Objective

 ANSI-99 SQL birleştirme sözdizimi ile **natural join** (doğal birleştirme) sorgusu oluşturup çalıştırmak

 ANSI-99 SQL ile **cross join** oluşturmak

 SQL için ANSI tarafından tanımlanmış bir standarda sahip olmanın önemini açıklamak

 Bir işletmenin, birden fazla veri kaynağından bilgi birleştirmesi gerektiği durumu açıklamak

Vocabulary

1. Rastgele bir koşula veya farklı isimlere sahip iki sütuna göre doğal birleştirme yapılmasına izin verir.  
   → JOIN ... ON
2. Belirtilen bir sütun adına göre eşitlik (equijoin) temelli birleştirme yapar.  
   → JOIN ... USING

**Try It / Solve It**

1.locations ve departments tablolarını location\_id sütunu ile birleştir. Sonuçları sadece **location\_id = 1400** olanlarla sınırla.

SELECT d.department\_id,d.department\_name, l.location\_id, l.city

FROM departments d

JOIN locations l ON d.location\_id = l.location\_id

WHERE l.location\_id = 1400;

2.d\_play\_list\_items, d\_track\_listings ve d\_cds tablolarını JOIN USING sözdizimi ile birleştir. Çıktıda: **song ID**, **CD number**, **title**, **comments** gösterilsin.

SELECT song\_id, cd\_number,title, comments

FROM d\_play\_list\_items

JOIN d\_track\_listings USING (song\_id)

JOIN d\_cds USING (cd\_number);

1. city = 'Seattle' olan şehirdeki **department\_id**'si 10, 20 veya 30 olan departmanların: **city**, **department\_name**, **location\_id**, **department\_id** bilgilerini göster.

SELECT l.city, d.department\_name, d.location\_id, d.department\_id

FROM departments d

JOIN locations l ON d.location\_id = l.location\_id

WHERE l.city = 'Seattle' AND d.department\_id IN (10, 20, 30);

1. "Americas" bölgesine ait **country\_name**, **region\_id** ve **region\_name**'i göster.

SELECT c.country\_name, r.region\_id, r.region\_name

FROM countries c

JOIN regions r ON c.region\_id = r.region\_id

WHERE r.region\_name = 'Americas';

1. employees ve jobs tablolarını birleştir. **first\_name**, **last\_name**, **hire\_date**, **job\_id**, **job\_title**, **max\_salary** göster.  
   Maaş üst limiti $12,000'dan fazla olan işleri alan çalışanları göster.

SELECT e.first\_name, e.last\_name, e.hire\_date, e.job\_id, j.job\_title, j.max\_salary

FROM employees e

JOIN jobs j ON e.job\_id = j.job\_id

WHERE j.max\_salary > 12000;

1. Tüm "stock clerk" çalışanların **job\_title**, **first\_name**, **last\_name**, **email** bilgilerini göster.

SELECT j.job\_title, e.first\_name, e.last\_name, e.email

FROM employees e

JOIN jobs j ON e.job\_id = j.job\_id

WHERE j.job\_title = 'Stock Clerk';

1. employees tablosunda employee\_id, first\_name, last\_name, manager\_id, manager first\_name, manager last\_name bilgilerini göster.

SELECT e.employee\_id, e.first\_name, e.last\_name, e.manager\_id, m.first\_name AS manager\_first\_name,

m.last\_name AS manager\_last\_name

FROM employees e

JOIN employees m ON e.manager\_id = m.employee\_id;

1. Tüm Kanada’daki konumlar için location\_id, city, department\_name bilgilerini göster.

SELECT l.location\_id, l.city, d.department\_name

FROM departments d

JOIN locations l ON d.location\_id = l.location\_id

WHERE l.country\_id = 'CA';

1. Departman ID’si 80, 90, 110, 190 olan çalışanlar için: manager\_id, department\_id, department\_name, first\_name, last\_name bilgilerini göster.

SELECT e.manager\_id, d.department\_id,d.department\_name, e.first\_name, e.last\_name

FROM employees e

JOIN departments d ON e.department\_id = d.department\_id

WHERE d.department\_id IN (80, 90, 110, 190);

1. İşe alım tarihi 7 Haziran 1994 olan çalışanlar için: employee\_id, last\_name, department\_id, department\_name, hire\_date bilgilerini göster.

SELECT e.employee\_id, e.last\_name, e.department\_id, d.department\_name, e.hire\_date

FROM employees e

JOIN departments d ON e.department\_id = d.department\_id

WHERE e.hire\_date = TO\_DATE('07-06-1994', 'DD-MM-YYYY');