

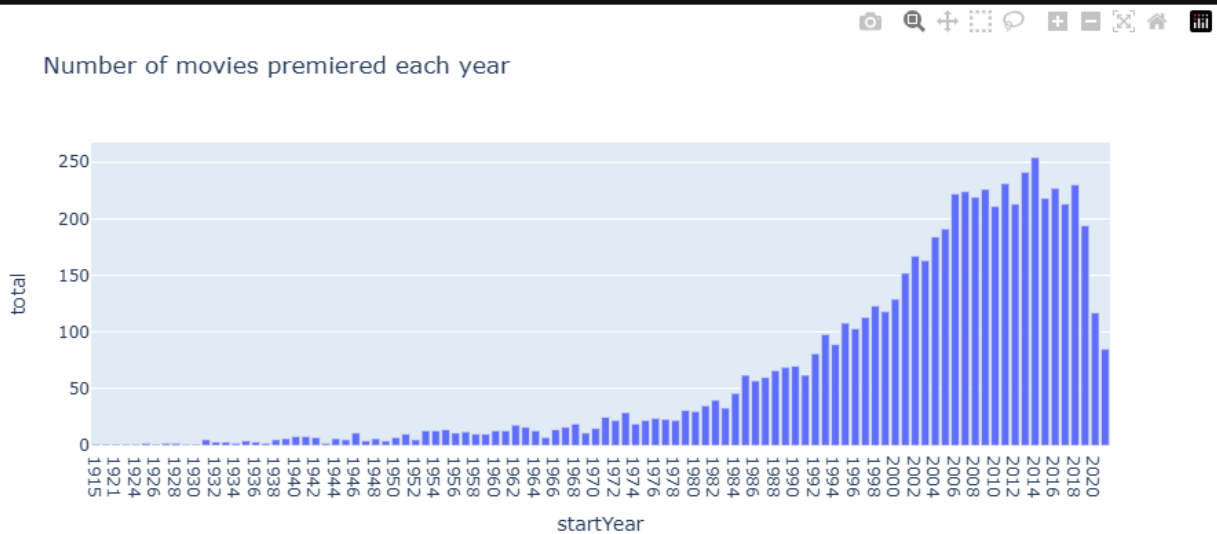
```
•[70]: %%sql --save query_example
```

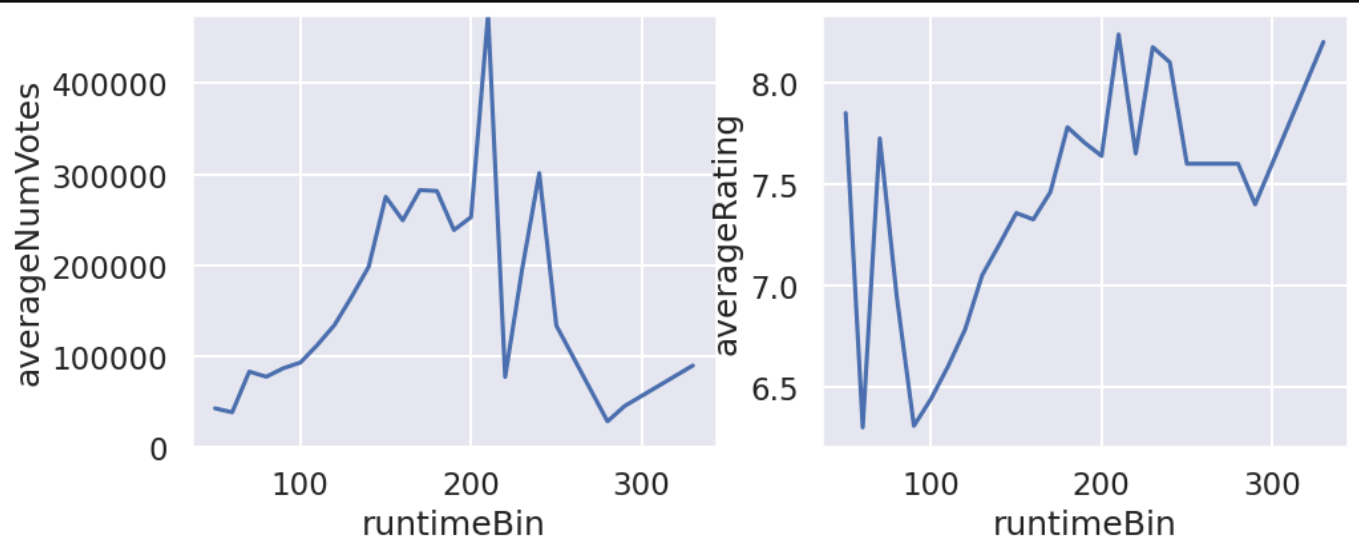
```
SELECT *  
FROM name  
LIMIT 15;
```

Running query in 'duckdb:///data/imdbmini.db'

[70]:	nconst	primaryName	birthYear	deathYear	primaryProfession
	1	Fred Astaire	1899	1987	soundtrack,actor,miscellaneous
	2	Lauren Bacall	1924	2014	actress,soundtrack
	3	Brigitte Bardot	1934	None	actress,soundtrack,music_department
	4	John Belushi	1949	1982	actor,soundtrack,writer
	5	Ingmar Bergman	1918	2007	writer,director,actor
	6	Ingrid Bergman	1915	1982	actress,soundtrack,producer
	7	Humphrey Bogart	1899	1957	actor,soundtrack,producer
	8	Marlon Brando	1924	2004	actor,soundtrack,director
	9	Richard Burton	1925	1984	actor,soundtrack,producer
	10	James Cagney	1899	1986	actor,soundtrack,director
	11	Gary Cooper	1901	1961	actor,soundtrack,producer
	12	Bette Davis	1908	1989	actress,soundtrack,make_up_department
	13	Doris Day	1922	2019	soundtrack,actress,producer
	14	Olivia de Havilland	1916	2020	actress,soundtrack
	15	James Dean	1931	1955	actor,miscellaneous

```
# Run this call to generate the bar plot; no further action is needed  
px.bar(res_q1b, x="startYear", y="total",  
       title="Number of movies premiered each year",  
       width=900, height=400)
```





```
%%sql --save query_q2
```

```
SELECT Name.primaryName AS name, COUNT(*) AS total
FROM ROLE
LEFT JOIN NAME on Name.nconst = Role.nconst
LEFT JOIN TITLE on Role.tconst = Title.tconst
WHERE Title.titleType = 'movie'
      AND Role.category LIKE 'act%'
GROUP BY Name.primaryName
ORDER BY total DESC, primaryName
LIMIT 10;
```

Running query in 'duckdb:///data/imdbmini.db'

name	total
Robert De Niro	64
Samuel L. Jackson	54
Nicolas Cage	53
Bruce Willis	49
Tom Hanks	46
Johnny Depp	43
Mark Wahlberg	41
Liam Neeson	40
Morgan Freeman	40
Adam Sandler	39

```

%%sql --save query_q5
SELECT primaryName AS name, SUM(CAST(averageRating AS float) * CAST(numVotes AS int))/SUM(CAST(numVotes AS int)) AS actorRating
FROM Name
JOIN Role on Role.nconst = Name.nconst
JOIN Rating on Role.tconst = Rating.tconst
JOIN Title on Title.tconst = Rating.tconst
WHERE CAST(numVotes AS int) > 22000 AND Title.titleType = 'movie' AND Role.category LIKE '%act%'
GROUP BY primaryName
HAVING COUNT(*) >= 20
ORDER BY actorRating DESC
LIMIT 10;

```

Running query in 'duckdb:///data/imdbmini.db'

name	actorRating
Diane Keaton	8.459749306914137
Al Pacino	8.143296547996608
Michael Caine	8.140341554923678
Leonardo DiCaprio	8.03496089527195
Christian Bale	7.989825259062718
Kevin Spacey	7.9348882127969205
Jack Nicholson	7.933033463121602
Clint Eastwood	7.866838620708466
Edward Norton	7.856717980284558
Tom Hanks	7.847982328558773

$$\text{actorRating} = \frac{\sum_m (\text{averageRating}[m] * \text{numVotes}[m])}{\sum_m \text{numVotes}[m]}$$