

SUBQUERIES AND JOIN

Senaryo: Kitap sayfası sayısı ortalama kitap sayfası sayısından fazla olan kitap isimlerini, yazar isim ve soy isimleri ile birlikte sıralayalım.

```
SELECT author.first_name, author.last_name, book.title, book.page_number FROM author
INNER JOIN book ON book.author_id = author.id
WHERE page_number >
(
SELECT AVG(page_number) FROM book
);
```

Senaryo: Film uzunluğu en fazla olan filmlerin isimlerini, uzunluğunu aktör isim ve soy isimleri ile birlikte yazdıralım.

```
SELECT film.title, film.length, actor.first_name, actor.last_name FROM actor
JOIN film_actor ON film_actor.actor_id = actor.actor_id
JOIN film ON film_actor.film_id = film.film_id
WHERE film.length =
(
SELECT MAX(length) FROM film
);
```

Senaryo: İsminde en az 4 tane e veya E bulunan kaç film vardır?

```
SELECT COUNT (*) FROM film
WHERE title ~~~* '%e%e%e%e%';
```

Senaryo: Kategori isimlerini ve kategori başına düşen film sayısını yazınız?

```
SELECT COUNT (*), category.name FROM category
JOIN film_category ON film_category.category_id = category.category_id
JOIN film ON film.film_id = category.category_id
GROUP BY category.name;
```

Senaryo: En çok film bulunan rating kategorisi hangisidir?

```
SELECT COUNT (*), rating FROM film
GROUP BY rating
ORDER BY COUNT (*) DESC
LIMIT 1;
```

Senaryo: Film tablosunda 'K' karakteri ile başlayan en uzun ve replacement_cost en az olan 3 film sıralayınız?

```
SELECT title, length, replacement_cost FROM film
WHERE title ~~~ 'K%'
ORDER BY length DESC, replacement_cost
LIMIT 3;
```

Senaryo: En çok alışveriş yapan müşterinin adı nedir?

```
SELECT SUM(amount), customer.first_name, customer.last_name FROM payment
JOIN customer ON customer.customer_id = payment.customer_id
GROUP BY payment.customer_id, customer.first_name, customer.last_name
ORDER BY SUM(amount) DESC
LIMIT 1;
```