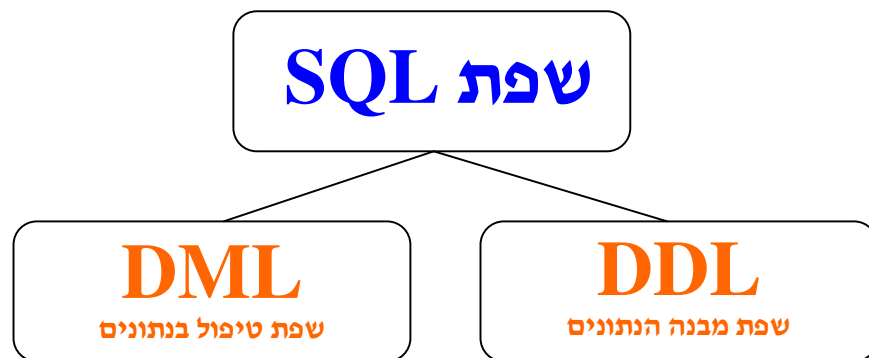




תרגול מס' 2

תזכורת

לאור העובדה שהראנו את היתרונות של המודל הטבלאי, נצטרך כעת ללמוד את השפה שאיתה נוכל לכתוב שאילתות לשליפת נתונים מבסיס הנתונים. השפה איתה נעבוד נקראת שפת SQL כאשר שפת SQL מחולקת ל 2 תתי שפות:



• DML - שפת טיפול בנתונים (Data Manipulation Language) - שפה המשמשת לטיפול בנתונים.

- פקודת SELECT - המשפט המוכר ביותר בקטגוריה זו, מחזיר נתונים מתוך טבלה/טבלאות
- פקודת INSERT מוסיפה שורות חדשות לטבלה קיימת
- פקודת UPDATE מעדכנת נתונים בשורות הקיימות בטבלה
- פקודת DELETE מוחקת שורות נתונים מטבלה

• DDL - שפת הגדרת נתונים (Data Definition Language) - שפה המשמשת לטיפול במבנה הנתונים (הוספת ומחיקת טבלאות, עמודות ואינדקסים), ולקביעת הרשאות גישה לרכיבי בסיס הנתונים.

- פקודת CREATE יוצרת פריט מבנה חדש - טבלה, עמודה, אינדקס וכד'
- פקודת ALTER משנה את תכונותיו של פריט קיים
- פקודת DROP מוחקת פריט קיים
- פקודת GRANT מעניקה זכות גישה למשתמש או לקבוצת משתמשים
- פקודת REVOKE מונעת זכות גישה ממשתמש או מקבוצה



מבנה פקודה בסיסית

מבנה כללי של פקודת SELECT שבו נשתמש לאורך כל הקורס:

SELECT *	שמות כל העמודות שנרצה לשלוח מכל הטבלאות המשתתפות בפעולת השליפה
FROM Students	שמות כל הטבלאות המשתתפות בפעולת השליפה
WHERE Students.id = 3344	כל התנאים שנרצה להתנות על המידע מהטבלאות המשתתפות בשליפה

הסבר על מבנה הפקודה:

- פקודת ה **SELECT** - ביצוע פעולה של בחירת עמודות מסוימות מתוך טבלה (מופרדים בפסיקים ביניהם).
- פקודת ה **FROM** – אוסף הטבלאות שישתתפו בשליפה הנוכחית.
- פקודת ה **WHERE** - ביצוע פעולה של בחירת (סינון) השורות המקיימות את התנאי או התנאים המוגדרים בה – פקודה זו **הינה אופציונאלית** בביצוע השליפה. ללא פקודה זו תתבצע שליפה של כל השורות בטבלה (שקול לאי מתן תנאי לסינון תוצאת השליפה).

מבנה פקודה מתקדמת

מבנה פקודת SQL מתקדמת (בסוף הקורס תדעו לכתוב פקודות מסוג זה ואפילו מסובכות יותר):

SELECT	Name, Count(*)
FROM	Students AS S
LEFT JOIN	Courses AS C
ON	S.courseID = C. courseID
WHERE	Students.id BETWEEN 3344 AND 3355
GROUP BY	Name
HAVING	Count(*) >2
ORDER BY	Name



יצירת טבלה חדשה ב SQL

על מנת ליצור טבלה חדשה ב SQL נצטרך לכתוב פקודה במבנה הבא :

```
CREATE TABLE table_name  
(  
    column_name1 data_type,  
    column_name2 data_type,  
    column_name3 data_type,  
    ....  
)
```

פירוט של 4 השלבים ביצירת טבלה חדשה :

1) הגדרת הסכמה של הטבלה : נגדיר את העמודות שאנו רוצים שיהיו בטבלה החדשה שניצור וליד כל עמודה נגדיר את ה TYPE שלה שיגדיר איזה ערכים היא יכולה לקבל :

```
CREATE TABLE Students  
(  
    Id int,  
    First_Name text,  
    Last_Name text,  
    Birth_Date date,  
    Hour_Salary int  
);
```

1

2) הגדרת שם הטבלה שתיווצר ותכיל את הגדרות הסכמה שלהן :

```
INSERT INTO Students
```

2

3) הכנסת ערכים לשורה בטבלה בהתאם לסכמה שהוגדרה :

```
VALUES( 123456789, 'Israel', 'Israeli', '1980/03/20', 120.40 );
```

3

הערה : כשנרצה להכניס מספר רשומות בפקודה אחת נוכל לרשום את הפקודה הבאה :

```
Values( , , , ), ( , , , ), ( , , , ), ( , , , )  
    רשומה 1 רשומה 2 רשומה 3 רשומה 4
```



4) הרצת שאילתא לשליפת כל הנתונים בטבלה במטרה לראות שהטבלה נבנתה והערכים שהוכנסו קיימים בה:

```
SELECT Id, First_Name, Last_Name, Birth_Date, Hour_Salary
FROM Students;
```

Students				
Id	First_Name	Last_name	Birth_date	Hour_Salary
0123456789	Israel	Israeli	20/03/1980	120.40

עבודה למול MySQL

MySQL - הינו שרת בסיס נתונים רב נימי רוב משתמשים מבוסס שפת SQL, האפליקציה הינה חנימית וניתנת להורדה חופשית בשנים האחרונות מסד נתונים זה זוכה לפופולריות רבה, ומאומץ גם על ידי ארגונים גדולים יחסית כ-Yahoo, NASA וחברת סווקי.

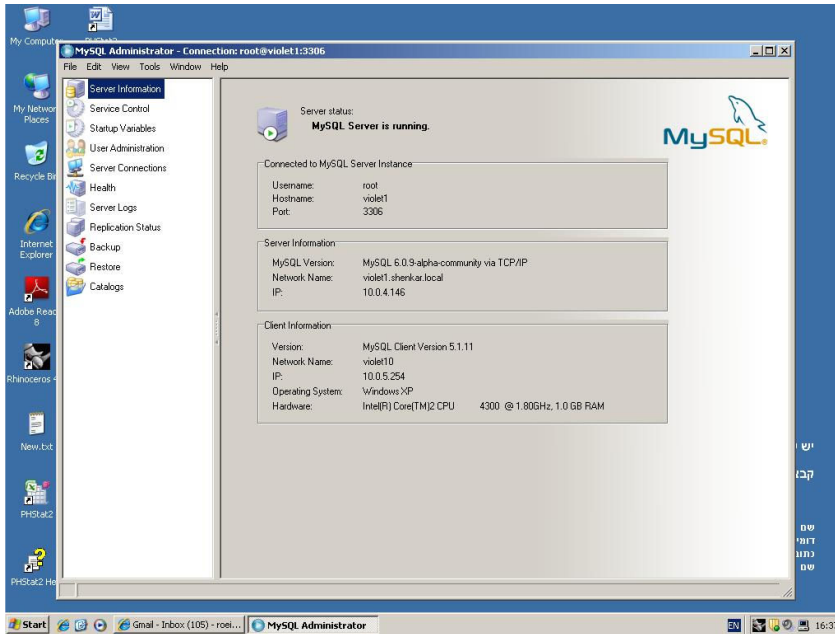
MySQL Administrator

מסך הניהול של התכנה MySQL Administrator המאפשר שליטה ויזואלית בסיסי הנתונים - ביצוע כל סוגי השאילתות (צפייה בתוכן טבלאות, שינוי מבנה טבלאות, הוספה, מחיקה וכו')

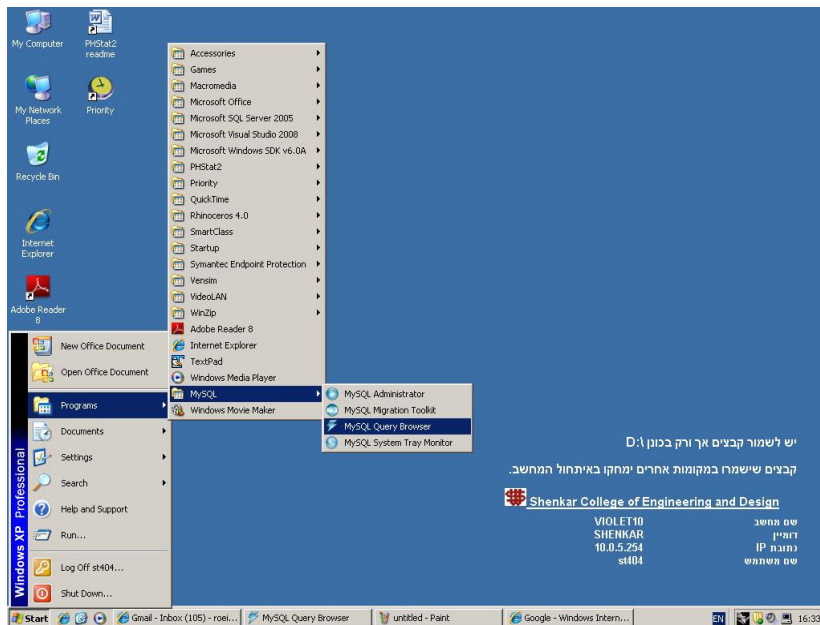
דפדפן השאילתות של MySQL שבאמצעותו נבנות שאילתות SQL באופן חזותי. מודל הניווט בכלי מחקה את המודל של דפדפן אינטרנט.



א) הגדרות כלליות



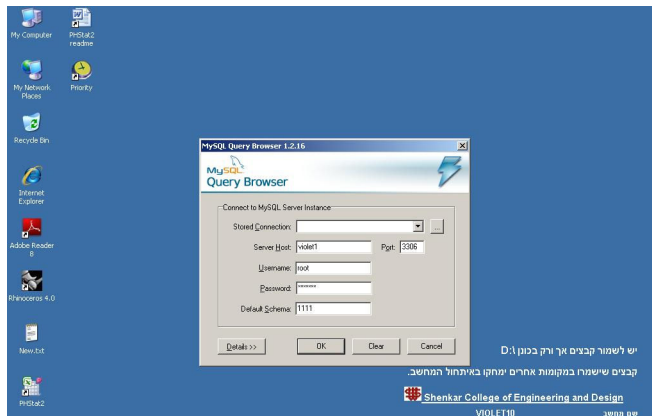
ב) כניסה למערכת



Start → Programs → MySql → MySql Query Browser

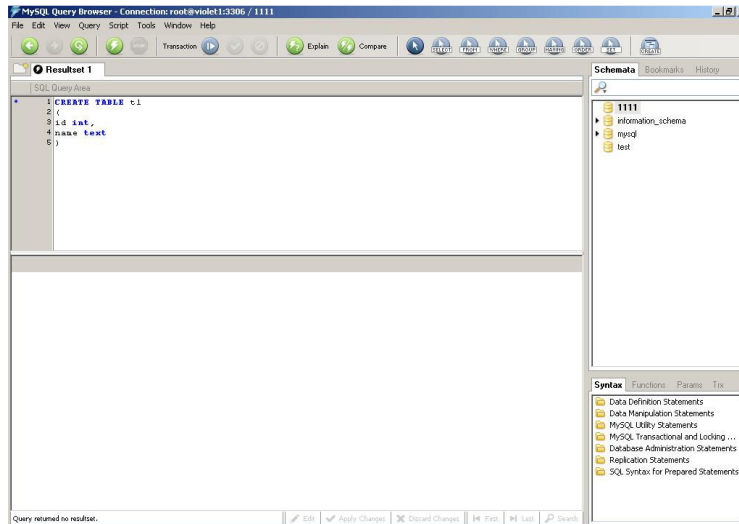


ג) התחברות לשרת בסיס הנתונים (בשנקר)



Server Host: Violet1
Username: root
Password: shenkar
Scheme: 1111 (students will enter the ID number)

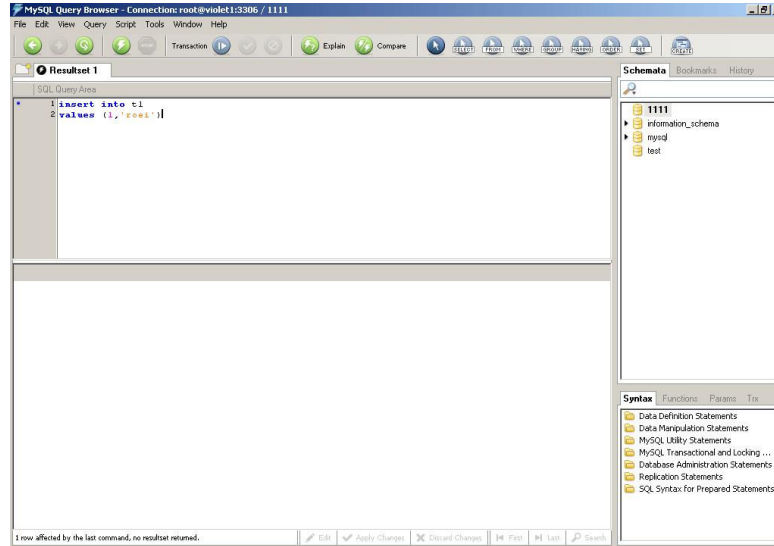
ד) יצירת טבלה חדשה (תחת סכמה 1111) - (DDL)



CREATE TABLE t1
(
 Id int ,
 Name text
)



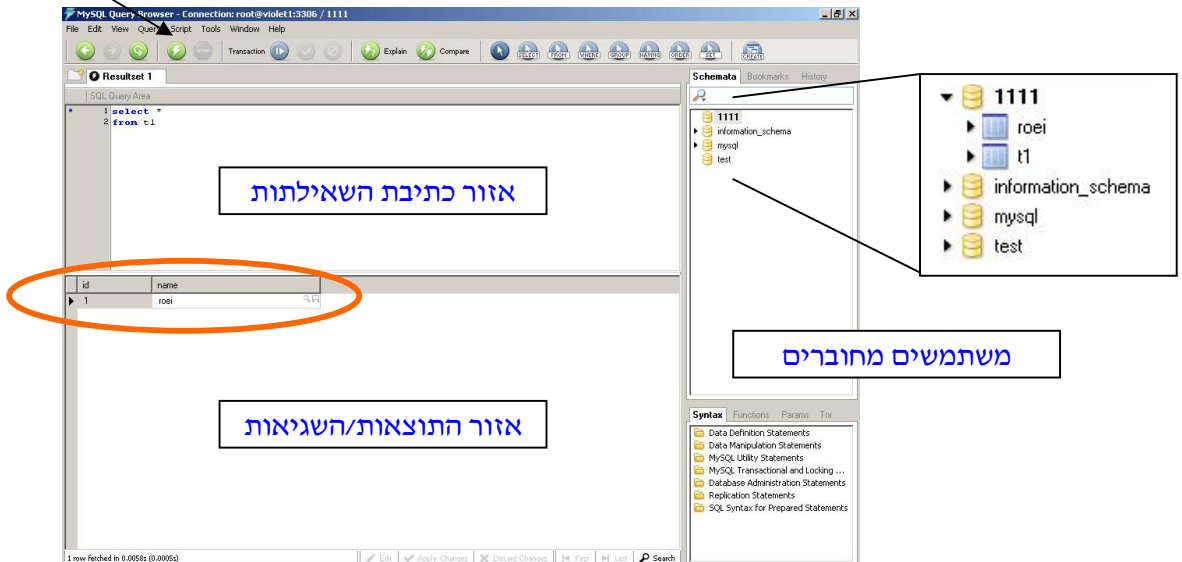
(ה) הכנסת ערכים לתוך הטבלה החדשה (DML)



**INSERT INTO t1
VALUES (1 , 'roei')**

כפתור להצגת השאילתות

(ו) שליפת נתונים מטבלה קיימת



**SELECT *
FROM t1**



כתיבת שאילתות פשוטות

נגדיר טבלת מידע שעליה נבצע את הדוגמאות עבור הפעולות לשיעור זה:

Students			
FirstName	LastName	ID	Grade
Avi	Cohen	3333	92
Ben	Levi	3456	80
Chen	Levin	4321	95
Debi	Dvir	3344	92

1) פקודת ה SELECT (חובה)

דוגמאות:

1) הצגת כל המידע מטבלת הסטודנטים:

```
SELECT *
FROM Students;
```

FirstName	LastName	ID	Grade
Avi	Cohen	3333	92
Ben	Levi	3456	80
Chen	Levin	4321	95
Debi	Dvir	3344	92

2) שליפת שמותיהם המלאים של כל הסטודנטים:

```
SELECT LastName , FirstName
FROM Students;
```

LastName	FirstName
Cohen	Avi
Levi	Ben
Levin	Chen
Dvir	Debi

3) שליפת רשימת הציונים של כל הסטודנטים:

```
SELECT Grade
FROM Students;
```

Grade
92
80
95
92



נוכל לראות שיש ציון המופיע פעמיים '92', ב SQL המצב הוא תקין ולכן כאשר נרצה לבטל הופעת שורות כפולות נוסיף את פקודת ה **DISTINCT** מיד לאחר פקודת ה **SELECT** ולפני רשימת העמודות שנרצה לשלוף:

SELECT DISTINCT Grade
FROM Students;

Grade
92
80
95

הערה: אם אחרי פקודת ה **DISTINCT** היינו רושמים מספר עמודות אזי היה מבטל כפילויות ברמת רשומה ולא ברמת השדה הבודד כמו במקרה לעיל.

פעולות נוספות שנוכל לבצע במהלך שליפת הנתונים:

דוגמאות:

1) נניח שלמרות שהתרגילים בקורס הם נורא קלים אך עדיין הוחלט לתת פקטור של 3 נקודות לכל הסטודנטים, נוכל לבצע זאת בפקודה אחת תוך כדי ביצוע השליפה:

נשלוף את כל ת.ז. של הסטודנטים והציון המעודכן שלהם:

SELECT id, grade + 3
FROM Students;

ID	Grade
3333	95
3456	83
4321	98
3344	95

2) נניח שנרצה להוסיף עמודה קבועה לטבלת התוצאה הנ"ל המסבירה שלסטודנט עם ת.ז. X יש ציון Y, נוכל לרשום זאת כך:

SELECT id, 'with grade', grade
FROM Students;

ID	With Grade	Grade
3333	With Grade	92
3456	With Grade	80
4321	With Grade	95
3344	With Grade	92

נקבל את המחרוזת שביקשנו גם בכותרת וגם בכל אחד מהשורות בטבלה.