre_id,
          SUM(s.number_of_tickets_sold) AS NUMBER_OF_TICKETS_SOLD,
          RANK() OVER (PARTITION BY TO_CHAR(d.sale_date, 'Mon') ORDER BY
                        SUM(s.number_of_tickets_sold) DESC) AS GENRE_RANK_BY_MONTH
  FROM saleFACT_v2 s, saleDIM_2 d, movie_genreBRIDGE_2 g
  WHERE s.sale_id = d.sale_id
  AND s.movie_id = g.movie_id
  GROUP BY TO_CHAR(d.sale_date, 'Mon'), g.genre_id
  ORDER BY TO_CHAR(d.sale_date, 'Mon'), g.genre_id) b,

  (SELECT TO_CHAR(d.sale_date, 'Mon') AS month, s.mode_id, g.genre_id,
          SUM(s.number_of_tickets_sold) AS NUMBER_OF_TICKETS_SOLD
  FROM saleFACT_v2 s, saleDIM_2 d, movie_genreBRIDGE_2 g
  WHERE s.sale_id = d.sale_id
  AND s.movie_id = g.movie_id
  GROUP BY TO_CHAR(d.sale_date, 'Mon'), g.genre_id, s.mode_id
  ORDER BY TO_CHAR(d.sale_date, 'Mon'), s.mode_id, g.genre_id) c

WHERE a.month = b.month
      AND a.month = c.month
      AND a.mode_id = c.mode_id
      AND b.genre_id = c.genre_id
ORDER BY TO_DATE(month, 'Mon'), BOOKINGMODE_RANK_BY_MONTH,
         GENRE_RANK_BY_MONTH;
```

MONTH	MODE_ID	GENRE_ID	NUMBER_OF_TICKETS_SOLD	BOOKINGMODE_RANK_BY_MONTH	GENRE_RANK_BY_MONTH
1 Jan	1	18	63	1	1
2 Jan	1	35	45	1	2
3 Jan	1	53	20	1	3
4 Jan	1	28	11	1	4
5 Jan	1	10749	36	1	5
6 Jan	1	12	17	1	6
7 Jan	1	80	12	1	7
8 Jan	1	878	11	1	8
9 Jan	1	9648	8	1	9
10 Jan	1	27	5	1	10
11 Jan	1	10751	3	1	11
12 Jan	1	14	8	1	12
13 Jan	1	36	5	1	13
14 Jan	1	37	6	1	14
15 Jan	1	10752	1	1	15
16 Jan	1	99	2	1	17
17 Jan	1	10770	1	1	17

REPORT 12 (PARTITION)

Query question:

Show ranking of each movie runtime based on the yearly total sales revenue and the ranking of yearly total sales revenue by each movie runtime.

Explanation:

Management will be able to know the runtime category preference of the customers and the yearly total sales revenue by each movie runtime.

Level 2

```
SELECT t.year AS Year, s.runtime_id, SUM(s.total_sales_revenue)
      AS TOTAL_SALES_REVENUE, RANK() OVER (PARTITION BY t.Year ORDER BY
      SUM(s.total_sales_revenue) DESC) AS RANK_BY_YEAR,
      RANK() OVER (PARTITION BY s.runtime_id ORDER BY
      SUM(s.total_sales_revenue) DESC) AS RANK_BY_RUNTIME
FROM saleFACT_v1 s, month_yearDIM t
WHERE s.monthyear_id = t.monthyear_id
GROUP BY t.year, s.runtime_id;
```

YEAR	RUNTIME_ID	TOTAL_SALES_REVENUE	RANK_BY_YEAR	RANK_BY_RUNTIME
1 2018	Long	7425	1	3
2 2018	Medium	5071	2	3
3 2018	Short	66	3	3
4 2019	Long	8096	1	2
5 2019	Medium	5346	2	1
6 2019	Short	253	3	1
7 2020	Long	8448	1	1
8 2020	Medium	5236	2	2
9 2020	Short	176	3	2
10 2021	Long	4235	1	4
11 2021	Medium	2607	2	4
12 2021	Short	11	3	4

Level 0

```
SELECT TO_CHAR(t.sale_date, 'YYYY') AS Year, m.runtime_id,
       SUM(s.total_sales_revenue) AS TOTAL_SALES_REVENUE,
       RANK() OVER (PARTITION BY TO_CHAR(t.sale_date, 'YYYY') ORDER BY
                    SUM(s.total_sales_revenue) DESC) AS RANK_BY_YEAR,
       RANK() OVER (PARTITION BY m.runtime_id ORDER BY
                    SUM(s.total_sales_revenue) DESC) AS RANK_BY_RUNTIME
FROM saleFACT_v2 s, saleDIM_2 t,
     (SELECT DISTINCT movie_id,
          CASE
              WHEN movie_runtime < 50 THEN 'Short'
              WHEN movie_runtime >= 50 AND movie_runtime <= 100 THEN 'Medium'
              ELSE 'Long'
          END AS runtime_id
     FROM movieDIM_2) m
WHERE s.movie_id = m.movie_id
      AND s.sale_id = t.sale_id
GROUP BY TO_CHAR(t.sale_date, 'YYYY'), m.runtime_id;
```

YEAR	RUNTIME_ID	TOTAL_SALES_REVENUE	RANK_BY_YEAR	RANK_BY_RUNTIME
1 2018	Long	7425	1	3
2 2018	Medium	5071	2	3
3 2018	Short	66	3	3
4 2019	Long	8096	1	2
5 2019	Medium	5346	2	1
6 2019	Short	253	3	1
7 2020	Long	8448	1	1
8 2020	Medium	5236	2	2
9 2020	Short	176	3	2
10 2021	Long	4235	1	4
11 2021	Medium	2607	2	4
12 2021	Short	11	3	4

Task C.4

Business Intelligence Reports

