```sql
--Task 1: Financial Performance
-- Write a query to list all films with the following information:
--• Exclude films with missing or zero budgets
 --• Sort results by ROI in descending order
SELECT
    PF.film ,
    YEAR(PF.release_date) AS 'Release year',
    CONCAT(BO.budget / 1000000.0 , 'M') AS [Budget],
    CONCAT(BO.box_office_worldwide / 1000000.0, 'M') AS [Worldwide gross],
    CAST(((BO.box_office_worldwide - BO.budget) * 100.0) / BO.budget AS DECIMAL(5,1)) AS [ROI percentage]
FROM
      ['pixar_films $'] PF
JOIN
      ['box_office $'] BO
ON
   PF.film = BO.film
WHERE
    BO.budget  IS NOT NULL AND BO.budget <> 0
ORDER BY [ROI percentage] DESC;
```

88 %

Results | Messages

	film	Release year	Budget	Worldwide gross	ROI percentage
11	Coco	2017	175M	815M	365.7
12	Monsters, Inc.	2001	115M	529M	360.0
13	Up	2009	175M	735M	320.0
14	Ratatouille	2007	150M	624M	316.0
15	Cars	2006	120M	462M	285.0
16	Monsters University	2013	200M	744M	272.0
17	A Bug's Life	1998	120M	363M	202.5
18	Brave	2012	185M	539M	191.4
19	WALL-E	2008	180M	521M	189.4
20	Cars 2	2011	200M	560M	180.0
21	Elemental	2023	200M	496M	148.0
22	Cars 3	2017	175M	384M	119.4
23	The Good Dinosaur	2015	175M	332M	89.7
24	Lightyear	2022	200M	226M	13.0
25	Soul	2020	150M	122M	-18.7
26	Onward	2020	175M	142M	-18.9
27	Turning Red	2022	175M	21.8134M	-87.5

```sql
------------------------------------------------------------
--Task 2: Award Analysis (1.5 Marks)
SELECT
   WN.film,
   SUM(CASE WHEN status ='WON' THEN 1 ELSE 0 END) AS 'AWARDS WIN',
   COUNT (*) AS 'Total nominations' ,
   (SUM(CASE WHEN status ='WON' THEN 1 ELSE 0 END)*100/ COUNT(*)) AS [WIN PRECENT]
FROM
   ( SELECT *
        FROM
           Movies_Rewards$
        WHERE
            status = 'WON' OR
            status = 'Nominated') WN
GROUP BY
   WN.film
HAVING
   SUM(CASE WHEN status ='WON' THEN 1 ELSE 0 END) > 0
ORDER BY [WIN PRECENT] DESC
```

96 %

Results | Messages

	film	AWARDS WIN	Total nominations	WIN PRECENT
1	Brave	1	1	100
2	Coco	2	2	100
3	Inside Out	1	2	50
4	Soul	2	4	50
5	The Incredibles	2	4	50
6	Toy Story 4	1	2	50
7	Up	2	5	40
8	Toy Story 3	2	5	40
9	Monsters, Inc.	1	4	25
10	Finding Nemo	1	4	25
11	Ratatouille	1	5	20
12	WALL-E	1	6	16

```sql
--*******************************************
--Task 3: Genre Profitability (2.0 Marks)

SELECT TOP 5
    MT.value,
    CAST(AVG(BO.box_office_worldwide)/1000000.0 AS DECIMAL(10,1)) AS [Average Worldwide Gross (M)],
    COUNT(*) AS 'Number of films'
FROM Movie_Type$ MT
JOIN
['box_office $'] BO
ON
MT.film = BO.film
GROUP BY MT.value
HAVING COUNT(*)  >3
ORDER BY  [Average Worldwide Gross (M)] DESC

-- Task 4: Director Impact Study (1.5 Marks)
-- For directors who have worked on two or more films, provide:
```

87 %

Results | Messages

	value	Average Worldwide Gross (M)	Number of films
1	Coming-of-Age	770.8	5
2	Urban Adventure	742.6	8
3	Quest	687.7	6
4	Supernatural Fantasy	647.3	7
5	Animal Adventure	644.0	5

```sql
-- Task 4: Director Impact Study (1.5 Marks)
SELECT
    PP.name ,
    COUNT(PP.film) AS 'MOVIE COUNT',
    CAST(AVG(PR.rotten_tomatoes_score) AS DECIMAL(5,1)) AS [Average Rotten Tomatoes Score],
    CAST(AVG(BO.box_office_worldwide) / 1000000.0 AS DECIMAL(10,1)) AS [Average Worldwide Gross (M)],
    CAST(AVG(PR.imdb_score) AS DECIMAL(5,1)) AS [Average IMDb score Sort the results ]
FROM
    pixar_people$ PP
JOIN
    ['public_response $'] PR
ON
    PR.film  = PP.film
JOIN
    ['box_office $'] BO
ON BO.film = PP.film
WHERE
    PP.role_type = 'Director'
GROUP BY PP.name
HAVING
    COUNT(DISTINCT PP.film) >= 2
ORDER BY AVG(BO.box_office_worldwide) DESC
```

88 %

Results | Messages

	name	MOVIE COUNT	Average Rotten Tomatoes Score	Average Worldwide Gross (M)	Average IMDb score Sort the results
1	Lee Unkrich	2	1.0	942.5	8.4
2	Brad Bird	3	1.0	831.7	7.9
3	Andrew Stanton	3	1.0	807.3	7.9
4	Pete Docter	4	1.0	561.0	8.1
5	John Lasseter	5	0.8	458.0	7.4
6	Dan Scanlon	2	0.8	443.0	7.3
7	Peter Sohn	2	0.7	414.0	6.8

```sql
--Task 5: Franchise Comparison
SELECT
    CASE
        WHEN MOVIES.film  LIKE 'Toy%' THEN  'Toy Story'
        WHEN MOVIES.film  LIKE 'Car%' THEN  'Car'
        WHEN MOVIES.film  LIKE 'Finding Dory%' THEN 'Finding Dory'
        WHEN MOVIES.film  LIKE 'Finding Nemo%' THEN 'Finding Nemo'
    END  AS  'Franchise name' ,
        ROUND(SUM(BO.box_office_worldwide) /1000000 ,2)AS 'Total worldwide gross (in millions)',
        COUNT(*) AS total_films,
        AVG(PF.run_time) AS 'Average runtime (in minutes)'
FROM
    (
    SELECT
        BO.film
    FROM
        ['box_office $'] BO
    WHERE
        BO.film LIKE 'Toy%' OR
        BO.film LIKE 'Car%' OR
        BO.film LIKE 'Finding Dory%' OR
        BO.film LIKE 'Finding Nemo%'
    )
        AS MOVIES
    JOIN
            ['box_office $'] BO
    ON
            MOVIES.film = BO.film
    JOIN
            ['pixar_films $'] PF
    ON      PF.film = MOVIES.film
    GROUP BY
        CASE
            WHEN MOVIES.film  LIKE 'Toy%' THEN  'Toy Story'
            WHEN MOVIES.film  LIKE 'Car%' THEN  'Car'
            WHEN MOVIES.film  LIKE 'Finding Dory%' THEN 'Finding Dory'
            WHEN MOVIES.film  LIKE 'Finding Nemo%' THEN 'Finding Nemo'
        END
    ORDER BY [Total worldwide gross (in millions)] DESC
```

55 %

Results | Messages

	Franchise name	Total worldwide gross (in millions)	total_films	Average runtime (in minutes)
1	Toy Story	3045	4	94
2	Car	1406	3	108
3	Finding Dory	1030	1	97
4	Finding Nemo	871	1	100

```sql
--Task 6: Budget Category Analysis
SELECT
   COUNT(*) AS film_count      ,
   ROUND(AVG(PR.Metacritic_Score) *100,2) AS ' Average Metacritic score' ,
   ROUND(AVG(BO.box_office_worldwide/1000000),2) AS 'Average worldwide gross (in millions)',

   CASE
       WHEN BO.budget < 100000000 THEN 'LOW'
       WHEN BO.budget >=100000000 AND BO.budget <= 150000000 THEN 'MEDIUM'
       WHEN BO.budget > 150000000 THEN 'HIGH'
   END AS budget_categories
FROM
       ['box_office $'] BO
   JOIN
       ['public_response $'] PR
   ON
       BO.film = PR.film

GROUP BY
   CASE
       WHEN BO.budget <  100000000  THEN 'LOW'
       WHEN BO.budget >= 100000000 AND BO.budget <=150000000 THEN 'MEDIUM'
       WHEN BO.budget >  150000000  THEN 'HIGH'
   END
```

87 %

Results | Messages

	film_count	Average Metacritic score	Average worldwide gross (in millions)	budget_categories
1	18	74.72	693.55	HIGH
2	4	90.75	601.75	LOW
3	5	81.8	420	MEDIUM