

# Data Analytics

## SQL Server Overview-basic



Building a better  
working world

- ▶ היכרות ראשונית
- ▶ סוגי נתונים בטבלה
- ▶ מפתח ראשי ומפתח זר
- ▶ סוגי קשרים בין טבלאות
- ▶ פקודות בסיסיות
- ▶ פעולות בסיסיות
- ▶ פעולות אריתמטיות מסכמות
- ▶ מניפולציות טקסטואליות
- ▶ התניות ברמת ה-SELECT
- ▶ חיבור בין טבלאות

- ▶ מכון תקנים האמריקאי (ANSI) הגדיר את SQL כשפה הסטנדרטית לעבודה מול מסדי נתונים טבלאיים (RDBMS).
- ▶ שפת תכנות המאפשרת ניהול תוכן במסדי נתונים טבלאיים.
- ▶ שפת תכנות המאפשרת שליפת מידע ממסדי נתונים טבלאיים, לצורך ניתוח ותחקור התוצאות.



**ORACLE®**  
**PL SQL**

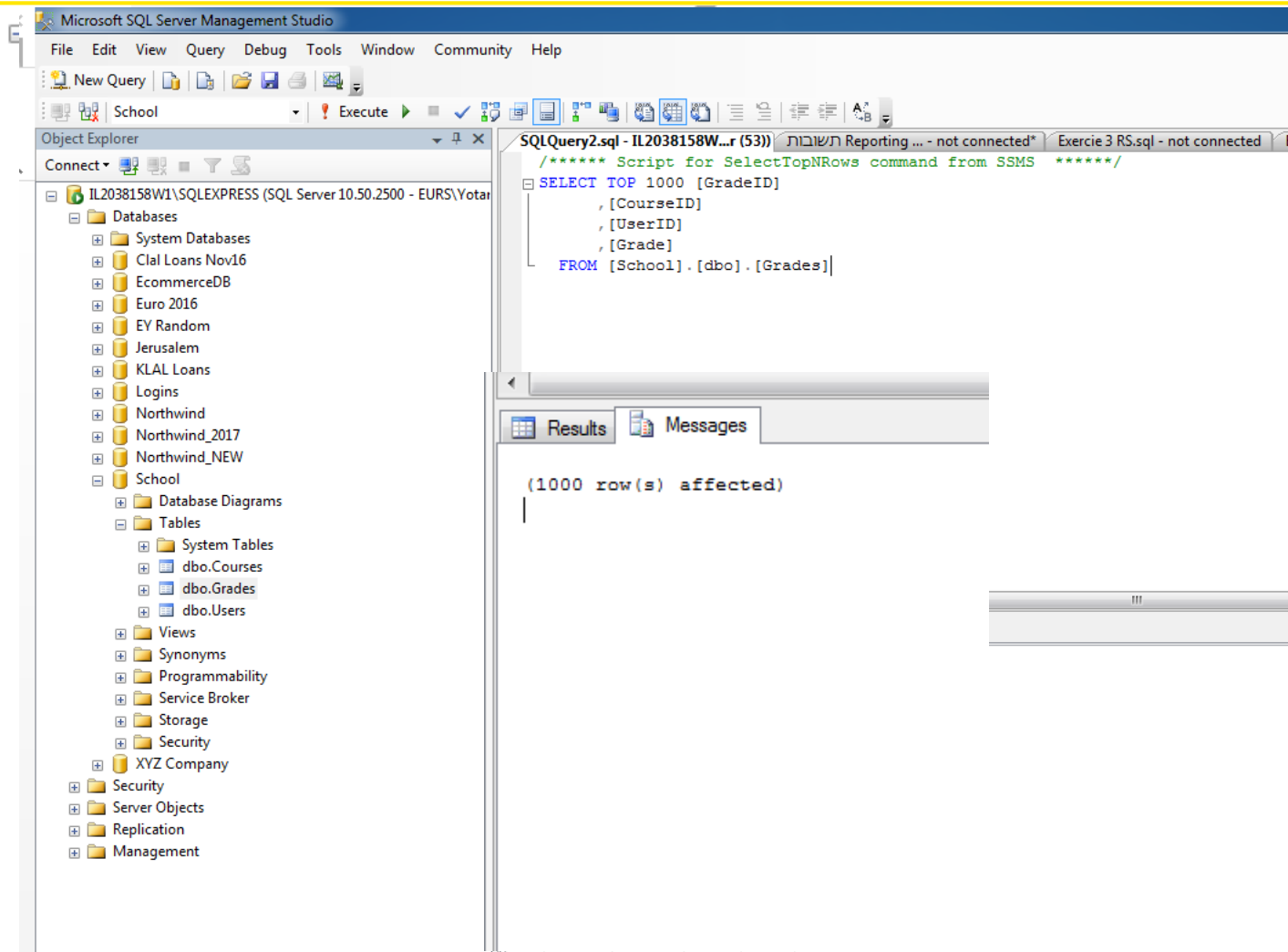


Microsoft®  
**SQL Server®**



<https://vimeo.com/24688457>

# Microsoft SQL Management Studio

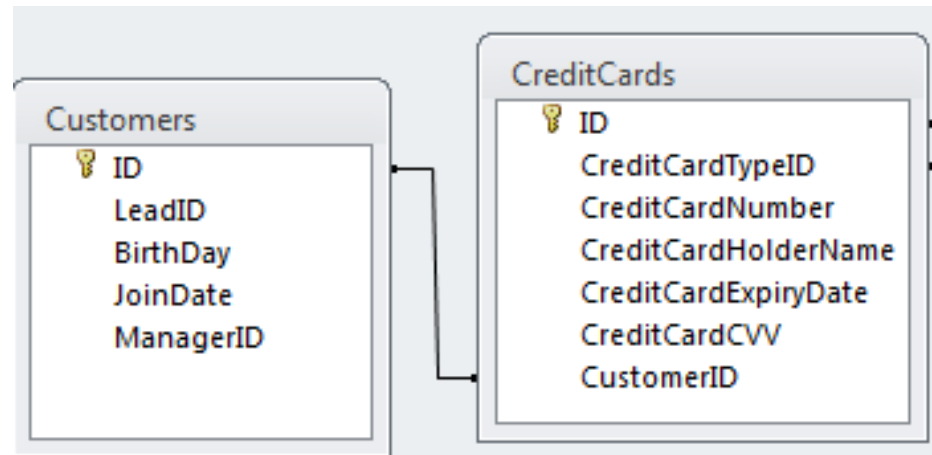


# סוגי נתונים בטבלה

---

- ▶ **CHAR(N)** - מחרוזת של תווים אלפא-נומריים באורך N.
- ▶ **VARCHAR(N)** - מחרוזת של תווים אלפא-נומריים בעלת אורך משתנה, כאשר N הוא האורך המקסימלי.
- ▶ **NUMBER(N)** - מספר, שלילי, חיובי, אפס, שלם או עשרוני.
- ▶ **DATE** - תאריך במבנה מוסכם כלשהו (למשל dd/mm/yyyy).
- ▶ **INTEGER** - מספר שלם בלבד (עד 32,000).
- ▶ **TIME** - משתנה היכול להכיל רק ערכים חוקיים של שעה ביממה במבנה מוסכם כלשהו (למשל hh:mm:ss).
- ▶ **DATETIME** - משתנה המכיל תאריך ושעה גם יחד.
- ▶ **BOOLEAN** - משתנה היכול להכיל ערך בינארי T/F.

# מפתח ראשי ומפתח זר



מפתח ראשי – Customers.ID ►

מפתח זר – CreditCards.CustomerID ►

# סוגי קשרים בין טבלאות

---



- ▶ **1:N** – בטבלה הראשונה כל מפתח מופיע פעם אחת בלבד (מפתח ראשי) ובטבלה השנייה מספר רב של פעמים (מפתח זר).
- ▶ **1:1** – בטבלה הראשונה כל מפתח מופיע פעם אחת בלבד (מפתח ראשי) וכך גם בטבלה השנייה.
- ▶ **N:N** – בטבלה הראשונה ובשנייה כל מפתח מופיע מספר רב של פעמים (מפתח זר).

**SELECT** <Fields, Actions>  
**FROM** <Table>  
**WHERE** <Conditions> מוריד שורות  
**GROUP BY** <Columns>  
**HAVING** <Conditions on Actions>  
**ORDER BY** <Columns>




# פקודות בסיסיות - SELECT

---

- ▶ בחירת השדות מהטבלה הנבחרת עבור תצוגה: 
- ▶ ניתן לבחור שם נוסף (Alias) עבור שדות נבחרים. 
- ▶ שימוש ב-\* יחזיר את כל השדות בטבלה.
- ▶ שימוש ב-Top X כדי להציג בשליפה רק את X השורות העליונות.
- ▶ ניתן לבצע פעולות בין שדות שונים – נלמד בהמשך.
- ▶ ניתן לבצע פעולות אריתמטיות על שדות שונים – נלמד בהמשך.
- ▶ ניתן לבצע מניפולציות טקסט על שדות שונים – נלמד בהמשך.

# פקודות בסיסיות - FROM

---

- ▶ בחירת שם הטבלה אשר ממנה נשלוף את הנתונים.
- ▶ ניתן לבחור שם נוסף (Alias) עבור הטבלה הנבחרת. 
- ▶ ניתן לשלוף ממספר טבלאות באותה שאילתא – נלמד בהמשך.

# פקודות בסיסיות - WHERE

---

- ▶ קביעת התנאים אותם השליפה צריכה לקיים.
- ▶ התנאים יכולים להיות עבור פעולות אריתמטיות בסיסיות או מניפולציות טקסטואליות – נלמד בהמשך.
- ▶ לא ניתן להכליל תנאים אשר כוללים פעולות אריתמטיות מסכמות.

# פקודות בסיסיות – ORDER BY

---

- ▶ קביעת סדר ההצגה של השורות לפי השדה – מהקטן לגדול (או להיפך), מ-A ל-Z (או להיפך).
- ▶ ניתן לרשום את מספר העמודה כפי שהיא מוצגת במסגרת השליפה במקום את שם העמודה.
- ▶ ASC – סדר עולה.
- ▶ DESC – סדר יורד.

# פקודות בסיסיות שיילמדו בהמשך

---

GROUP BY ▶

HAVING ▶

משמעות	אופרטור
חיבור, חיסור, כפל וחילוק	+, -, *, /
שווה	=
לא שווה	!=, <>
גדול מ	>
גדול מ .. או שווה ל..	>=
קטן מ	<
קטן מ.. או שווה ל ..	=<
בין שני ערכים	BETWEEN ... AND ...
לא בין שני ערכים	NOT BETWEEN... AND
שווה לאחד הערכים שברשימה	IN(Values)
הערך הוא NULL	IS NULL
שווה לאחד הערכים שברשימה	NOT IN(Values)
הערך אינו NULL	IS NOT NULL
השוואה לתבנית מחרוזתית	LIKE (WILDCRADS)
מציג ערך אחד, במידה ויש כפולים	DISTINCT

# פעולות אריתמטיות מסכמות

פעולה	משמעות
COUNT	ספירה
SUM	סכימה
AVG	חישוב ממוצע
MIN	הצגת ערך מינימלי
MAX	הצגת ערך מקסימלי
ABS	הצגת ערך מוחלט



```
SELECT Country_Name, COUNT(CustomerID) FROM Customers  
GROUP BY Country_Name
```

# הסרת כפילויות מהתצוגה - DISTINCT

---

▶ פונקציית DISTINCT תמיד תהיה בתחילת ה-SELECT.

```
SELECT DISTINCT CustomerName  
FROM Customers
```

▶ ניתן להשתמש ביותר משדה אחד עם DISTINCT, המשמעות היא שהסרת הכפילויות תתבצע עבור השילוב החד-חד ערכי של כל השדות.

```
SELECT DISTINCT CustomerName, Country  
FROM Customers
```

▶ ניתן להשתמש ב-DISTINCT בתוך פונקציית COUNT, המשמעות היא שהספירה תתבצע על ערכים חד-חד ערכיים.

```
SELECT COUNT(DISTINCT CustomerName)  
FROM Customers
```



# מניפולציות טקסטואליות

משמעות	פעולה
מחזיר את מספר התווים המצוין, כאשר ההתחלה מצד שמאל.	LEFT ( <i>string, length</i> )
מחזיר את מספר התווים המצוין, כאשר ההתחלה מצד ימין.	RIGHT( <i>string, length</i> )
מחזיר את מספר התווים המצוין, כאשר ההתחלה הינה לפי התו המצוין.	SUBSTRING( <i>expression, char start, length</i> )

משמעות	פעולה
ממיר את השדה שמצוין מסוג שדה קיים לסוג שדה חדש.	<code>CAST(<i>field AS datatype</i>)</code> <input type="text" value="משנה באופן זמני"/>
ממיר את השדה שמצוין מסוג שדה קיים לסוג שדה חדש, כאשר ניתן לשנות את מבנה סוג השדה (בעיקר תאריך)	<code>CONVERT(datatype, field, [style])</code> <input type="text" value="משנה באופן קבוע"/>

# פעולות על תאריכים

פעולה	משמעות
YEAR(Field)	מחלץ שנה מתאריך
MONTH(Field)	מחלץ חודש מתאריך
 DAY(Field)	מחלץ יום מתאריך
DATEADD(interval, num, date)	מוסיף / מפחית ערכים מתאריך
DATEDIFF(interval, start date, end date)	מציג את ההפרש בין שתי תאריכים

# התניות ברמת ה-SELECT

---

ניתן לבצע התניות על שדות שנבחרו ברמת ה-SELECT כאשר חייבת להיות תוצאה. ▶

סינטקס: ▶

SELECT

CASE WHEN CustomerID < 100 THEN 'A'

WHEN CustomerID BETWEEN 100 AND 500 THEN 'B'

ELSE 'C'

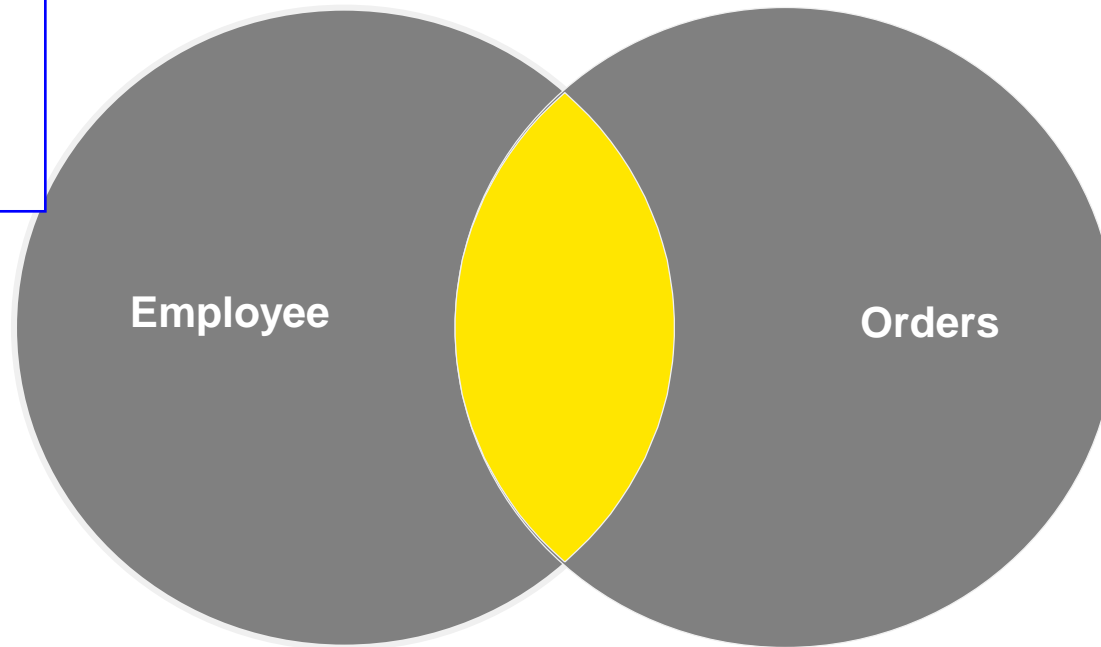
END AS GroupType

FROM Customers

# חיבור בין טבלאות – INNER JOIN

כשמחברים טבלאות, נשים קודם את הטבלה החשובה שאליה אנחנו עושים חיבורים/ חיסורים. הראשונה תמיד תהיה החשובה.

ON, כשעושים את החיבור נשים קודם את הטבלה האחרונה, ורק אז את הראשונה



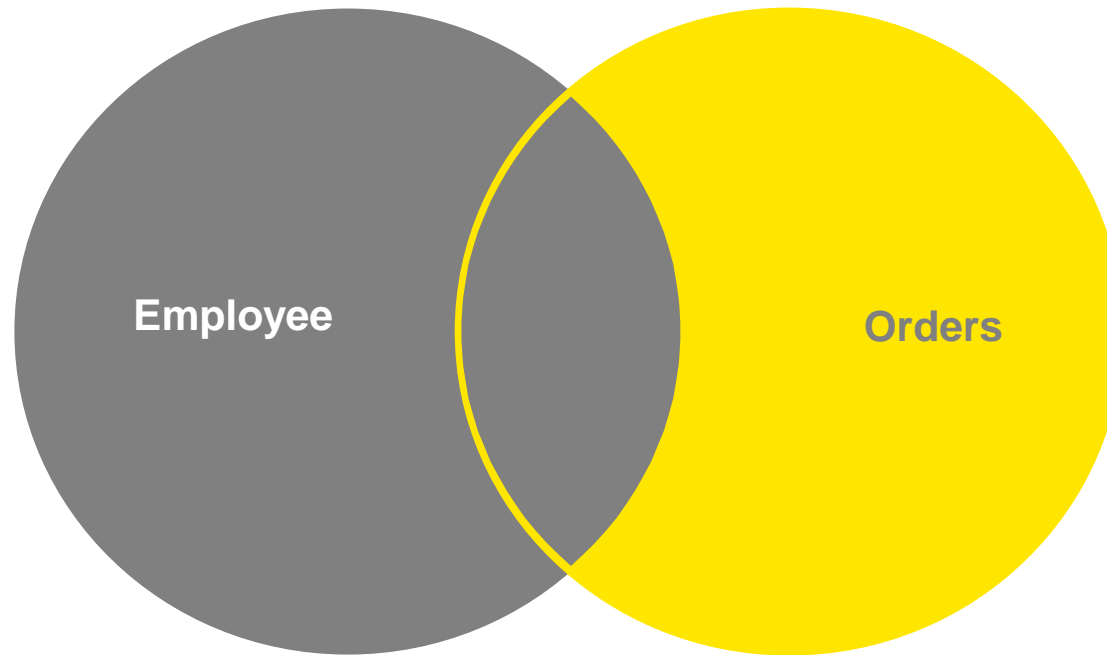
Result



Exclude

```
SELECT *  
FROM Table1 a INNER JOIN  
Table2 b ON a.ID = b.ID
```

# LEFT JOIN – חיבור בין טבלאות



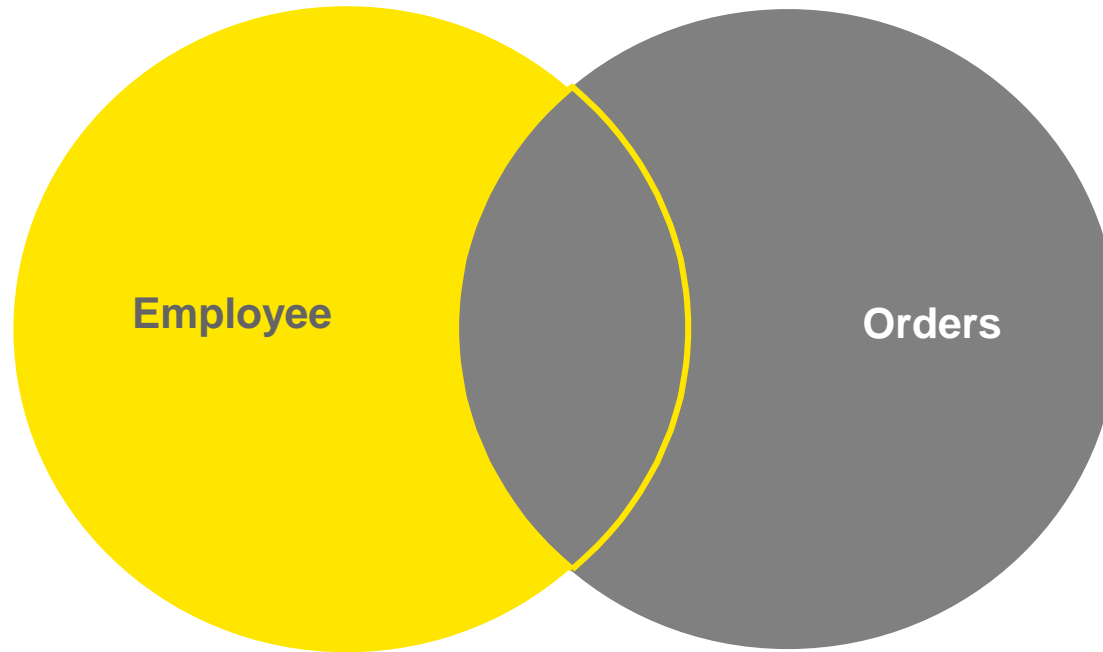
Result



Exclude

```
SELECT *  
FROM Table1 a LEFT JOIN  
Table2 b ON a.ID = b.ID
```

# חיבור בין טבלאות – RIGHTJOIN



```
SELECT *  
FROM Table1 a RIGHT JOIN  
Table2 b ON a.ID = b.ID
```

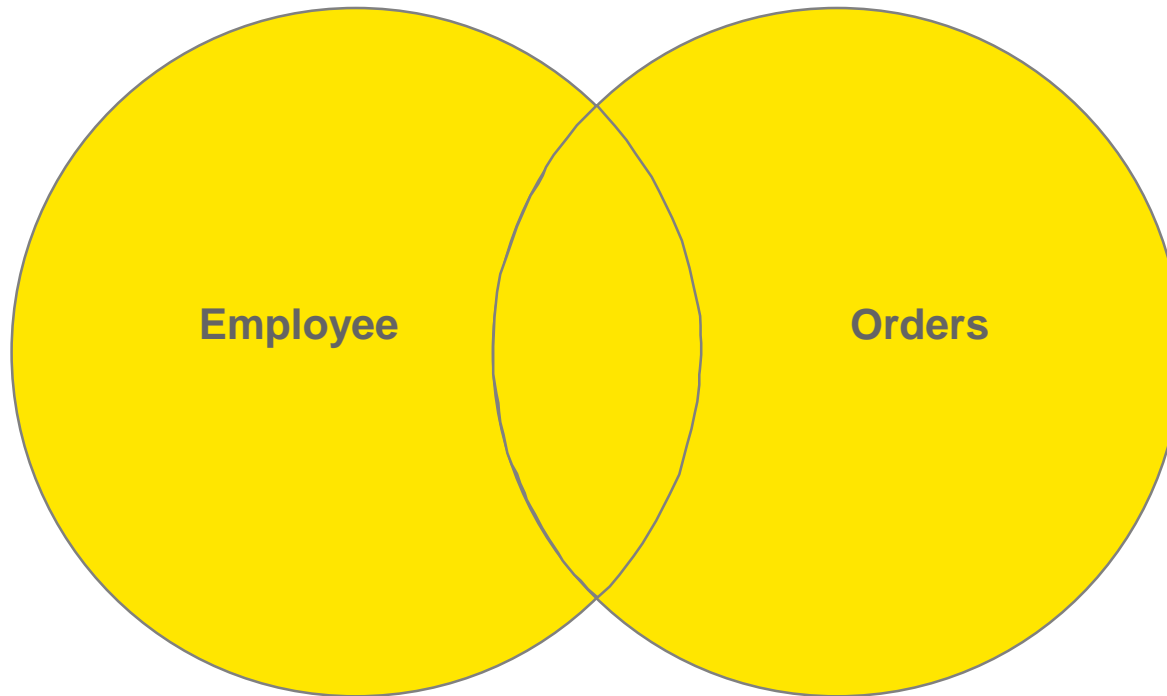


Result



Exclude

# חיבור בין טבלאות – FULL OUTER JOIN



```
SELECT *  
FROM Table1 a FULL OUTER JOIN  
Table2 b ON a.ID = b.ID
```

 Result  
 Exclude



<http://www.w3schools.com/sql/default.asp> ►

---

# שאלות?