

פרויקט בבסיסי נתונים

מחלקת IT של מכון לב



מגישים:

ישראל דזיקוב 316481514

מרדכי כהן 206958035

תוכן עניינים

5	שלב א'.....
5	תיאור כללי של הארגון.....
5	תיאור מילולי של הארגון.....
5	תיאור מילולי של הטבלאות.....
6	פירוט הישויות.....
8	דיאגרמת ERD.....
9	תרשים DSD.....
10	קשרים בין ישויות.....
11	פקודות.....
11	Create.....
13	Drop.....
14	Insert.....
17	Select.....
17	Delete.....
17	Update.....
18	שלב ב'.....
18	אכלוס הטבלאות.....
18	1. Data Generation - יצירת נתונים.....
27	2. ODBC Importer.....
29	3. Text Importer.....
31	4. הכנסה ידנית.....
33	שאלות.....
33	שאלת ראשונה – החלפת מדפסות.....
34	שאלת שניה – איזון בין הצוותים.....
35	שאלת שלישית – שדרוג מחשבים.....
36	שאלת רביעית – תמונת מצב.....
37	שאלת חמישית – הספקים המובילים.....
37	שאלת שישית – מיון הפרויקטים.....
39	שאלת שביעית – הפרויקטים הרווחיים.....
40	שאלת שמינית – הדרכת שימוש בתוכנות.....
41	אינדקסים.....

41	אינדקסים שמשפרים את זמן הריצה.....
41	נשפר את הביצוע של השאילתה הראשונה – החלפת מדפסות.....
42	נשפר את הביצוע של השאילתה השנייה – איזון בין הצוותים.....
43	נשפר את הביצועים של השאילתה הרביעית – תמונת מצב.....
44	נשפר את הביצוע של השאילתה החמישית – הספקים המובילים.....
45	אינדקסים לא מועילים.....
45	אינדקס לא מועיל בשאילתה השלישית – שדרוג מחשבים.....
46	אינדקס לא מועיל בשאילתה השמינית – הדרכת שימוש בתוכנות.....
47	שלב ג'.....
47	שמירת הפרויקט ב-GitHub.....
48	גיבוי הנתונים.....
49	Grants.....
50	שאילתה ראשונה.....
51	שאילתה שניה.....
52	שאילתה שלישית.....
53	שלב ד'.....
53	Views.....
53	1. נרצה ליצור טבלה עם פרטים חלקיים של עובדים ומידע על התפקיד והצוות שלהם.....
54	שאילתה 1.....
55	שאילתה 2.....
56	2. נרצה ליצור טבלה שמורכבת מפרטים על פרויקט ועל הקומה שבה יושב הצוות שעובד עליו.....
57	שאילתה 1.....
58	שאילתה 2.....
59	פונקציות.....
59	1. הערכה אחרונה של עובד.....
60	2. פרויקט קרוב של צוות.....
61	פרוצדורות.....
61	1. חיבור מחשב למדפסת.....
63	2. מחיקת פרויקטים ישנים.....
64	טריגרים.....
64	1. מחיקת מדפסת.....
65	2. הוספת מחשב.....
66	שלב ה'.....

66דוחות של שאילתות עם פרמטרים
66שאילה ראשונה עם פרמטרים
67שאילה שניה עם פרמטרים
68גרפים
68שאילתה ראשונה + גרף
69שאילתה שניה + גרף

שלב א'

תיאור כללי של הארגון

בפרויקט נבנה בסיס נתונים למרכז התמיכה והתקשוב של מכון לב, מחלקת ה IT.

צוות ה IT של המרכז האקדמי לב דואג לכל החלק הטכנולוגי התפעולי של המכון.

צוות ה IT אחראי:

- 1) לעזור ולטפל בתקלות הסטודנטים, המרצים וההנהלה.
- 2) לתחזק ולתפעל את אתרי המכון כגון מודל, לב נט, מידע ורישום.
- 3) לטפל במעבדות ובכיתות הלימוד מבחינת ציוד טכני של מחשבים, מצלמות, מקרנים ומסכים.
- 4) לתחזק את המדפסות הפזורות במכון.
- 5) לתחזק ולהתקין תוכנות הנדרשות למחשבי המכון.
- 6) לתחזק את ה WIFI.

תיאור מילולי של הארגון

לכל עובד יש תפקיד והערכה (כלומר ותק ואיכות עבודה שנתית). בכל צוות יש עובדים המשוויכים אליו, ולכל צוות יש ראש צוות שמנהל אותו ואחראי עליו.

לכל צוות יש חדר אחד בו הם יושבים ופרויקטים שהוא אחראי עליהם (יכול להיות יותר מפרויקט אחד לצוות).
לכל עובד יש מחשב אחד ובו מותקנים תוכנות שונות שנצרכות לצורך עבודה במכון וכן מחוברים אליו מדפסות.
לכל תוכנה ולכל מדפסת יש ספק אחד שאחראי עליה ועל התחזוקה שלה.

תיאור מילולי של הטבלאות

- 1) **Employee** (EmID, EmName, EmPhone, EmMail, DataOfBirth)
- 2) **Team** (TeID, TeName)
- 3) **Role** (RoleID, RoleName)
- 4) **Evaluation** (EvID, Grade, Year)
- 5) **Room** (RoomNumber, Space, Floor)
- 6) **Project** (proID, ProName, DueData, Profit, Cost)
- 7) **Computer** (C_ID, Procssor, RAM, SSD, YearOfPurchase)
- 8) **Printer** (PriID, Model, Compeny)
- 9) **Connected_to** (C_ID, PriID)
- 10) **Software** (SoID, So_Name, Version)
- 11) **Installed_on** (C_ID, SoID)
- 12) **Supplier** (SuID, Su_Name, Addres, Su_phone, Su_mail)

פירוט הישויות
Employee - עובדים

תעודת זהות (מפתח)	EmID
שם	EmName
טלפון	EmPhone
מייל	EmMail
תאריך הלידה	DataOfBirth
מזהה תפקיד (מפתח זר)	RoleID
מזהה צוות (מפתח זר)	TeID
מזהה מחשב (מפתח זר)	C_ID

Team – צוות

מספר מזהה (מפתח)	TeID
שם	TeName
מספר חדר (מפתח זר)	RoomNumber

Role – תפקיד

מספר מזהה (מפתח)	RoleID
שם תפקיד	RoleName

Evaluation – הערכה

מספר מזהה (מפתח)	EvID
ציון	Grade
וותק (בשנים)	Year
מזהה עובד (מפתח זר)	EmID

Room – חדר

מספר חדר (מפתח)	RoomNumber
שטח	Space
קומה	Floor

Project – פרויקט

מספר מזהה (מפתח)	proID
שם	ProName
תאריך יעד	DueData
רווח	Profit
עלויות	Cost
מזהה צוות (מפתח זר)	TeID

Computer – מחשבים

C_ID	מספר מזהה (מפתח)
Proccssor	סוג מעבד
RAM	זיכרון גישה אקראית
SSD	כונן שבבי
YearOfPurchase	תאריך יצור

Printer – מדפסת

PrilID	מספר מזהה (מפתח)
Model	דגם
Compeny	חברה מייצרת
SuID	מזהה ספק (מפתח זר)

Connected_To – מחשב מחובר למדפסת

PrilID	מספר מדפסת (מפתח) , (מפתח זר)
CID	מספר מחשב (מפתח) , (מפתח זר)

Software – תוכנות

SoID	מספר מזהה (מפתח)
So_Name	שם
Version	גרסה
SuID	מזהה ספק (מפתח זר)

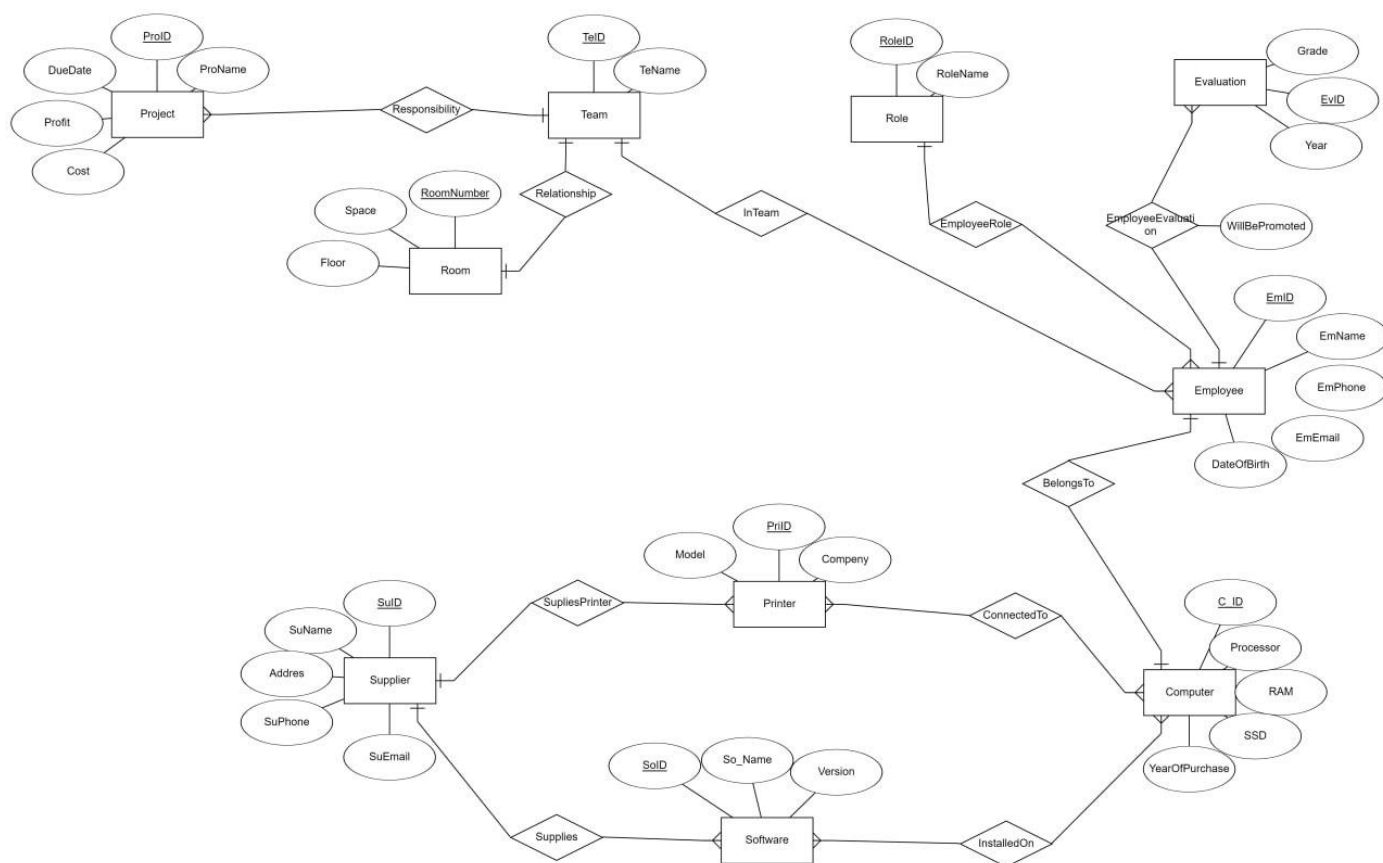
Installed_On – תוכנה מותקנת על מחשב

SoID	מספר תוכנה (מפתח) , (מפתח זר)
CID	מספר מחשב (מפתח) , (מפתח זר)

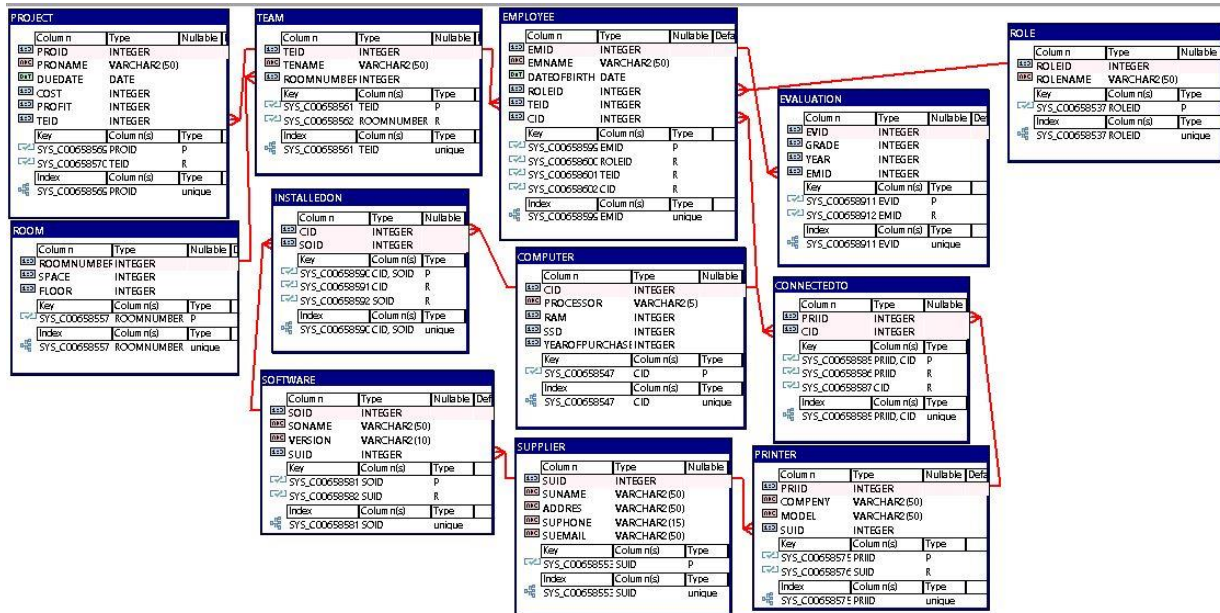
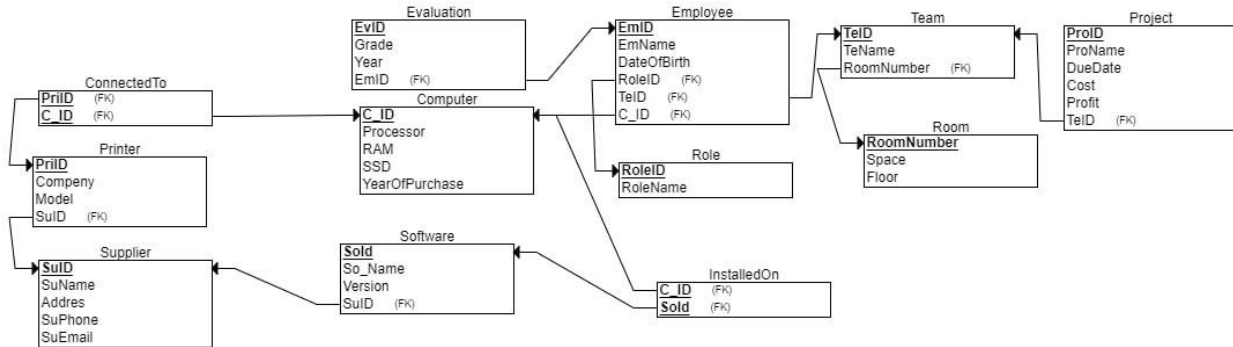
Supplier – ספק

SuID	מספר ספק (מפתח)
Su_Name	שם
Addres	כתובת
Su_phone	טלפון
Su_mail	מייל

דיאגרמת ERD



תרשים DSD



קשרים בין ישויות

שם הקשר	פירוט
Employee_Evaluation	קשר יחיד לרבים בין עובד להערכה. לכל עובד יכולות להיות כמה הערכות
Employee_Role	קשר יחיד לרבים בין תפקיד לעובד. לכל תפקיד יכולים להיות מספר עובדים באותו תפקיד
Employee_Team	קשר יחיד לרבים בין צוות לעובד. כל צוות מכיל מספר עובדים
Team_Room	קשר יחיד ליחיד בין צוות לחדר. כל צוות יושב לבדו בחדר מסוים
Team_responsibility_Project	קשר יחיד לרבים בין צוות לפרויקט. כל צוות אחראי על מספר פרויקטים
Computer_belongs_To_Employee	קשר יחיד ליחיד בין עובד למחשב. לכל עובד יש מחשב אישי שלו
Computer_Connected_To_Printer	קשר רבים לרבים בין מחשב למדפסת. כל מחשב מחובר לכמה מדפסות וכל מדפסת מחוברת לכמה מחשבים
Software_InstalledOn_Computer	קשר רבים לרבים בין תוכנה למחשב. כל תוכנה מותקנת על כמה מחשבים ועל מחשב מותקנים כמה תוכנות
Printer_Supplied_By_Supplier	קשר יחיד לרבים בין ספק למדפסת. כל ספק מספק כמה מדפסות
Software_Supplied_By_Supplier	קשר יחיד לרבים בין ספק לתוכנה. כל ספק מספק כמה תוכנות



```
CREATE TABLE Evaluation
(
    EvID INT NOT NULL,
    Grade INT NOT NULL,
    Year INT NOT NULL,
    EmID INT NOT NULL,
    PRIMARY KEY (EvID),
    FOREIGN KEY (EmID) REFERENCES Employee(EmID)
);
```

```
CREATE TABLE Computer
(
    CID INT NOT NULL,
    Processor VARCHAR(5) NOT NULL,
    RAM INT NOT NULL,
    SSD INT NOT NULL,
    YearOfPurchase INT NOT NULL,
    PRIMARY KEY (CID)
);
```

```
CREATE TABLE Team
(
    TeID INT NOT NULL,
    TeName VARCHAR(50) NOT NULL,
    RoomNumber INT NOT NULL,
    PRIMARY KEY (TeID),
    FOREIGN KEY (RoomNumber) REFERENCES Room(RoomNumber)
);
```

```
CREATE TABLE Supplier
(
    SuID INT NOT NULL,
    SuName VARCHAR(50) NOT NULL,
    Address VARCHAR(50) NOT NULL,
    SuPhone VARCHAR(15) NOT NULL,
    SuEmail VARCHAR(50) NOT NULL,
    PRIMARY KEY (SuID)
);
```

```
CREATE TABLE Software
(
    SoId INT NOT NULL,
    SoName VARCHAR(50) NOT NULL,
    Version VARCHAR(10) NOT NULL,
    SuID INT NOT NULL,
    PRIMARY KEY (SoId),
    FOREIGN KEY (SuID) REFERENCES Supplier(SuID)
);
```

```
CREATE TABLE Room
(
    RoomNumber INT NOT NULL,
    Space INT NOT NULL,
    Floor INT NOT NULL,
    PRIMARY KEY (RoomNumber)
);
```

```
CREATE TABLE Project
(
    ProID INT NOT NULL,
    ProName VARCHAR(50) NOT NULL,
    DueDate DATE NOT NULL,
    Cost INT NOT NULL,
    Profit INT NOT NULL,
    TeID INT NOT NULL,
    PRIMARY KEY (ProID),
    FOREIGN KEY (TeID) REFERENCES Team(TeID)
);
```

```
CREATE TABLE Printer
(
    PriID INT NOT NULL,
    Company VARCHAR(50) NOT NULL,
    Model VARCHAR(50) NOT NULL,
    SuID INT NOT NULL,
    PRIMARY KEY (PriID),
    FOREIGN KEY (SuID) REFERENCES Supplier(SuID)
);
```

```
CREATE TABLE Role
(
  RoleID INT NOT NULL,
  RoleName VARCHAR(50) NOT NULL,
  PRIMARY KEY (RoleID)
);
```

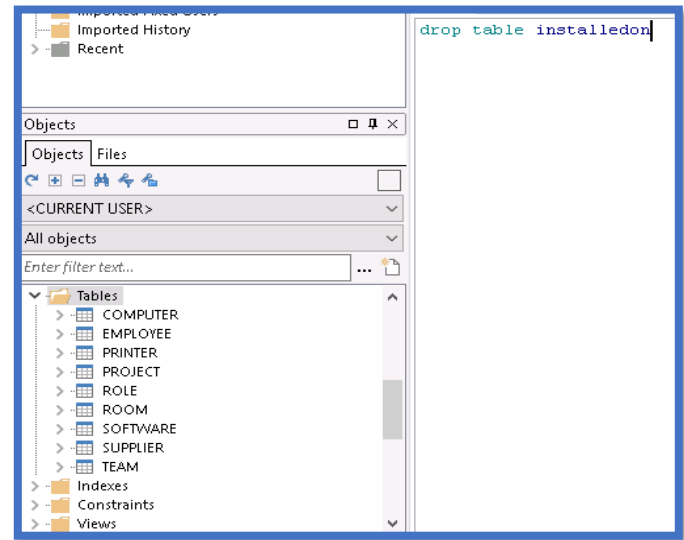
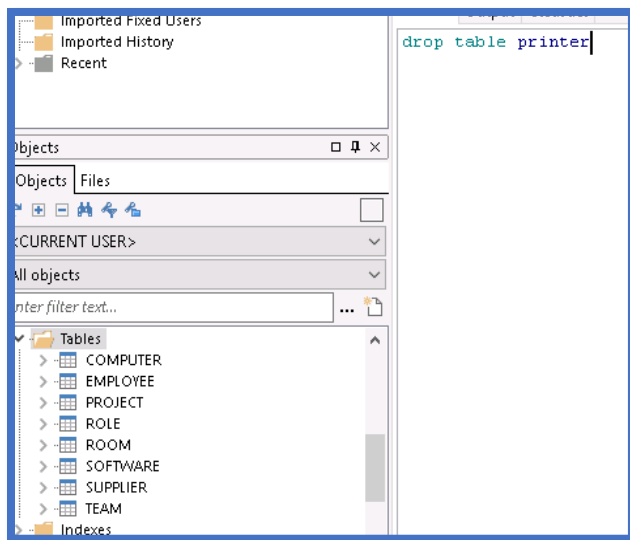
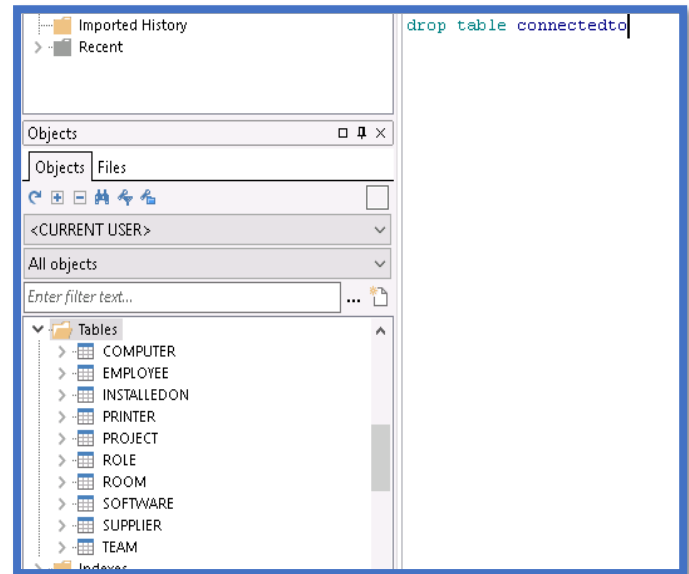
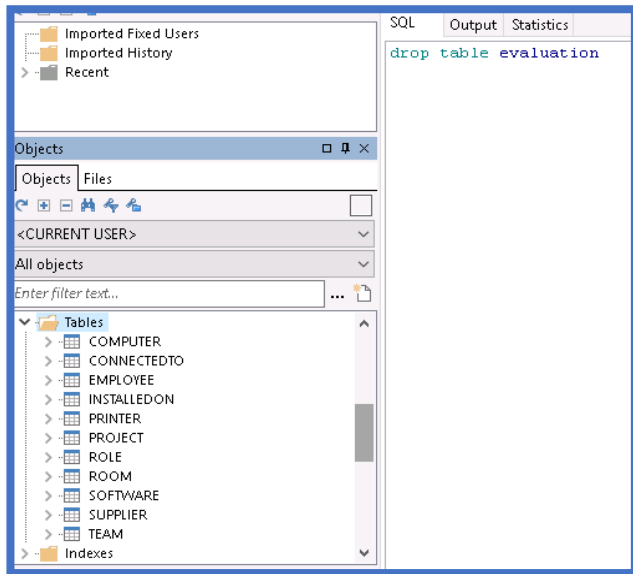
```
CREATE TABLE ConnectedTo
(
  PriID INT NOT NULL,
  CID INT NOT NULL,
  PRIMARY KEY (PriID, CID),
  FOREIGN KEY (PriID) REFERENCES Printer(PriID),
  FOREIGN KEY (CID) REFERENCES Computer(CID)
);
```

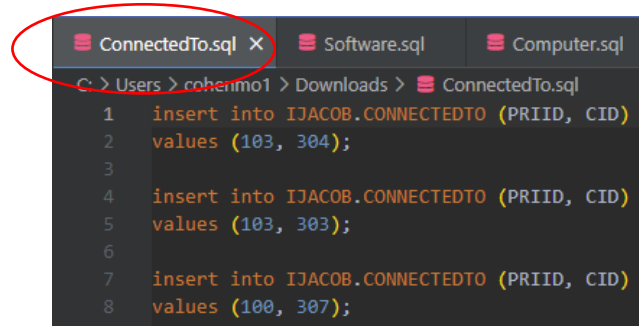
```
CREATE TABLE Employee
(
  EmID INT NOT NULL,
  EmName VARCHAR(50) NOT NULL,
  DateOfBirth DATE NOT NULL,
  RoleID INT NOT NULL,
  TeID INT NOT NULL,
  CID INT NOT NULL,
  PRIMARY KEY (EmID),
  FOREIGN KEY (RoleID) REFERENCES Role(RoleID),
  FOREIGN KEY (TeID) REFERENCES Team(TeID),
  FOREIGN KEY (CID) REFERENCES Computer(CID)
);
```

```
CREATE TABLE InstalledOn
(
  CID INT NOT NULL,
  SoId INT NOT NULL,
  PRIMARY KEY (CID, SoId),
  FOREIGN KEY (CID) REFERENCES Computer(CID),
  FOREIGN KEY (SoId) REFERENCES Software(SoId)
);
```



Drop

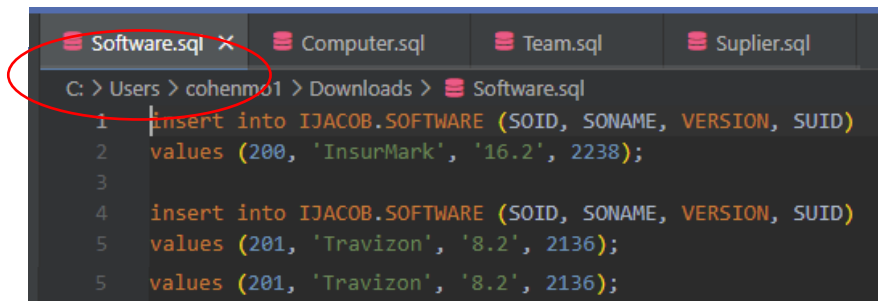




```

C:\Users\cohenmo1\Downloads\ConnectedTo.sql
1 insert into IJACOB.CONNECTEDTO (PRIID, CID)
2 values (103, 304);
3
4 insert into IJACOB.CONNECTEDTO (PRIID, CID)
5 values (103, 303);
6
7 insert into IJACOB.CONNECTEDTO (PRIID, CID)
8 values (100, 307);

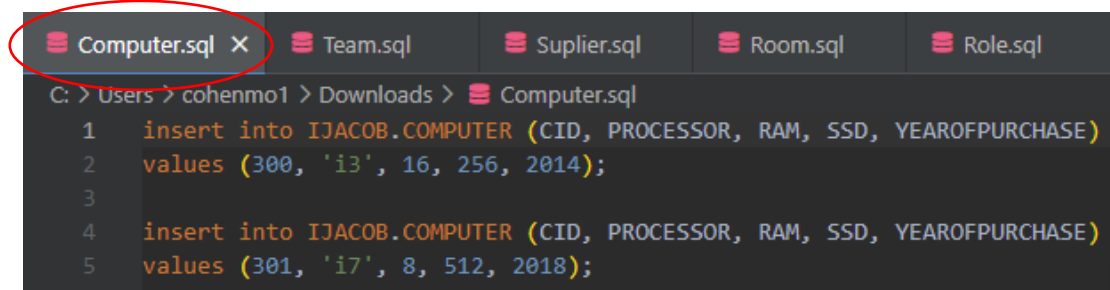
```



```

C:\Users\cohenmo1\Downloads\Software.sql
1 insert into IJACOB.SOFTWARE (SOID, SONAME, VERSION, SUID)
2 values (200, 'InsurMark', '16.2', 2238);
3
4 insert into IJACOB.SOFTWARE (SOID, SONAME, VERSION, SUID)
5 values (201, 'Travizon', '8.2', 2136);
6
7 insert into IJACOB.SOFTWARE (SOID, SONAME, VERSION, SUID)
8 values (201, 'Travizon', '8.2', 2136);

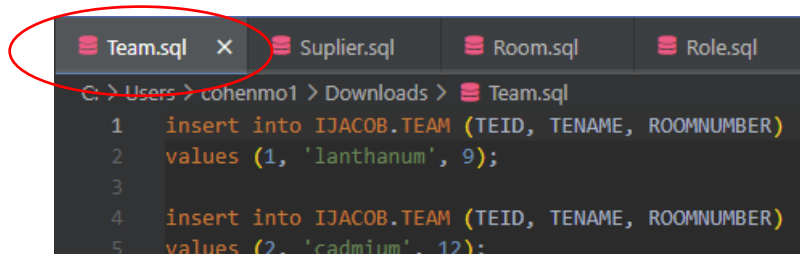
```



```

C:\Users\cohenmo1\Downloads\Computer.sql
1 insert into IJACOB.COMPUTER (CID, PROCESSOR, RAM, SSD, YEAROFPURCHASE)
2 values (300, 'i3', 16, 256, 2014);
3
4 insert into IJACOB.COMPUTER (CID, PROCESSOR, RAM, SSD, YEAROFPURCHASE)
5 values (301, 'i7', 8, 512, 2018);

```



```

C:\Users\cohenmo1\Downloads\Team.sql
1 insert into IJACOB.TEAM (TEID, TENAME, ROOMNUMBER)
2 values (1, 'lanthanum', 9);
3
4 insert into IJACOB.TEAM (TEID, TENAME, ROOMNUMBER)
5 values (2, 'cadmium', 12);

```

```
Supplier.sql X Room.sql Role.sql Project.sql Printer.sql InstalledOn.sql E
C: > Users > cohenmo1 > Downloads > Supplier.sql
1 insert into IJACOB.SUPPLIER (SUID, SNAME, ADDRES, SUPHONE, SUEMAIL)
2 values (2000, 'Visa International', '616 Paxton Street', 514816272, 'martin.crimson@visainter
3
4 insert into IJACOB.SUPPLIER (SUID, SNAME, ADDRES, SUPHONE, SUEMAIL)
5 values (2034, 'Arkidata', '16 Jude', 589855889, 'v.loggia@arkidata.com');
```

```
Role.sql X Project.sql Printer.sql Insta
C: > Users > cohenmo1 > Downloads > Role.sql
1 insert into IJACOB.ROLE (ROLEID, ROLENAME)
2 values (100, 'Network architect');
3
4 insert into IJACOB.ROLE (ROLEID, ROLENAME)
5 values (125, 'Wireless network engineer');
```

```
Room.sql X Role.sql Project.sql Printer.sql
C: > Users > cohenmo1 > Downloads > Room.sql
1 insert into IJACOB.ROOM (ROOMNUMBER, SPACE, FLOOR)
2 values (1, 34, 3);
3
4 insert into IJACOB.ROOM (ROOMNUMBER, SPACE, FLOOR)
5 values (2, 55, 2);
```

```
Project.sql X Printer.sql InstalledOn.sql Evaluation.sql Employee.sql
C: > Users > cohenmo1 > Downloads > Project.sql
1 insert into IJACOB.PROJECT (PROID, PRONAME, DUEDATE, COST, PROFIT, TEID)
2 values (1, '2 Year Warranty On Combo's.', to_date('03-03-2022', 'dd-mm-yyyy'), 6980, 9961, 1)
3
4 insert into IJACOB.PROJECT (PROID, PRONAME, DUEDATE, COST, PROFIT, TEID)
5 values (2, 'Combo AMD Duron 1.2Ghz cpu Gigabyte GA-7VKMLS m/b ', to_date('17-01-2021', 'dd-mm-
```

```
Printer.sql X InstalledOn.sql Evaluation.sql Employee.sql
C: > Users > cohenmo1 > Downloads > Printer.sql
1 insert into IJACOB.PRINTER (PRIID, COMPENY, MODEL, SUID)
2 values (100, 'National Bankcard Systems', '14"/15" Filter Screen', 2408);
3
4 insert into IJACOB.PRINTER (PRIID, COMPENY, MODEL, SUID)
5 values (101, 'ConAgra', 'Achieve Gold Dual Fan 550Watt PowerSupply.', 2340);
```

```
Evaluation.sql X Employee.sql
C: > Users > cohenmo1 > Downloads > Evaluation.sql
1 insert into IJACOB.EVALUATION (EVID, GRADE, YEAR, EMID)
2 values (500, 5, 2021, 920);
3
4 insert into IJACOB.EVALUATION (EVID, GRADE, YEAR, EMID)
5 values (501, 9, 2023, 1030);
```

```
InstalledOn.sql X Evaluation.sql Employee.sql
C: > Users > cohenmo1 > Downloads > InstalledOn.sql
1 insert into IJACOB.INSTALLEDON (CID, SOID)
2 values (312, 212);
3
4 insert into IJACOB.INSTALLEDON (CID, SOID)
5 values (302, 200);
```

```
Employee.sql X
C: > Users > cohenmo1 > Downloads > Employee.sql
1 insert into IJACOB.EMPLOYEE (EMID, EMNAME, DATEOFBIRTH, ROLEID, TEID, CID, EmPhone, EmEmail)
2 values (900, 'MikiBarrymore', to_date('05-07-2002', 'dd-mm-yyyy'), 150, 5, 315, 0536542816, '
3
4 insert into IJACOB.EMPLOYEE (EMID, EMNAME, DATEOFBIRTH, ROLEID, TEID, CID, EmPhone, EmEmail)
5 values (910, 'VictoriaDupree', to_date('17-09-1976', 'dd-mm-yyyy'), 225, 9, 314, 0523660895,
```


	EMID	EMNAME	DATEOFBIRTH	ROLEID	TEID	CID	EMPHONE	EMEMAIL	ROWID
1	900	MikiBarrymore	05/07/2002	150	5	315	536542816	miki@hfn.com	AACjrZAAGAAAsM0AAA
2	910	VictoriaDupree	17/09/1976	225	9	314	523660895	victoria.dupree@uem.com	AACjrZAAGAAAsM0AAB
3	920	WillemBelles	25/02/1998	150	13	304	51893739	willem.belles@genhisgrill.au	AACjrZAAGAAAsM0AAC
4	930	RosannaFeore	28/11/2002	200	7	311	531457495	rosanna.feore@royalgold.com	AACjrZAAGAAAsM0AAD
5	940	WesDuschel	14/01/1998	425	2	311	514215033	wes.d@ezecastlesoftware.de	AACjrZAAGAAAsM0AAE
6	950	BobbiWeaving	31/03/1994	375	5	307	552292873	bweaving@ntas.com	AACjrZAAGAAAsM0AAF
7	960	JulianaRyder	14/11/1981	425	2	314	544381215	juliana.ryder@loreal.com	AACjrZAAGAAAsM0AAG
8	970	MurrayNoseworthy	09/12/1975	275	11	314	504554326	mnoseworthy@sfgo.com	AACjrZAAGAAAsM0AAH
9	980	PabloShandling	28/07/1989	175	13	306	551336703	pablos@unica.com	AACjrZAAGAAAsM0AAI
10	990	AlecKapanka	09/07/1978	100	3	306	548118428	alec.kapanka@clorox.com	AACjrZAAGAAAsM0AAJ
11	1000	DenzelBradford	23/12/1990	325	5	309	501988384	denzel.bradford@abforms.dk	AACjrZAAGAAAsM0AAK
12	1010	GeneNelligan	18/02/1973	275	13	302	518200503	g.nelligan@staffforce.fr	AACjrZAAGAAAsM0AAL
13	1020	NedStone	17/04/1993	225	7	302	514272506	n.stone@ris.de	AACjrZAAGAAAsM0AAM
14	1030	KaronTinsley	09/11/1985	150	11	314	51643292	ktinsley@wyeth.fr	AACjrZAAGAAAsM0AAN

	EMID	GRADE	YEAR	EMID	ROWID
1	500	5	2021	920	AACKI5AAGAAAGE2AAA
2	501	9	2023	1030	AACKI5AAGAAAGE2AAB
3	502	4	2023	1000	AACKI5AAGAAAGE2AAC
4	503	7	2022	940	AACKI5AAGAAAGE2AAD
5	504	7	2024	920	AACKI5AAGAAAGE2AAE
6	505	9	2022	1000	AACKI5AAGAAAGE2AAF
7	506	4	2023	1010	AACKI5AAGAAAGE2AAG
8	507	10	2020	940	AACKI5AAGAAAGE2AAH
9	508	3	2021	1030	AACKI5AAGAAAGE2AAI
10	509	6	2025	980	AACKI5AAGAAAGE2AAJ
11	510	10	2025	1020	AACKI5AAGAAAGE2AAK
12	511	2	2020	930	AACKI5AAGAAAGE2AAL
13	512	3	2023	1030	AACKI5AAGAAAGE2AAM
14	513	8	2024	1020	AACKI5AAGAAAGE2AAN
15	514	3	2020	930	AACKI5AAGAAAGE2AAO
16	515	10	2021	990	AACKI5AAGAAAGE2AAP
17	516	4	2023	980	AACKI5AAGAAAGE2AAQ
18	517	1	2020	930	AACKI5AAGAAAGE2AAR
19	518	3	2023	970	AACKI5AAGAAAGE2AAS
20	519	3	2023	960	AACKI5AAGAAAGE2AAT

	PRIID	CID	ROWID
1	100	307	AACKI9AAGAAAGEmAAC
2	101	303	AACKI9AAGAAAGEmAAH
3	101	309	AACKI9AAGAAAGEmAAY
4	101	313	AACKI9AAGAAAGEmAAa
5	103	303	AACKI9AAGAAAGEmAAB
6	103	304	AACKI9AAGAAAGEmAAA
7	104	307	AACKI9AAGAAAGEmAAL
8	104	312	AACKI9AAGAAAGEmAAJ
9	104	315	AACKI9AAGAAAGEmAAI
10	105	309	AACKI9AAGAAAGEmAAZ
11	105	313	AACKI9AAGAAAGEmAAR
12	106	303	AACKI9AAGAAAGEmAAS
13	106	306	AACKI9AAGAAAGEmAAP
14	106	307	AACKI9AAGAAAGEmAAT
15	107	302	AACKI9AAGAAAGEmAAO
16	107	310	AACKI9AAGAAAGEmAAU
17	107	311	AACKI9AAGAAAGEmAAX
18	107	312	AACKI9AAGAAAGEmAAE
19	108	310	AACKI9AAGAAAGEmAAF
20	108	315	AACKI9AAGAAAGEmAAK
21	109	304	AACKI9AAGAAAGEmAAN
22	109	308	AACKI9AAGAAAGEmAAQ

	CID	PROCESSOR	RAM	SSD	YEAROFPURCHASE	ROWID
1	300	i3	16	256	2014	AACjrHAAGAAAsMEAAA
2	301	i7	8	512	2018	AACjrHAAGAAAsMEAAAB
3	302	i7	8	256	2022	AACjrHAAGAAAsMEAAAC
4	303	i7	8	512	2017	AACjrHAAGAAAsMEAAD
5	304	i5	32	256	2023	AACjrHAAGAAAsMEAAE
6	305	i9	16	512	2023	AACjrHAAGAAAsMEAAF
7	306	i7	16	256	2017	AACjrHAAGAAAsMEAAG
8	307	i5	16	256	2023	AACjrHAAGAAAsMEAAH
9	308	i3	32	512	2015	AACjrHAAGAAAsMEAAI
10	309	i3	32	128	2016	AACjrHAAGAAAsMEAAJ
11	310	i7	8	256	2016	AACjrHAAGAAAsMEAAK
12	311	i3	8	128	2023	AACjrHAAGAAAsMEAAL
13	312	i5	8	256	2016	AACjrHAAGAAAsMEAAM
14	313	i9	16	128	2019	AACjrHAAGAAAsMEAAN
15	314	i3	16	256	2014	AACjrHAAGAAAsMEAAO
16	315	i5	32	256	2023	AACjrHAAGAAAsMEAAP

Select

```
select duedate
from project
```

	DUEDATE
1	03/03/2022
2	17/01/2021
3	29/01/2020
4	01/05/2021
5	31/07/2024
6	12/03/2022
7	31/01/2025
8	17/03/2020
9	05/01/2023
10	21/06/2021
11	03/11/2020
12	13/04/2023
13	11/11/2021
14	02/10/2021

```
select cid
from computer
```

	CID
1	300
2	301
3	302
4	303
5	304
6	305
7	306
8	307
9	308
10	309
11	310
12	311
13	312
14	313
15	314
16	315

```
select dateofbirth
from employee
```

	DATEOFBIRTH
1	05/07/2002
2	17/09/1976
3	25/02/1998
4	28/11/2002
5	14/01/1998
6	31/03/1994
7	14/11/1981
8	09/12/1975
9	28/07/1989
10	09/07/1978
11	23/12/1990
12	18/02/1973
13	17/04/1993
14	09/11/1985

```
select suemail
from supplier
```

	SUEMAIL
1	martin.crimson@visainternational.de
2	v.loggia@arkidata.com
3	terry@labradanutrition.de
4	russell.gandolfini@waltdisney.th
5	pam.blanchett@abs.com
6	lnakai@ssi.br
7	neneh.a@pib.com
8	dave.latifah@pioneerdatasystems.com
9	markr@tarragonrealty.com
10	laura.chambers@qssgroup.com
11	cevin.serbedzija@ibm.de
12	ahmad.koteas@mls.cz
13	carrie@newmedia.com
14	mikok@spd.com

Delete

SQL	Output	Statistics
<pre>delete from ev109ation where grade < 5 delete from employee where emid = 930 delete from connectedto where priID = 109 delete from installedon where cid = 303</pre>		

Update

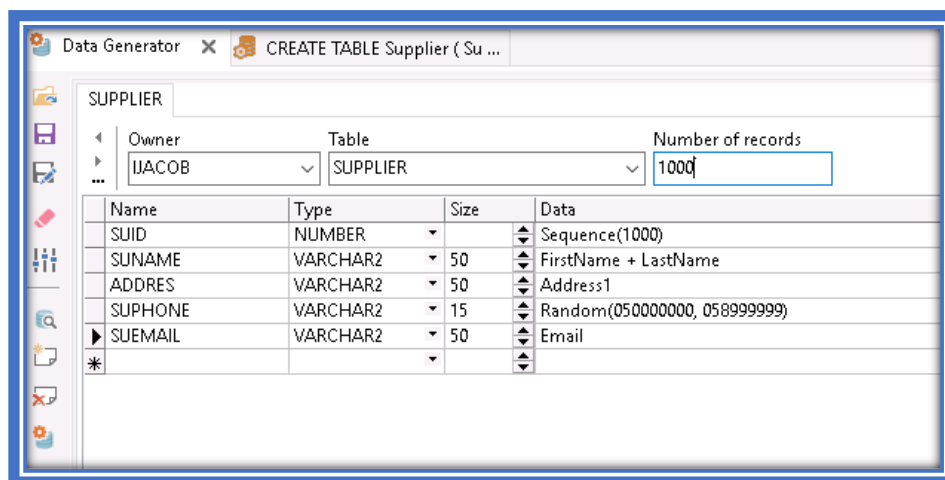
<pre>update evaluation set grade = grade + 1 update project set cost = 30000 where proID = 4 update employee set RoleID = 175 where RoleID = 150 update supplier set suName 'Associated business' where suName = 'Associated business System'</pre>
--



שלב ב'

אכלוס הטבלאות

1. Data Generation - יצירת נתונים



	SUID	SUNAME	ADDRESS	SUPPHONE	SUEMAIL	ROWID
1	1405	RosarioBlair	11 Borgnine Street	50496872	rosario.blair@elite.com	AAClquAAGAAA57rAAA
2	1406	KimberlyHanley	58 Kapanka Ave	50222595	kimberly.hanley@tastefullysimple.com	AAClquAAGAAA57rAAB
3	1407	BethPerez	23 King	50948891	beth@proclarity.com	AAClquAAGAAA57rAAC
4	1408	IsabellaMoss	24 Edie Road	54057580	isabella.moss@caliber.com	AAClquAAGAAA57rAAD
5	1409	BrittanyGarfunkel	48 Merrill Road	53012235	brittany.garfunkel@mai.de	AAClquAAGAAA57rAAE
6	1410	BuffyAbraham	62 Glenn Ave	53016539	buffy.abraham@americanvanguard.uk	AAClquAAGAAA57rAAF
7	1411	GlenFrampton	38 Ferraz vasconcelos Street	54402831	g.frampton@nha.au	AAClquAAGAAA57rAAG
8	1412	LilaMasur	512 Elwes Road	53736251	lila.masur@printcafesoftware.com	AAClquAAGAAA57rAAH
9	1413	ToriWhitwam	72 Marburg Ave	52975571	tori.whitwam@colgatepalmolive.com	AAClquAAGAAA57rAAI
10	1414	LorettaLynne	2 Nicks Street	57905336	loretta.l@wci.de	AAClquAAGAAA57rAAJ
11	1415	EttaFuray	7 Aaron Blvd	52167995	etta.f@dancor.br	AAClquAAGAAA57rAAK
12	1416	RickSmith	24 Mos Road	53444125	rick.smith@csi.de	AAClquAAGAAA57rAAL
13	1417	RoyNivola	25 Peabo Street	56388933	roy.nivola@atlanticnet.nl	AAClquAAGAAA57rAAM
14	1418	EdgarDe Niro	83 Norderstedt Drive	53665431	edgar.deniro@pragmatechsoftware.cy	AAClquAAGAAA57rAAN
15	1419	HerbieReno	60 Brooks Ave	50454416	herbie.reno@esteelauder.uk	AAClquAAGAAA57rAAO
16	1420	JannFreeman	79 Meppel Drive	58741680	jann.freeman@multimedialive.pt	AAClquAAGAAA57rAAP
17	1421	JakeRubinek	77 Askew Drive	58156824	jake.rubinek@lemproducts.uk	AAClquAAGAAA57rAAQ
18	1422	BrendanClark	31 Ponty Road	55482618	brendanc@hotmail.com	AAClquAAGAAA57rAAR
19	1423	JoaquinLevert	91 Wine	52994883	joaquin.levert@printcafesoftware.se	AAClquAAGAAA57rAAS
20	1424	LarnelleDickinson	12 O'Connor Ave	58830957	larnelle.dickinson@fiberlink.fr	AAClquAAGAAA57rAAT
21	1425	SusanRonstadt	598 Gordie Blvd	55907820	susan@volkswagen.ca	AAClquAAGAAA57rAAU
22	1426	JakeMcDormand	14 Keith Drive	58593859	j.mcdormand@bedfordbancshares.se	AAClquAAGAAA57rAAV
23	1427	WalterMetcalfe	229 Englewood Drive	57442734	walter@sis.ca	AAClquAAGAAA57rAAW
24	1428	ElvisShocked	7 Allan	54387671	elvis.shocked@codykramerimports.com	AAClquAAGAAA57rAAX
25	1429	ColeyZellweger	11st Street	56143106	coley.zellweger@capstone.com	AAClquAAGAAA57rAAZ
26	1430	IanMeniketti	810 Taryn Drive	54512540	ian.meniketti@inzone.uk	AAClquAAGAAA57rAAa
27	1431	LievChao	887 Bern Blvd	55152230	liev@horizonorganic.com	AAClquAAGAAA57rAAb
28	1432	AndyCazale	33 Pottendorf Road	54257156	andy.cazale@pearllawgroup.com	AAClquAAGAAA57rAAc
29	1433	FatsStiles	4 Oberwangen Road	56672303	fats@mattel.au	AAClquAAGAAA57rAAd
30	1434	RemyFonda	86 Mindy	52560029	remy.fonda@whitewave.com	AAClquAAGAAA57rAAe

TEAM			
Owner	Table	Number of records	
UACOB	TEAM	1000	
Name	Type	Size	Data
TEID	NUMBER		Sequence(1)
TENAME	VARCHAR2	50	City
ROOMNUMBER	NUMBER		List(select ROOMNUMBER from ROOM)
*			

	TEID	TENAME	ROOMNUMBER	ROWID
1	1	Sparrows Point	760	AAClqdAAGAAA5+bAAA
2	2	Purley	363	AAClqdAAGAAA5+bAAB
3	3	Birkenhead	251	AAClqdAAGAAA5+bAAC
4	4	Roanoke	153	AAClqdAAGAAA5+bAAD
5	5	California	632	AAClqdAAGAAA5+bAAE
6	6	Grand-mere	897	AAClqdAAGAAA5+bAAF
7	7	Algermissen	82	AAClqdAAGAAA5+bAAG
8	8	Sale	327	AAClqdAAGAAA5+bAAH
9	9	Braintree	243	AAClqdAAGAAA5+bAAI
10	10	Niles	808	AAClqdAAGAAA5+bAAJ
11	11	West Chester	294	AAClqdAAGAAA5+bAAK
12	12	Takamatsu	810	AAClqdAAGAAA5+bAAL
13	13	Dubai	293	AAClqdAAGAAA5+bAAM
14	14	King of Prussia	586	AAClqdAAGAAA5+bAAN
15	15	Belgrad	9	AAClqdAAGAAA5+bAAO
16	16	Ittigen	416	AAClqdAAGAAA5+bAAP
17	17	Trieste	639	AAClqdAAGAAA5+bAAQ
18	18	Suberg	406	AAClqdAAGAAA5+bAAR
19	19	Amherst	619	AAClqdAAGAAA5+bAAS
20	20	Brentwood	527	AAClqdAAGAAA5+bAAT
21	21	Burgess Hill	904	AAClqdAAGAAA5+bAAU
22	22	Rorschach	564	AAClqdAAGAAA5+bAAV
23	23	Chapeco	21	AAClqdAAGAAA5+bAAW
24	24	Tokushima	719	AAClqdAAGAAA5+bAAX
25	25	Bad Camberg	120	AAClqdAAGAAA5+bAAY
*				

Owner	Table	Number of records	
JACOB	PROJECT	2000	
Name	Type	Size	Data
PROID	NUMBER		Sequence(1)
PRONAME	VARCHAR2	50	LastName
DUE DATE	DATE		Random(1, 28) + '/' + Random(1, 12) + '/' + Random(2021, 2030)
COST	NUMBER		Random(1000, 20000)
PROFIT	NUMBER		Random(3000, 30000)
TEID	NUMBER		List(select TEID from TEAM)
*			

	PROID	PRONAME	DUEDATE	COST	PROFIT	TEID	ROWID
1	1	Perez	04/09/2244 22:14:26	14071	15245	943	AAClqfAAGAAA5+rAAA
2	2	McGill	18/03/2929 14:36:48	10920	18759	765	AAClqfAAGAAA5+rAAB
3	3	Goldberg	19/02/2486 00:50:15	10101	12262	892	AAClqfAAGAAA5+rAAC
4	4	O'Neill	06/05/2323 20:23:34	11779	26032	233	AAClqfAAGAAA5+rAAD
5	5	Fox	08/09/1986 19:29:16	2583	4627	796	AAClqfAAGAAA5+rAAE
6	6	Kinnear	21/07/2935 22:38:29	12861	15934	510	AAClqfAAGAAA5+rAAF
7	7	Cash	10/01/2309 06:06:11	2140	11091	696	AAClqfAAGAAA5+rAAG
8	8	McNarland	13/06/2727 10:57:30	9400	20647	940	AAClqfAAGAAA5+rAAH
9	9	de Lancie	05/04/2834 12:28:39	1304	27305	837	AAClqfAAGAAA5+rAAI
10	10	Oszajca	18/02/2819 17:36:23	6943	5855	924	AAClqfAAGAAA5+rAAJ
11	11	Hershey	07/09/2734 01:54:46	16366	26368	103	AAClqfAAGAAA5+rAAK
12	12	Sossamon	13/09/2690 15:18:00	6536	4170	818	AAClqfAAGAAA5+rAAL
13	13	Gagnon	23/05/2032 14:56:15	4830	25918	469	AAClqfAAGAAA5+rAAM
14	14	Cusack	26/03/1919 15:55:26	17688	10875	272	AAClqfAAGAAA5+rAAN
15	15	Guest	05/12/2230 20:21:33	4311	29592	576	AAClqfAAGAAA5+rAAO
16	16	Perez	29/08/2786 21:20:00	8221	12219	685	AAClqfAAGAAA5+rAAP
17	17	Bedelia	02/06/2262 12:33:41	4808	23724	550	AAClqfAAGAAA5+rAAQ
18	18	Gunton	08/01/2128 12:18:00	1206	9530	600	AAClqfAAGAAA5+rAAR
19	19	Merchant	14/11/1901 13:39:38	2370	29574	745	AAClqfAAGAAA5+rAAS
20	20	Ledger	21/11/2163 03:28:48	17506	20022	601	AAClqfAAGAAA5+rAAT
21	21	Curtis	10/01/1935 01:56:38	6786	3219	546	AAClqfAAGAAA5+rAAU
22	22	Epps	04/03/2157 10:37:55	15801	16337	825	AAClqfAAGAAA5+rAAV
23	23	Bridges	27/11/2971 05:16:13	3338	18654	503	AAClqfAAGAAA5+rAAW
24	24	Maxwell	03/08/2863 06:50:58	14540	24972	971	AAClqfAAGAAA5+rAAX
25	25	Kramer	01/02/2383 21:02:26	7079	20218	226	AAClqfAAGAAA5+rAAY
*							

EMPLOYEE			
Owner		Table	Number of records
IACOB		EMPLOYEE	23000
Name	Type	Size	Data
EMID	NUMBER		Sequence(111111111)
EMNAME	VARCHAR2	50	FirstName + LastName
DATEOFBIRTH	DATE		Random(1, 28) + '/' + Random(1, 12) + '/' + Random(1960, 2005)
ROLEID	NUMBER		List(select ROLEID from ROKE)
TEID	NUMBER		List(select TEID from TEAM)
CID	NUMBER		List(select CID from COMPUTER)
*			

	EMID	EMNAME	DATEOFBIRTH	ROLEID	TEID	CID	ROWID
1	291134595	Kimberly Danger	13/05/2459 02:33:38	656	75	721	AAClqqAAGAAA5/HAAA
2	291137936	Dylan Hynde	10/07/2658 07:24:14	73	77	920	AAClqqAAGAAA6AxAxA
3	291137937	Kirsten Diehl	06/06/2831 20:56:41	573	267	383	AAClqqAAGAAA6AxAxB
4	291137938	Nina Bancroft	26/03/2667 19:36:20	549	978	219	AAClqqAAGAAA6AxAxC
5	291137939	Tobey Hagerty	07/10/3035 10:26:32	450	767	772	AAClqqAAGAAA6AxAxD
6	291137940	Miles Larter	21/08/2062 12:49:32	453	894	203	AAClqqAAGAAA6AxAxE
7	291137941	Angela Patton	04/01/2651 06:55:43	25	120	499	AAClqqAAGAAA6AxAxF
8	291137942	Steve Lorenz	26/12/2731 11:43:00	540	828	55	AAClqqAAGAAA6AxAxG
9	291137943	Matthew Heche	03/06/2345 03:14:06	973	442	808	AAClqqAAGAAA6AxAxH
10	291137944	Vince Sisto	28/09/2856 22:28:42	642	178	939	AAClqqAAGAAA6AxAxI
11	291137945	Hector Cockburn	25/06/3013 10:55:20	20	86	139	AAClqqAAGAAA6AxAxJ
12	291137946	John Anderson	11/10/2424 02:13:29	609	425	431	AAClqqAAGAAA6AxAxK
13	291137947	Ani Applegate	01/08/3056 10:06:57	874	333	565	AAClqqAAGAAA6AxAxL
14	291137948	Buddy Lawrence	30/10/2862 09:50:24	243	822	963	AAClqqAAGAAA6AxAxM
15	291137949	Marc Fienness	29/03/2719 22:23:05	167	77	234	AAClqqAAGAAA6AxAxN
16	291137950	Will Lynch	21/05/1965 02:46:10	241	76	253	AAClqqAAGAAA6AxAxO
17	291137951	Leonardo Michael	06/09/2545 05:03:41	392	515	798	AAClqqAAGAAA6AxAxP
18	291137952	Owen Blades	22/07/2934 23:56:58	126	45	750	AAClqqAAGAAA6AxAxQ
19	291137953	Warren Purefoy	23/05/3012 02:05:25	839	832	858	AAClqqAAGAAA6AxAxR
20	291137954	Neil Pesci	26/05/2253 19:32:09	189	416	66	AAClqqAAGAAA6AxAxS
21	291137955	Richie Martinez	10/09/2984 14:21:50	533	891	399	AAClqqAAGAAA6AxAxT
22	291137956	Chubby Moody	29/07/2253 03:22:10	425	397	416	AAClqqAAGAAA6AxAxU
23	291137957	Trace Bryson	23/01/2877 08:04:50	174	919	225	AAClqqAAGAAA6AxAxV
24	291137958	Kurtwood Mattea	29/05/2516 20:14:03	470	405	56	AAClqqAAGAAA6AxAxW
25	291137959	Lee Flanagan	15/10/1933 08:34:39	69	194	402	AAClqqAAGAAA6AxAxX
26	291137960	James Van Der Beek	16/03/2023 10:11:00	540	750	710	AAClqqAAGAAA6AxAxY

EVALUATION

Owner	Table	Number of records
UACOB	EVALUATION	1000

Name	Type	Size	Data
EVID	NUMBER		Sequence(1)
GRADE	NUMBER		Random(1, 10)
YEAR1	NUMBER		Random(2016, 2022)
EMID	NUMBER		List(select EMID from EMPLOYEE)

	EVID	GRADE	YEAR1	EMID	ROWID
1	321	10	2017	307134743	AAClqsAAGAAA6EjAAA ...
2	322	4	2017	111137791	AAClqsAAGAAA6EjAAB ...
3	323	1	2020	111138133	AAClqsAAGAAA6EjAAC ...
4	324	2	2017	111136120	AAClqsAAGAAA6EjAAD ...
5	325	9	2020	111111680	AAClqsAAGAAA6EjAAE ...
6	326	1	2016	111137340	AAClqsAAGAAA6EjAAF ...
7	327	5	2021	111115716	AAClqsAAGAAA6EjAAG ...
8	328	1	2017	111113010	AAClqsAAGAAA6EjAAH ...
9	329	7	2018	291134413	AAClqsAAGAAA6EjAAI ...
10	330	5	2020	111111586	AAClqsAAGAAA6EjAAJ ...
11	331	2	2018	111113637	AAClqsAAGAAA6EjAAK ...
12	332	5	2018	111134608	AAClqsAAGAAA6EjAAL ...
13	333	8	2019	111138123	AAClqsAAGAAA6EjAAM ...
14	334	7	2016	291135659	AAClqsAAGAAA6EjAAN ...
15	335	3	2016	111111249	AAClqsAAGAAA6EjAAO ...
16	336	4	2021	111138703	AAClqsAAGAAA6EjAAP ...
17	337	6	2016	291138979	AAClqsAAGAAA6EjAAQ ...
18	338	9	2022	291134577	AAClqsAAGAAA6EjAAR ...
19	339	3	2019	291137532	AAClqsAAGAAA6EjAAS ...
20	340	2	2019	291137214	AAClqsAAGAAA6EjAAT ...
21	341	10	2018	307138275	AAClqsAAGAAA6EjAAU ...
22	342	2	2016	307138877	AAClqsAAGAAA6EjAAV ...
23	343	5	2020	111114787	AAClqsAAGAAA6EjAAW ...
24	344	7	2021	111111698	AAClqsAAGAAA6EjAAX ...
25	345	6	2016	111137468	AAClqsAAGAAA6EjAAY ...

SOFTWARE

Owner

Table

Number of records

UJACOB

SOFTWARE

1000

Name	Type	Size	Data
SOID	NUMBER		Sequence(1)
SONAME	VARCHAR2	50	FirstName
VERSION	VARCHAR2	10	Random(1, 19) + '.' + Random(0, 9)
SUID	NUMBER		List(select SUID from SUPPLIER)
*			

	SOID	SONAME	VERSION	SUID	ROWID
1	1	Stewart	5.4	1443	AAClqiAAGAAA58TAAA
2	2	Oliver	6.7	1575	AAClqiAAGAAA58TAAB
3	3	Karon	5.1	1333	AAClqiAAGAAA58TAAC
4	4	Rhona	9.8	1984	AAClqiAAGAAA58TAAD
5	5	Gene	7.2	1653	AAClqiAAGAAA58TAAE
6	6	Chuck	1.2	1230	AAClqiAAGAAA58TAAF
7	7	Marina	2.9	1241	AAClqiAAGAAA58TAAG
8	8	Amy	15.2	1020	AAClqiAAGAAA58TAAH
9	9	Ceili	13.3	1815	AAClqiAAGAAA58TAAI
10	10	Wayne	16.7	1668	AAClqiAAGAAA58TA AJ
11	11	Raul	1.8	1190	AAClqiAAGAAA58TAAK
12	12	Corey	11.3	1588	AAClqiAAGAAA58TAAL
13	13	Vonda	5.4	1552	AAClqiAAGAAA58TAAM
14	14	Phoebe	3.2	1860	AAClqiAAGAAA58TAAN
15	15	Liam	14.6	1809	AAClqiAAGAAA58TAAO
16	16	Jackson	8.9	1848	AAClqiAAGAAA58TAAP
17	17	Sigourney	9.2	1987	AAClqiAAGAAA58TAAQ
18	18	Loretta	16.5	1888	AAClqiAAGAAA58TAAR
19	19	CeCe	10.5	1701	AAClqiAAGAAA58TAAS
20	20	Murray	3.7	1038	AAClqiAAGAAA58TAAT
21	21	Edgar	11.1	1681	AAClqiAAGAAA58TAAU
22	22	Ramsey	15.1	1883	AAClqiAAGAAA58TAAV
23	23	Sinead	2.9	1370	AAClqiAAGAAA58TAAW
24	24	Kelly	7.0	1414	AAClqiAAGAAA58TAA X
25	25	Miles	6.6	1572	AAClqiAAGAAA58TAA Y
26	26	Lonny	6.5	1960	ΔΔClniΔΔGΔΔΔ58TΔΔ7

INSTALLEDON

Owner	Table	Number of records
IJACOB	INSTALLEDON	1000

Name	Type	Size	Data
CID	NUMBER		List(select CID from COMPUTER)
SOID	NUMBER		List(select SOID from SOFTWARE)
*			

	CID	SOID	ROWID	
1	2	39	AAClqkAAGAAA58jAGf	...
2	3	772	AAClqkAAGAAA58jAIG	...
3	5	280	AAClqkAAGAAA58IACV	...
4	5	309	AAClqkAAGAAA58IACi	...
5	5	471	AAClqkAAGAAA58jAA7	...
6	6	753	AAClqkAAGAAA58IADN	...
7	7	52	AAClqkAAGAAA58jADI	...
8	8	712	AAClqkAAGAAA58jABF	...
9	9	646	AAClqkAAGAAA58IADg	...
10	9	777	AAClqkAAGAAA58jABw	...
11	10	675	AAClqkAAGAAA58jAEn	...
12	10	805	AAClqkAAGAAA58IADR	...
13	11	983	AAClqkAAGAAA58jAGz	...
14	12	105	AAClqkAAGAAA58IAFZ	...
15	12	536	AAClqkAAGAAA58IACH	...
16	15	494	AAClqkAAGAAA58jACS	...
17	15	662	AAClqkAAGAAA58IAGQ	...
18	18	258	AAClqkAAGAAA58jAAx	...
19	25	4	AAClqkAAGAAA58jABR	...
20	25	283	AAClqkAAGAAA58jAEa	...
21	26	641	AAClqkAAGAAA58IAF1	...
22	26	647	AAClqkAAGAAA58IAB/	...
23	26	756	AAClqkAAGAAA58jAEw	...
24	26	991	AAClqkAAGAAA58jAAr	...
25	27	771	AAClqkAAGAAA58jACp	...

PRINTER

Owner: UACOB Table: PRINTER Number of records: 1000

Name	Type	Size	Data
PRIID	NUMBER		Sequence()
COMPENY	VARCHAR2	50	List('HP', 'Canon', 'Epson', 'Brother', 'Lexmark', 'Samsung', 'Xerox', 'Ricoh', 'Konica Minolta', 'Dell', 'Panasonic', 'Kyocera', 'Oki', 'Toshiba', 'Sharp', 'Fuj...
MODEL	VARCHAR2	50	List('OfficeJet Pro 9015e', 'PIXMA TS9120', 'Expression Photo XP-8600', 'HL-L2390DW', 'EcoTank ET-3760', 'Color LaserJet Pro MFP M479fdw', 'imageR...
SUID	NUMBER		List(select SUID from SUPPLIER)

	PRIID	COMPENY	MODEL	SUID	ROWID
1	1	Konica Minolta	KX-MB1520	1421	AACKI7AAGAAAGFDAAA
2	2	Fujitsu	SP C360SFNw	1778	AACKI7AAGAAAGFDAAB
3	3	Fujitsu	KX-MB1520	1511	AACKI7AAGAAAGFDAAC
4	4	Oki	PIXMA TS9120	1349	AACKI7AAGAAAGFDAAD
5	5	Konica Minolta	PIXMA TS9120	1389	AACKI7AAGAAAGFDAAE
6	6	Lexmark	OfficeJet Pro 9015e	1044	AACKI7AAGAAAGFDAAF
7	7	Epson	MX-5141N	1896	AACKI7AAGAAAGFDAAG
8	8	Oki	HL-L2390DW	1081	AACKI7AAGAAAGFDAAH
9	9	Oki	Color LaserJet Pro MFP M479fdw	1065	AACKI7AAGAAAGFDAAI
10	10	Dell	Color LaserJet Pro MFP M479fdw	1612	AACKI7AAGAAAGFDAAJ
11	11	Oki	HL-L2390DW	1584	AACKI7AAGAAAGFDAAK
12	12	Sharp	OfficeJet Pro 9015e	1033	AACKI7AAGAAAGFDAAL
13	13	Samsung	KX-MB1520	1540	AACKI7AAGAAAGFDAAM
14	14	Sharp	OfficeJet Pro 9015e	1388	AACKI7AAGAAAGFDAAN
15	15	Kyocera	KX-MB1520	1304	AACKI7AAGAAAGFDAAO
16	16	Sharp	PIXMA TS9120	1323	AACKI7AAGAAAGFDAAP
17	17	HP	C1760nvw	1164	AACKI7AAGAAAGFDAAQ
18	18	Fujitsu	ScanSnap iX1500	1873	AACKI7AAGAAAGFDAAR
19	19	Kyocera	Color LaserJet Pro MFP M479fdw	1408	AACKI7AAGAAAGFDAAS
20	20	Toshiba	HL-L2390DW	1274	AACKI7AAGAAAGFDAAT
21	21	Sharp	C1760nvw	1505	AACKI7AAGAAAGFDAAU
22	22	Epson	SP C360SFNw	1218	AACKI7AAGAAAGFDAAV
23	23	Toshiba	DocuPrint P365 dw	1961	AACKI7AAGAAAGFDAAW
24	24	Canon	OfficeJet Pro 9015e	1370	AACKI7AAGAAAGFDAAX
25	25	Samsung	imageRUNNER ADVANCE C5235A	1813	AACKI7AAGAAAGFDAAY

CONNECTEDTO

Owner	Table	Number of records
UJACOB	CONNECTEDTO	20000

Name	Type	Size	Data
PRIID	NUMBER		List(select PRIID from PRINTER)
CID	NUMBER		List(select CID from COMPUTER)
*			

		PRIID	CID	ROWID	
▶	1	1	68	AACkI9AAGAAAAGEIACw	...
	2	1	162	AACkI9AAGAAA59eACz	...
	3	1	180	AACkI9AAGAAA59sAGb	...
	4	1	228	AACkI9AAGAAA5 +NAHe	...
	5	1	238	AACkI9AAGAAA59DACo	...
	6	1	297	AACkI9AAGAAA59AAB0	...
	7	1	407	AACkI9AAGAAA59uAD5	...
	8	1	432	AACkI9AAGAAA5 +OAAAL	...
	9	1	466	AACkI9AAGAAA59pABB	...
	10	1	588	AACkI9AAGAAA59vAFh	...
	11	1	589	AACkI9AAGAAA5 +MAHr	...
	12	1	645	AACkI9AAGAAA59DAI0	...
	13	1	670	AACkI9AAGAAAAGEmAG/	...
	14	1	777	AACkI9AAGAAA59GAF8	...
	15	1	822	AACkI9AAGAAA59qAA3	...
	16	1	948	AACkI9AAGAAA59cAEI	...
	17	2	78	AACkI9AAGAAA59uACP	...
	18	2	88	AACkI9AAGAAA59pAGp	...
	19	2	298	AACkI9AAGAAA59bAE8	...
	20	2	302	AACkI9AAGAAA59GAH/	...
	21	2	304	AACkI9AAGAAA59bAFM	...
	22	2	329	AACkI9AAGAAA59aABs	...
	23	2	338	AACkI9AAGAAAAGEnADU	...
	24	2	390	AACkI9AAGAAA5 +OAFo	...
	25	2	456	AACkI9AAGAAA59qAHo	...
	26	2	546	AACkI9AAGAAA59sAGv	...



ODBC Importer .2

	A	B	C	D
	ROOMNUMBER	SPACE	FLOOR	
1	1	27	3	
2	2	79	6	
3	3	95	5	
4	4	81	4	
5	5	56	2	
6	6	44	3	
7	7	87	5	
8	8	25	5	
9	9	94	9	
10	10	31	3	
11	11	56	5	
12	12	55	1	
13	13	25	8	
14	14	38	3	
15	15	49	4	
16	16	100	9	
17	17	40	3	
18	18	93	4	
19	19	12	9	
20	20	19	9	
21	21	81	5	
22	22	19	2	
23	23	33	9	
24	24	46	6	
25	25	82	8	
26	26	54	3	
27	27	13	4	
28	28	83	1	
29	29	63	4	
30	30	90	2	
31	31	62	5	
32	32	26	2	

ODBC Importer X

Data from ODBC Data to Oracle

Connection

User / System DSN: Excel Files

User Name: ijacob

Password: *****

Connect Disconnect

Table / Query

☒ Import Table

'Z:\ROOM.xls'; data\$

☐ Import Query Result

View Data

ODBC Importer

Data from ODBC Data to Oracle

General

Owner: jacob Table: ROOM

Commit every...: 100

☒ Overwrite duplicates ☐ Delete records

☐ Ignore duplicates ☐ Truncate table

Initializing Script: ...

Finalizing Script: ...

Fields

ROOMNUMBER -> ROOMNUMBER

SPACE -> SPACE

FLOOR -> FLOOR

Field: FLOOR

Fieldtype: (Float)

Create SQL

SQL function: ...

additional Oracle processing, for example: substr(%, 1, 20)

Result Preview

ROOMNUMBER	SPACE	FLOOR
25.0	0.0	
98.0	1.0	

Import to Script Close jacob@labdbwin Help

	ROOMNUMBER	SPACE	FLOOR	ROWID	
1	1	27	3	AAAClqbAAGAAA58EAAA	...
2	2	79	6	AAAClqbAAGAAA58EAAB	...
3	3	95	5	AAAClqbAAGAAA58EAAC	...
4	4	81	4	AAAClqbAAGAAA58EAAD	...
5	5	56	2	AAAClqbAAGAAA58EAAE	...
6	6	44	3	AAAClqbAAGAAA58EAAF	...
7	7	87	5	AAAClqbAAGAAA58EAAG	...
8	8	25	5	AAAClqbAAGAAA58EA AH	...
9	9	94	9	AAAClqbAAGAAA58EAAI	...
10	10	31	3	AAAClqbAAGAAA58EAAJ	...
11	11	56	5	AAAClqbAAGAAA58EAAK	...
12	12	55	1	AAAClqbAAGAAA58E AAL	...
13	13	25	8	AAAClqbAAGAAA58E AAM	...
14	14	38	3	AAAClqbAAGAAA58E AAN	...
15	15	49	4	AAAClqbAAGAAA58E AAO	...
16	16	100	9	AAAClqbAAGAAA58E AAP	...
17	17	40	3	AAAClqbAAGAAA58E AAQ	...
18	18	93	4	AAAClqbAAGAAA58E AAR	...
19	19	12	9	AAAClqbAAGAAA58E AAS	...
20	20	19	9	AAAClqbAAGAAA58E AAT	...
21	21	81	5	AAAClqbAAGAAA58E AAU	...
22	22	19	2	AAAClqbAAGAAA58E AAV	...



Text Importer .3

COMPUTER.csv

Data from Textfile Data to Oracle

General

Owner: Table:

Commit every...: ☒ Overwrite duplicates ☐ Delete records ☐ Ignore duplicates ☐ Truncate table

Initializing Script:

Finalizing Script:

Fields

Field1 CID -> CID
Field2 PROCESSOR -> PROCESSOR
Field3 RAM -> RAM
Field4 SSD -> SSD
Field5 YEAROFFPURCHASE -> YEAROFFPURCHASE

Field:
Fieldtype:

SQL function:
additional Oracle processing, for example: substr(%, 1, 20)

Result Preview

CID	PROCESSOR	RAM	SSD	YEAROFFPURCHASE
1	i3	8	256	2022
2	i9	16	512	2019

ijacob@labdbwin - COMPUTER.csv loaded, 18 KB

COMPUTER.csv

Data from Textfile Data to Oracle

File Data

CID, PROCESSOR, RAM, SSD, YEAROFFPURCHASE

1, 13, 8, 256, 2022
2, 19, 16, 512, 2019
3, 13, 8, 128, 2017
4, 17, 8, 1024, 2021
5, 13, 4, 1024, 2017
6, 13, 32, 128, 2018
7, 19, 32, 1024, 2019
8, 10, 32, 64, 2021

Configuration

General

Fieldcount:

☒ End at line-end
☒ Name in header
☒ Skip empty lines

Field Start

☐ Relative position ☐ Absolute position ☐ Character

Field End

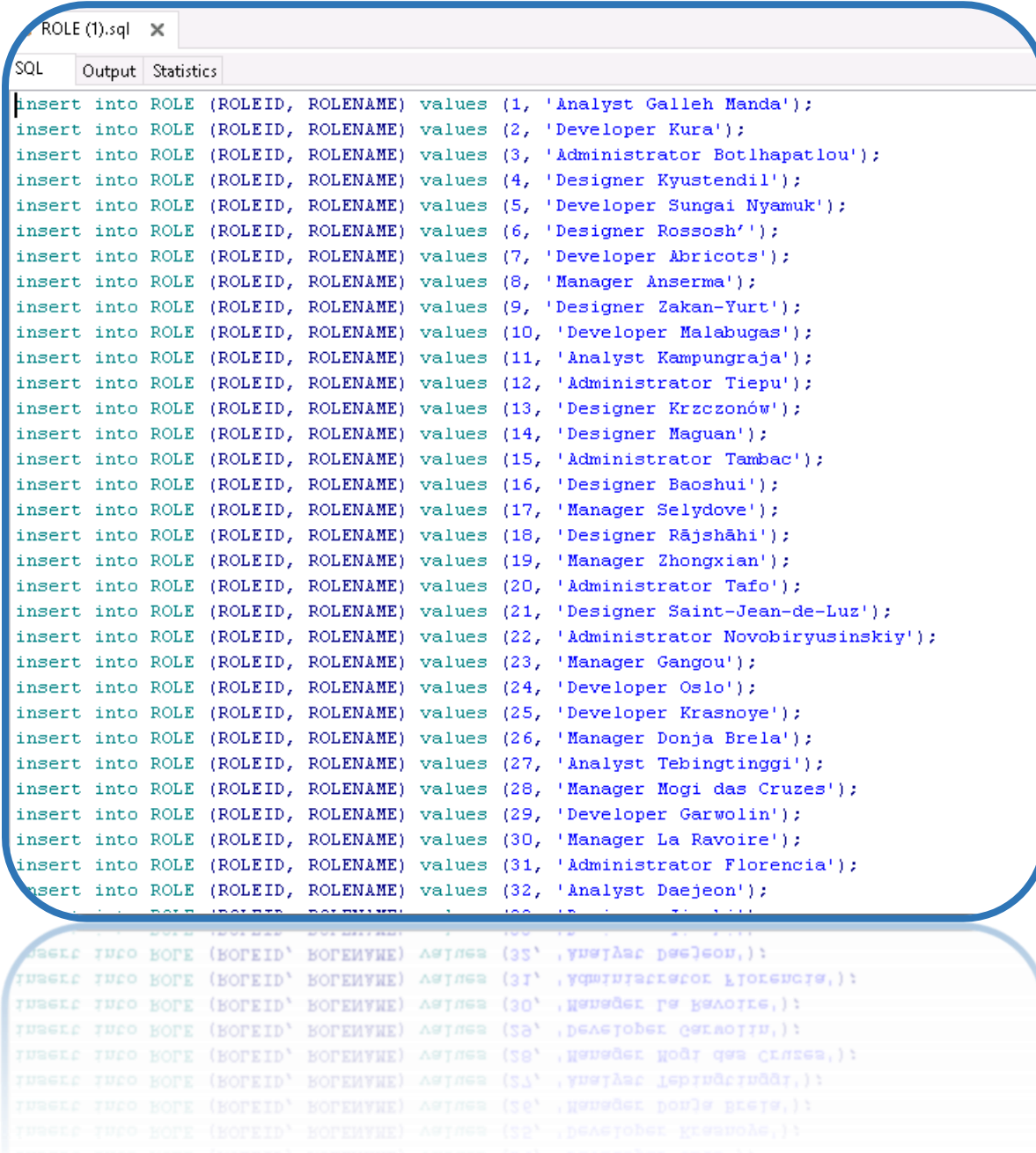
☐ Length ☐ Character

Result Preview

CID	PROCESSOR	RAM	SSD	YEAROFFPURCHASE
1	i3	8	256	2022
2	i9	16	512	2019

ijacob@labdbwin - COMPUTER.csv loaded, 18 KB

	CID	PROCESSOR	RAM	SSD	YEAROFPURCHASE	ROWID
1	636	i7	32	256	2020	AACjrhAAGAAA\$MDAAA ...
2	637	i3	4	1024	2021	AACjrhAAGAAA\$MDAAB ...
3	638	i7	8	128	2022	AACjrhAAGAAA\$MDAAC ...
4	639	i9	16	128	2016	AACjrhAAGAAA\$MDAAD ...
5	640	i9	32	128	2017	AACjrhAAGAAA\$MDAAE ...
6	641	i9	32	1024	2016	AACjrhAAGAAA\$MDAAF ...
7	642	i3	16	128	2022	AACjrhAAGAAA\$MDAAG ...
8	643	i7	4	512	2021	AACjrhAAGAAA\$MDAAH ...
9	644	i3	32	128	2022	AACjrhAAGAAA\$MDAAI ...
10	645	i3	4	512	2018	AACjrhAAGAAA\$MDAAJ ...
11	646	i7	16	1024	2022	AACjrhAAGAAA\$MDAAK ...
12	647	i5	4	128	2016	AACjrhAAGAAA\$MDAAL ...
13	648	i7	8	128	2016	AACjrhAAGAAA\$MDAAM ...
14	649	i3	32	128	2018	AACjrhAAGAAA\$MDAAN ...
15	650	i9	32	1024	2020	AACjrhAAGAAA\$MDAAO ...
16	651	i5	32	256	2017	AACjrhAAGAAA\$MDAAP ...
17	652	i9	4	512	2020	AACjrhAAGAAA\$MDAAQ ...
18	653	i5	32	128	2022	AACjrhAAGAAA\$MDAAR ...
19	654	i5	16	512	2023	AACjrhAAGAAA\$MDAAS ...
20	655	i5	4	128	2023	AACjrhAAGAAA\$MDAAT ...
21	656	i5	16	256	2016	AACjrhAAGAAA\$MDAAU ...
22	657	i7	4	128	2020	AACjrhAAGAAA\$MDAAV ...



	ROLEID	ROLENAME	ROWID
▶ 1	514	Administrator Xingcheng	... AAClqnAAGAAA57bAAA ...
2	515	Administrator Sendafa	... AAClqnAAGAAA57bAAB ...
3	516	Analyst Penghu	... AAClqnAAGAAA57bAAC ...
4	517	Designer Kanzaki	... AAClqnAAGAAA57bAAD ...
5	518	Manager Pangani	... AAClqnAAGAAA57bAAE ...
6	519	Designer Toguchin	... AAClqnAAGAAA57bAAF ...
7	520	Developer Concepci?n del Uruguay	... AAClqnAAGAAA57bAAG ...
8	521	Administrator Cincinnati	... AAClqnAAGAAA57bAAH ...
9	522	Administrator Zuocun	... AAClqnAAGAAA57bAAI ...
10	523	Analyst Pagaru?a	... AAClqnAAGAAA57bAAJ ...
11	524	Designer Yujiawu	... AAClqnAAGAAA57bAAK ...
12	525	Developer Leicheng	... AAClqnAAGAAA57bAAL ...
13	526	Developer Ban?	... AAClqnAAGAAA57bAAM ...
14	527	Designer Mchinji	... AAClqnAAGAAA57bAAN ...
15	528	Analyst Krebetkrajan	... AAClqnAAGAAA57bAAO ...
16	529	Analyst Rychvald	... AAClqnAAGAAA57bAAP ...
17	530	Administrator Kula	... AAClqnAAGAAA57bAAQ ...
18	531	Manager Negreiros	... AAClqnAAGAAA57bAAR ...
19	532	Manager Gengzhuang	... AAClqnAAGAAA57bAAS ...
20	533	Administrator Angra dos Reis	... AAClqnAAGAAA57bAAT ...
21	534	Analyst Lamalewar	... AAClqnAAGAAA57bAAU ...
22	535	Designer Tabaga	... AAClqnAAGAAA57bAAV ...
23	536	Analyst Dakoro	... AAClqnAAGAAA57bAAW ...
24	537	Administrator Fenglu	... AAClqnAAGAAA57bAAX ...
25	538	Manager Makoua	... AAClqnAAGAAA57bAAY ...
*			...

שאלות ראשונה – החלפת מדפסות

תיאור מילולי: רוצים להחליף את המדפסות בקומה 3 ורוצים לדעת כמה מדפסות סיפק כל ספק

הקוד:

```
SELECT Supplier.SuName, COUNT(ConnectedTo.PriID) as PrinterCount
FROM Supplier
JOIN Printer ON Supplier.SuID = Printer.SuID
JOIN ConnectedTo ON Printer.PriID = ConnectedTo.PriID
JOIN Computer ON ConnectedTo.CID = Computer.CID
JOIN Employee ON Computer.CID = Employee.CID
JOIN Team ON Employee.TeID = Team.TeID
JOIN Room ON Team.RoomNumber = Room.RoomNumber
WHERE Room.Floor = 3
GROUP BY Supplier.SuName
```

תוצאות וזמן ריצה:

	SURNAME	PRINTERCOUNT
1	Frances Rebhorn	63
2	Curtis Garber	54
3	Laurence Coburn	154
4	Roscoe Rydell	53
5	Oro Logue	87
6	Kylie Kimball	74
7	Fats Armatrading	98
8	Forest Mohr	51
9	Mint Tankard	98
10	Anthony Suvari	73
11	Cameron Van Damme	39
12	Nathan Ininn	44

624 rows selected in 21.694 seconds

תיאור מילולי: יש צוותים בקומה 2 שלא כל כך מצליחים ורוצים למזג אותם עם צוותים טובים ולכן מחפשים צוותים שיש בהם לפחות 2 עובדים שיש להם הערכה עם ציון מעל 6 והצוות יושב בקומה 2

הקוד:

```
SELECT Project.ProID, Project.ProName
FROM Project
INNER JOIN Team ON Project.TeID = Team.TeID
INNER JOIN Employee ON Team.TeID = Employee.TeID
INNER JOIN Evaluation ON Employee.EmID = Evaluation.EmID
WHERE Employee.TeID = Team.TeID
      AND Evaluation.Grade > 6
      AND Project.TeID IN (
        SELECT TeID
        FROM Employee
        INNER JOIN Evaluation ON Employee.EmID = Evaluation.EmID
        WHERE Evaluation.Grade > 6
        GROUP BY TeID
        HAVING COUNT(DISTINCT Employee.EmID) >= 2
      )
GROUP BY Project.ProID, Project.ProName
HAVING COUNT(DISTINCT Employee.EmID) >= 2;
```

התוצאות וזמן הריצה:

SQL		
Output		
Statistics		
<pre>SELECT project.proid, project.proname FROM project INNER JOIN team ON project.teid = team.teid INNER JOIN employee ON employee.teid = team.teid INNER JOIN evaluation ON employee.emid = evaluation.emid WHERE evaluation.grade > 6 AND project.teid IN (SELECT teid FROM employee INNER JOIN evaluation ON employee.emid = evaluation.emid WHERE evaluation.grade > 6 GROUP BY teid HAVING COUNT(DISTINCT evaluation.emid) >= 2)</pre>		
	PROID	PRONAME
1	1424	Downie
2	1274	Lerner
3	1224	Uggams
4	1869	Redford
5	1802	Neville
6	269	Wells
7	1757	Dzundza
8	1585	Eckhart
9	367	Blige
10	863	Redford
11	1689	Lyonne
12	1782	Mohr
13	1042	Gibson

138 rows selected in 1.014 seconds

תיאור מילולי: רוצים להחליף מחשבים איטיים (RAM 4) עבור עובדים שנמצאים בצוותים מצטיינים שהביאו רווחים של מעל 10000 לחברה ולנתק את הישנים מהמדפסות שלהם.

דקוד:

```
SQL Output Statistics
SELECT c.cid
FROM Computer c
JOIN Employee e ON e.cid = c.cid
JOIN Team t ON e.teid = t.teid
JOIN Project p ON p.teid = t.teid
JOIN Connectedto con ON c.cid = con.cid
WHERE c.ram = 4
group by c.cid, e.emid, e.emname
having sum(p.profit) > 10000;
```

תוצאות זמן ריצה:

The screenshot shows the SQL Developer interface. The SQL window contains the following query:

```
SELECT c.cid
FROM computer c
JOIN employee e ON e.cid = c.cid
JOIN team t ON e.teid = t.teid
JOIN project p ON p.teid = t.teid
JOIN connectedto con ON c.cid = con.cid
WHERE c.ram = 4
GROUP BY c.cid, e.emid, e.emname
HAVING SUM(p.profit) > 10000;
```

The Results window displays the following data:

CID
1
2
3
4
5
6
7
8
9
10
11
12
13
14

The status bar at the bottom indicates: 180 rows selected in 1.235 seconds.

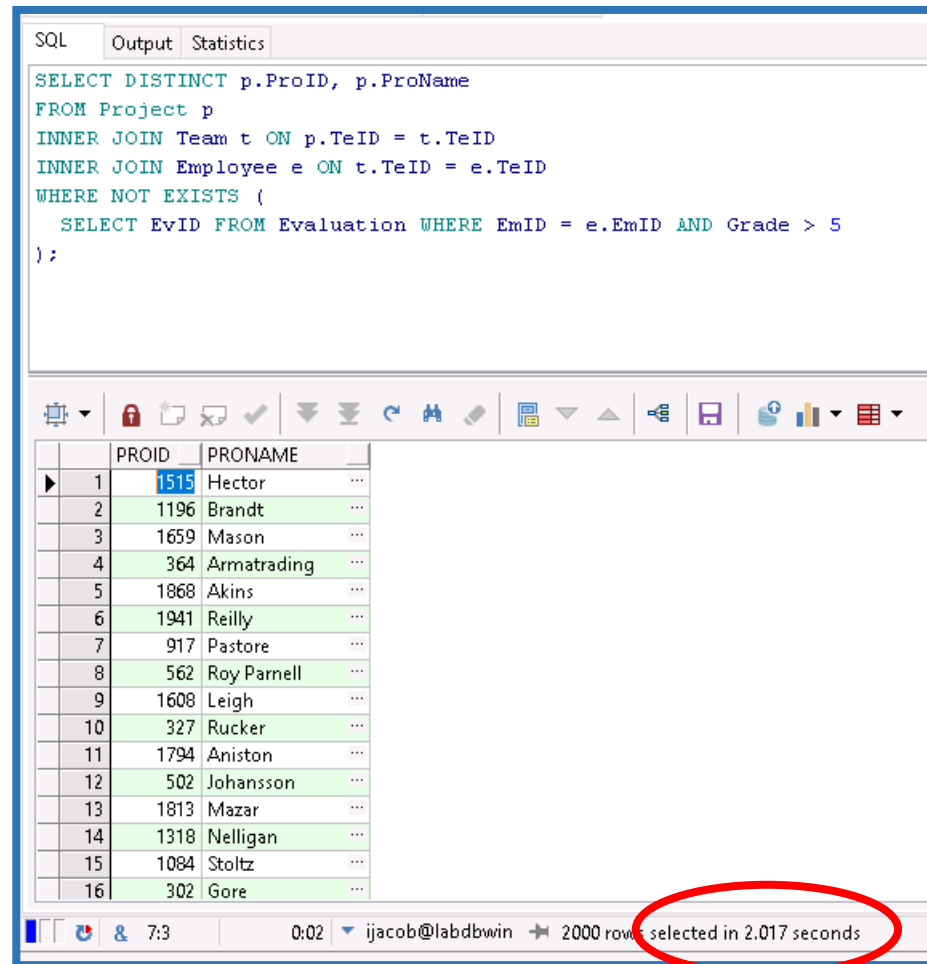
שאילתה רביעית – תמונת מצב

תיאור מילולי: רוצים לקבל את כל הפרויקטים שיש בצוות שלהם עובדים חלשים (אף פעם לא קיבלו הערכה של מעל 5)

הקוד:

```
SELECT DISTINCT p.ProID, p.ProName
FROM Project p
INNER JOIN Team t ON p.TeID = t.TeID
INNER JOIN Employee e ON t.TeID = e.TeID
WHERE NOT EXISTS (
    SELECT EvID FROM Evaluation WHERE EmID = e.EmID AND Grade > 5
);
```

תוצאות וזמן ריצה:



SQL Output Statistics

```
SELECT DISTINCT p.ProID, p.ProName
FROM Project p
INNER JOIN Team t ON p.TeID = t.TeID
INNER JOIN Employee e ON t.TeID = e.TeID
WHERE NOT EXISTS (
    SELECT EvID FROM Evaluation WHERE EmID = e.EmID AND Grade > 5
);
```

	PROID	PRONAME	
1	1515	Hector	...
2	1196	Brandt	...
3	1659	Mason	...
4	364	Armatrading	...
5	1868	Akins	...
6	1941	Reilly	...
7	917	Pastore	...
8	562	Roy Parnell	...
9	1608	Leigh	...
10	327	Rucker	...
11	1794	Aniston	...
12	502	Johansson	...
13	1813	Mazar	...
14	1318	Nelligan	...
15	1084	Stoltz	...
16	302	Gore	...

7:3 0:02 ijacob@labdbwin 2000 rows selected in 2.017 seconds

שאלת חמישית – הספקים המובילים

תיאור מילולי: רוצים למצוא את כל הספקים שמספקים מוצרים טובים – מוצרים שמותקנים או מחוברים ל-10 מחשבים שהעובד שהמחשב שייך לו קיבל לפחות פעם אחת ציון של מעל 7

הקוד:

```
SELECT DISTINCT s.SuID, s.SuName
FROM Supplier s
JOIN Printer p ON s.SuID = p.SuID
JOIN ConnectedTo c ON p.PriID = c.PriID
JOIN Computer m ON c.CID = m.CID
JOIN InstalledOn i ON m.CID = i.CID
JOIN Software o ON i.SoId = o.SoId
JOIN Employee e ON m.CID = e.CID
JOIN Evaluation v ON e.EmID = v.EmID
WHERE (p.PriID IN (
    SELECT PriID
    FROM ConnectedTo
    GROUP BY PriID
    HAVING COUNT(DISTINCT CID) > 10
) OR o.SoId IN (
    SELECT SoId
    FROM InstalledOn
    GROUP BY SoId
    HAVING COUNT(DISTINCT CID) > 10
)) AND e.EmID IN (
    SELECT EmID
    FROM Employee
    WHERE CID = m.CID
) AND v.Grade > 7
```

תוצאות וזמן ריצה:

	SUID	SUNAME
1	4435	Donal Kilmer
2	4520	Frederic Ruffalo
3	4492	Ricardo Wincott
4	4362	Murray Dzundza
5	4288	Derek Matheson
6	4168	Jean Margolyes
7	4799	Lee Ammons
8	4962	Kyle Stevens
9	4587	Daryl Collette
10	4843	Tobey Cocker
11	4490	Ritchie Bush
12	4120	Denzel Camp
13	4473	Balthazar Walsh
14	4557	Mira Henstridge
15	4352	Crispin Allison
16	4688	Emily Johnson
17	4370	Gloria Marie

58 rows selected in 23.422 seconds

שאילתה שישית – מיון הפרויקטים

תיאור מילולי: רוצים למיין את הפרויקטים לפי טיב העובדים שעובדים עליו. נבקש את כל הפרויקטים עם הממוצע הערכות של העובדים של הצוות שמשויך לפרויקט הספציפי.

הקוד:

```
SELECT Project.ProName, Project.DueDate, Team.TeName, AVG(Evaluation.Grade) AS AverageGrade
FROM Project
INNER JOIN Team ON Project.TeID = Team.TeID
INNER JOIN Employee ON Team.TeID = Employee.TeID
INNER JOIN Evaluation ON Employee.EmID = Evaluation.EmID
GROUP BY Project.ProName, Project.DueDate, Team.TeName
ORDER BY AverageGrade DESC;
```

התוצאות וזמן הריצה:

SQL					
Output					
Statistics					
<pre>SELECT Project.ProName, Project.DueDate, Team.TeName, AVG(Evaluation.Grade) AS FROM Project INNER JOIN Team ON Project.TeID = Team.TeID INNER JOIN Employee ON Team.TeID = Employee.TeID INNER JOIN Evaluation ON Employee.EmID = Evaluation.EmID GROUP BY Project.ProName, Project.DueDate, Team.TeName ORDER BY AverageGrade DESC;</pre>					
	PRONAME	DUEDATE	TENAME	AVERAGEGRADE	
1	Hewitt	04/08/2023	Webster Groves	10	
2	Kweller	13/05/2028	Delafield	10	
3	Secada	16/12/2027	W?rth	10	
4	Lever	11/03/2018	Boston	10	
5	Sharp	10/02/2021	Cherepovets	10	
6	Curtis	22/10/2029	Schaumburg	10	
7	Shearer	28/08/2027	Herdecke	10	
8	Silverman	03/07/2026	Chambersburg	10	
9	Gooding	19/07/2026	Newcastle upon Tyne	10	
10	Coe	07/03/2030	Hilversum	10	
11	Rauhofer	06/06/2017	Harahan	10	
12	Coburn	18/06/2025	Visselh?vede	10	
13	Campbell	25/01/2016	Olshuf	10	
1:40 0:01 ijacob@labdbwin 1271 rows selected in 1.168 seconds					

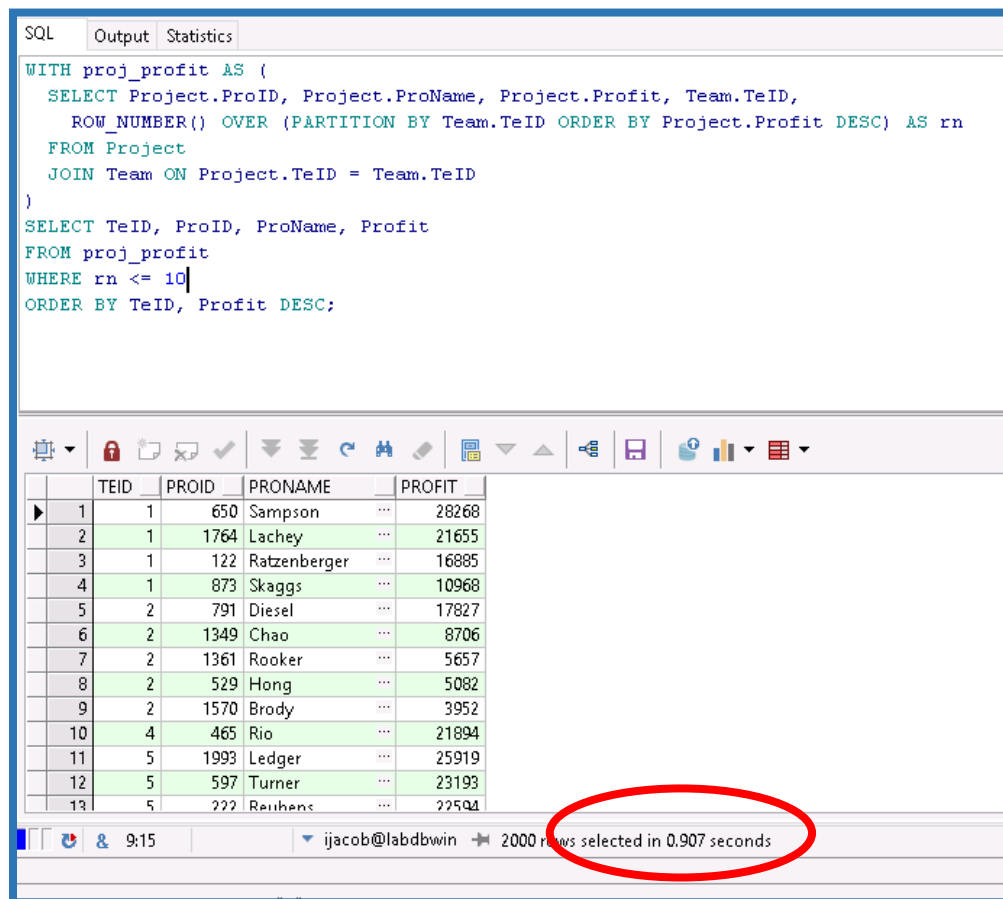
שאלתה שביעית – הפרויקטים הרווחיים

תיאור מילולי: רוצים למצוא את 10 הפרויקטים הרווחיים של כל צוות כדי לקבל אינדיקציה על התועלת שהחברה מקבלת מכל צוות.

הקוד:

```
WITH proj_profit AS (  
  SELECT Project.ProID, Project.ProName, Project.Profit, Team.TeID,  
         ROW_NUMBER() OVER (PARTITION BY Team.TeID ORDER BY Project.Profit DESC) AS rn  
  FROM Project  
  JOIN Team ON Project.TeID = Team.TeID  
)  
SELECT TeID, ProID, ProName, Profit  
FROM proj_profit  
WHERE rn <= 10  
ORDER BY TeID, Profit DESC;
```

התוצאות וזמן ריצה:



	TEID	PROID	PRONAME	PROFIT
1	1	650	Sampson	28268
2	1	1764	Lachey	21655
3	1	122	Ratzenberger	16885
4	1	873	Skaggs	10968
5	2	791	Diesel	17827
6	2	1349	Chao	8706
7	2	1361	Rooker	5657
8	2	529	Hong	5082
9	2	1570	Brody	3952
10	4	465	Rio	21894
11	5	1993	Ledger	25919
12	5	597	Turner	23193
13	5	222	Reithens	22504

ijacob@labdbwin 2000 rows selected in 0.907 seconds

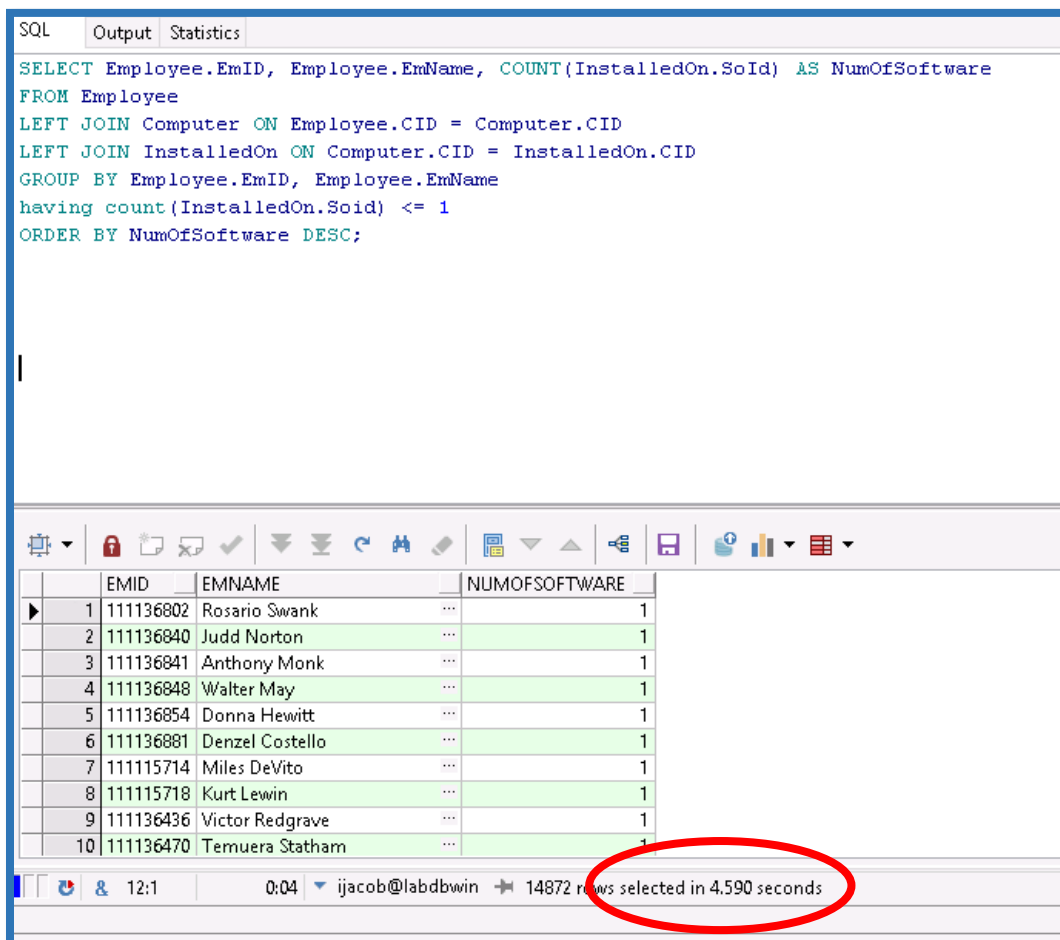
שאלתה שמינית – הדרכת שימוש בתוכנות

תואר מילולי: רוצים לדעת מיהם העובדים שעל המחשב שלהם מותקנים מקסימום תוכנה אחת, כדי להעביר להם הדרכה בעיין שימוש בתוכנות.

הקוד:

```
SELECT Employee.EmID, Employee.EmName, COUNT(InstalledOn.SoId) AS NumOfSoftware
FROM Employee
LEFT JOIN Computer ON Employee.CID = Computer.CID
LEFT JOIN InstalledOn ON Computer.CID = InstalledOn.CID
GROUP BY Employee.EmID, Employee.EmName
having count(InstalledOn.Soid) <= 1
ORDER BY NumOfSoftware DESC;
```

התוצאות וזמן ריצה:

SQL	Output	Statistics
<pre>SELECT Employee.EmID, Employee.EmName, COUNT(InstalledOn.SoId) AS NumOfSoftware FROM Employee LEFT JOIN Computer ON Employee.CID = Computer.CID LEFT JOIN InstalledOn ON Computer.CID = InstalledOn.CID GROUP BY Employee.EmID, Employee.EmName having count(InstalledOn.Soid) <= 1 ORDER BY NumOfSoftware DESC;</pre>		
		
EMID	EMNAME	NUMOFSOFTWARE
1	111136802 Rosario Swank	1
2	111136840 Judd Norton	1
3	111136841 Anthony Monk	1
4	111136848 Walter May	1
5	111136854 Donna Hewitt	1
6	111136881 Denzel Costello	1
7	111115714 Miles DeVito	1
8	111115718 Kurt Lewin	1
9	111136436 Victor Redgrave	1
10	111136470 Temuera Statham	1

14872 rows selected in 4.590 seconds

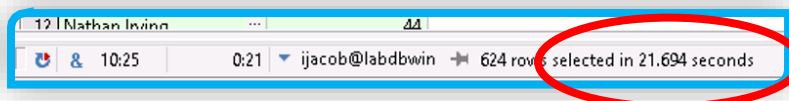
אינדקסים

הסבר קצר מה זה אינדקסים: אינדקס הוא שיטת כוונון ביצועים המאפשרת שליפה מהירה יותר של רשומות. אינדקס יוצר ערך עבור כל ערך המופיע בעמודות האינדקס.

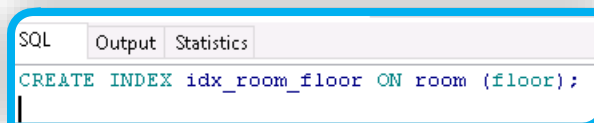
אינדקסים משפרים את זמן הריצה

נשפר את הביצוע של השאילתה הראשונה – החלפת מדפסות

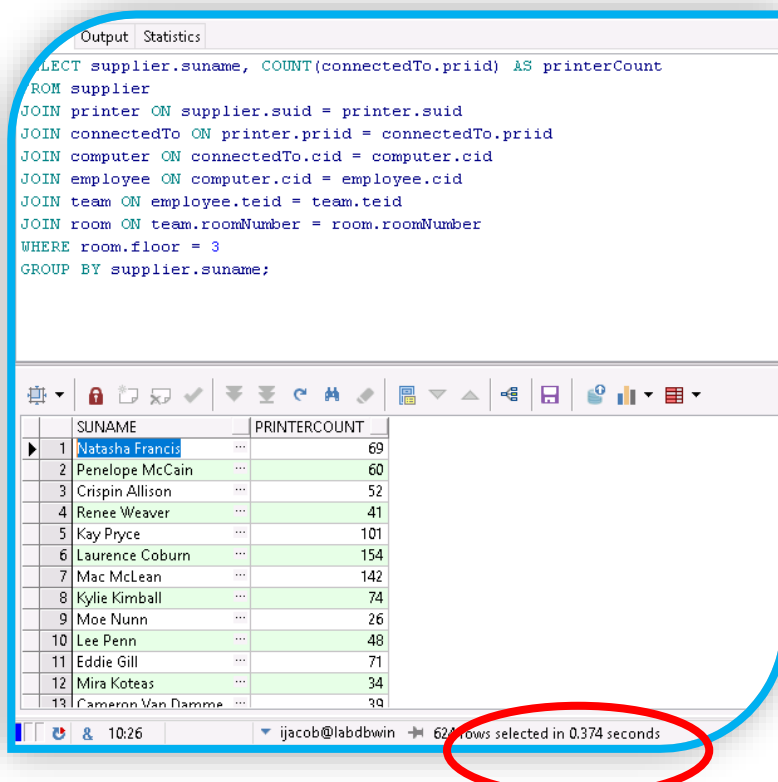
לפני השינוי:



נוסיף אינדקס שמחלק את הטבלה ROOM לפי קומות:



אחרי השינוי:



נשפר את הביצוע של השאילתה השנייה – איזון בין הצוותים

לפני השינוי:



נוסיף אינדקס שמחלק את הטבלה EVALUATION לפי ציונים של כל עובד:

```
CREATE INDEX idx_evaluation_emid_grade ON Evaluation (Emid, Grade);
```

אחרי השינוי:

A screenshot of a database client showing the execution of a query. The query is displayed in the 'Output' tab, and the results are shown in a table below. The status bar at the bottom indicates '138 rows selected in 0.112 seconds', which is circled in red.

The SQL query is:

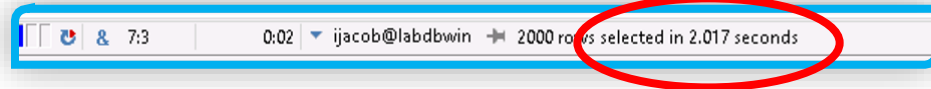
```
SELECT project.proid, project.proname
FROM project
INNER JOIN team ON project.teid = team.teid
INNER JOIN employee ON employee.teid = team.teid
INNER JOIN evaluation ON employee.emid = evaluation.emid
WHERE evaluation.grade > 6
AND project.teid IN (
  SELECT teid
  FROM employee
  INNER JOIN evaluation ON employee.emid = evaluation.emid
  WHERE evaluation.grade > 6
  GROUP BY teid
  HAVING COUNT(DISTINCT evaluation.emid) >= 2
)
```

The resulting table has two columns: PROID and PRONAME. The data is as follows:

	PROID	PRONAME
1	1424	Downie
2	1274	Lerner
3	1224	Uggams
4	1869	Redford
5	1802	Neville
6	269	Wells
7	1757	Dzundza
8	1585	Eckhart
9	367	Blige
10	863	Redford
11	1689	Lyonne
12	1782	Mohr

נשפר את הביצועים של השאילתה הרביעית – תמונת מצב

לפני השינוי:



נוסיף אינדקס שמחלק את הטבלה EVALUATION לפי ציונים של כל עובד:

```
CREATE INDEX idx_evaluation_emid_grade ON Evaluation (Emid, Grade);
```

אחרי השינוי:

A screenshot of a database query execution status bar. The bar is blue and white. It shows the user 'ijacob@labdbwin' and the time '0:02'. The status '2000 rows selected in 0.632 seconds' is highlighted with a red circle.

Output Statistics

```
SELECT DISTINCT p.projid, p.proname  
FROM project p  
INNER JOIN team t ON p.teid = t.teid  
INNER JOIN employee e ON e.teid = t.teid  
WHERE NOT EXISTS(  
SELECT evid FROM evaluation WHERE emid = e.emid AND grade > 5  
);
```

	PROID	PRONAME
1	1515	Hector
2	1196	Brandt
3	1659	Mason
4	364	Armatrading
5	1868	Akins
6	1941	Reilly
7	917	Pastore
8	562	Roy Parnell
9	1608	Leigh
10	327	Rucker
11	1794	Aniston
12	502	Johansson
13	1813	Mazar
14	1318	Nelligan
15	1084	Stoltz
16	302	Gore

נשפר את הביצוע של השאילתה החמישית – הספקים המובילים

לפני שינוי:



נוסיף אינדקס שמחלק את הטבלה PRINTER לפי ספק ומדפסת:

```
CREATE INDEX idx_printer_suid_priid ON printer (suid, priid);
```

אחרי שינוי:

A screenshot of a database query execution bar. The bar is blue with a white background. It shows the user 'ijacob@labdbwin' and the time '0:21'. The status bar at the bottom indicates '588 rows selected in 21.247 seconds'. The text '588 rows selected in 21.247 seconds' is circled in red.

The query text is as follows:

```
SELECT DISTINCT s.suid, s.suname
FROM supplier s
JOIN printer p ON s.suid = p.suid
JOIN connectedTo c ON c.priid = p.priid
JOIN computer m ON c.cid = m.cid
JOIN installedON i ON m.cid = i.cid
JOIN software o ON i.soid = o.soid
JOIN employee e ON e.cid = m.cid
JOIN evaluation v ON e.emid = v.emid
WHERE (p.priid IN (
SELECT priid
FROM connectedTo
GROUP BY priid
HAVING COUNT(DISTINCT cid) > 10
))
```

	SUID	SUNAME
1	4435	Donal Kilmer
2	4520	Frederic Ruffalo
3	4492	Ricardo Wincott
4	4362	Murray Dzundza
5	4288	Derek Matheson
6	4168	Jean Margolyes
7	4799	Lee Ammons
8	4962	Kyle Stevens
9	4587	Daryl Collette
10	4843	Tobey Cocker
11	4490	Ritchie Bush
12	4120	Denzel Camp

אינדקסים לא מועילים

אינדקס לא מועיל בשאילתה השלישית – שדרוג מחשבים

לפני שינוי:



נוסיף אינדקס שמחלק את הטבלה COMPUTER לפי מספר מזהה ומעבד:

```
CREATE INDEX idx_computer_cid ON Computer(cid, Processor);
```

אחרי שינוי:

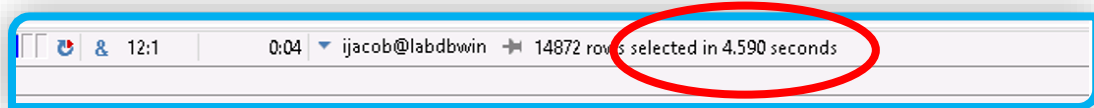
A screenshot of a database query execution window. The window has two tabs: 'Output' and 'Statistics'. The 'Output' tab is active, showing a SQL query. Below the query is a toolbar with various icons. Below the toolbar is a table with 14 rows and 2 columns: 'CID' and an unnamed column. The table contains the following data:

	CID
1	356
2	218
3	316
4	509
5	66
6	75
7	337
8	645
9	540
10	797
11	84
12	59
13	913
14	60

At the bottom of the window, a status bar shows a timer at 0:01, a user 'ijacob@labdbwin', and a message '4480 rows selected in 1.240 seconds'. The text '4480 rows selected in 1.240 seconds' is circled in red.

אינדקס לא מועיל בשאילתה השמינית – הדרכת שימוש בתוכנות

לפני שינוי:



נוסיף אינדקס שמחלק את הטבלה EMPLOYEE לפי ת.ז ושם:

```
CREATE INDEX idx_Employee_emid_emname ON EMPLOYEE(Emid, Emname)
```

אחרי שינוי:

A screenshot of a database query execution status bar. The bar is light blue with a darker blue border. It contains the following text: '8:1', '0:04', 'ijacob@labdbwin', and '14872 rows selected in 4.609 seconds'. The text '14872 rows selected in 4.609 seconds' is circled in red.

Below the status bar, there is a table with the following data:

	EMID	EMNAME	NUMOFSOFTWARE
1	111136802	Rosario Swank	1
2	111136840	Judd Norton	1
3	111136841	Anthony Monk	1
4	111136848	Walter May	1
5	111136854	Donna Hewitt	1
6	111136881	Denzel Costello	1
7	111115714	Miles DeVito	1
8	111115718	Kurt Lewin	1
9	111136436	Victor Redgrave	1
10	111136470	Temuera Statham	1
11	111136476	Alessandro Pantoliano	1
12	111136495	Burt Claxton	1



שלב ג'

שמירת הפרויקט ב-GitHub

The screenshot shows the GitHub repository page for 'israeljacob / project-in-DB'. The repository is private and has 47 commits. The commit history shows a sequence of commits: 'Stage 1' (fixes), 'Stage 2' (finished stage 2), and 'README.md' (Initial commit). The 'README.md' file is highlighted with a red circle and labeled 'project-in-DB'.

israeljacob / project-in-DB Private

Unwatch 1 Fork 0 Star 0

Code Issues Pull requests Actions Projects Security Insights Settings

main 1 branch 0 tags

Go to file Add file Code

About

No description, website, or topics provided.

Readme

0 stars

1 watching

0 forks

Releases

No releases published

Create a new release

Packages

No packages published

Publish your first package

israeljacob finished stage 2 5ea99be 2 weeks ago 47 commits

Stage 1 fixes 2 weeks ago

Stage 2 finished stage 2 2 weeks ago

README.md Initial commit 2 months ago

README.md

project-in-DB

The screenshot shows the GitHub repository page for 'israeljacob / project-in-DB' with the 'Code' tab selected. The commit history is displayed, showing a sequence of commits: 'Stage 1' (fixes), 'Stage 2' (finished stage 2), and 'README.md' (Initial commit). The 'Stage 1' commit is highlighted with a red circle and labeled 'project-in-DB / Stage 1 /'.

israeljacob / project-in-DB Private

Unwatch 1 Fork 0 Star 0

Code Issues Pull requests Actions Projects Security Insights Settings

Code

main + Q

Go to file

Stage 1

Computer.sql

ConnectedTo.sql

DBCreate.sql

Employee.sql

Evaluation.sql

InstalledOn.sql

MYERD.png

MYLPLSQLDSD.jpg

MyERDDSD.png

Printer.sql

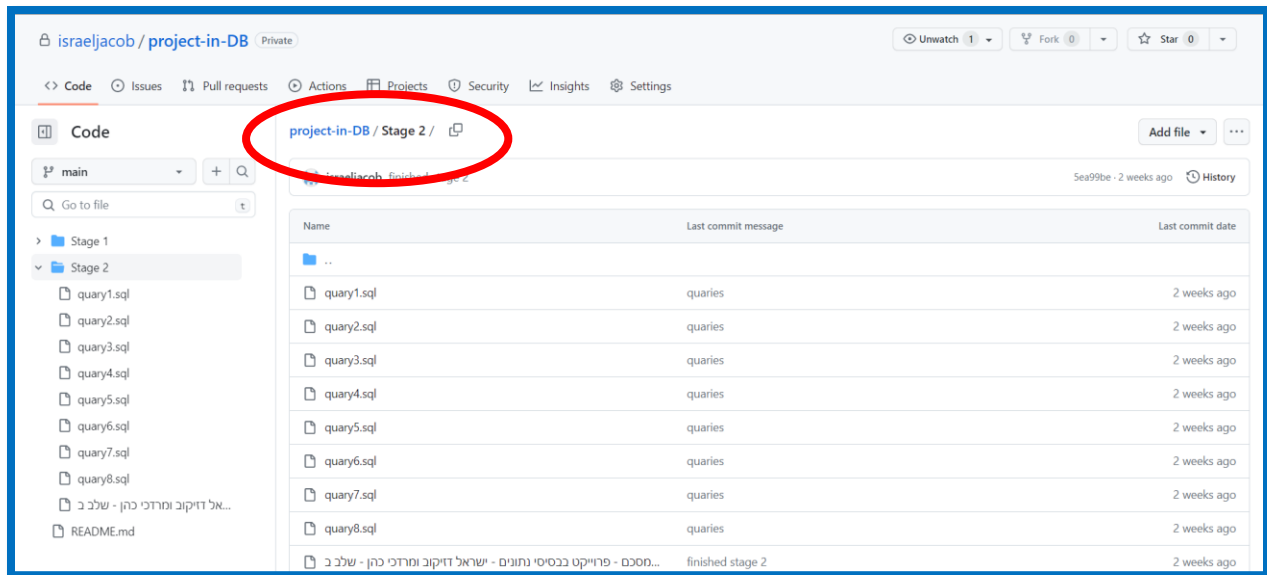
Project.sql

Role.sql

project-in-DB / Stage 1 /

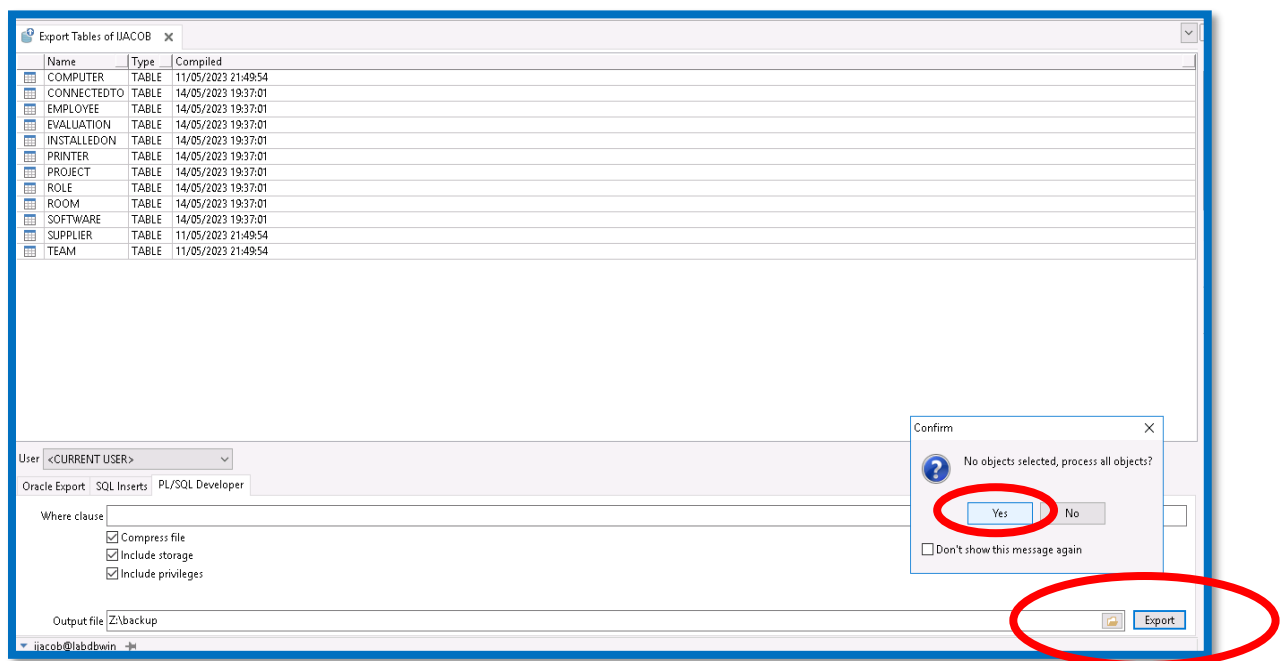
israeljacob fixes 8d311e3 - 2 weeks ago History

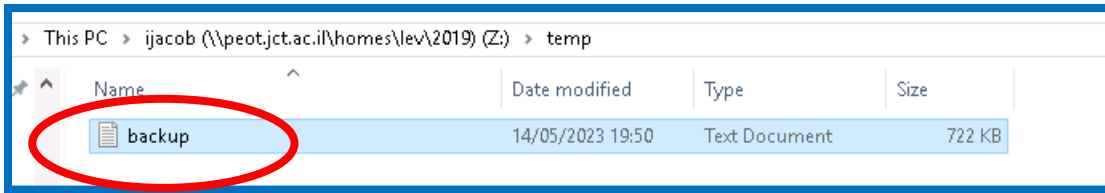
Name	Last commit message	Last commit date
..		
Computer.sql	fixes	2 weeks ago
ConnectedTo.sql	fixes	2 weeks ago
DBCreate.sql	fixes	2 weeks ago
Employee.sql	fixes	2 weeks ago
Evaluation.sql	fixes	2 weeks ago
InstalledOn.sql	fixes	2 weeks ago
MYERD.png	fixes	2 weeks ago
MYLPLSQLDSD.jpg	fixes	2 weeks ago
MyERDDSD.png	fixes	2 weeks ago



גיבוי הנתונים

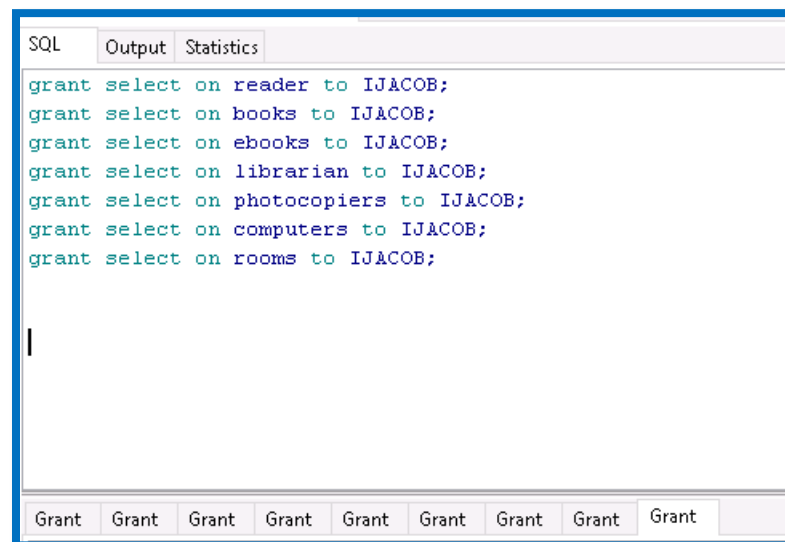
נכנסנו ללשונית **tools** ובחרנו **export tables**. בחרנו את הלשונית **PL/SQL developer** והכנסנו את כתובת היעד בה יישמר הגיבוי. לאחר לחיצה על **export** נשאלנו האם ברצוני לגבות את כל הטבלאות לחצנו על אישור **Yes** ונוצר קובץ הגיבוי ובו נתוני כל הטבלאות.





Grants

:7177

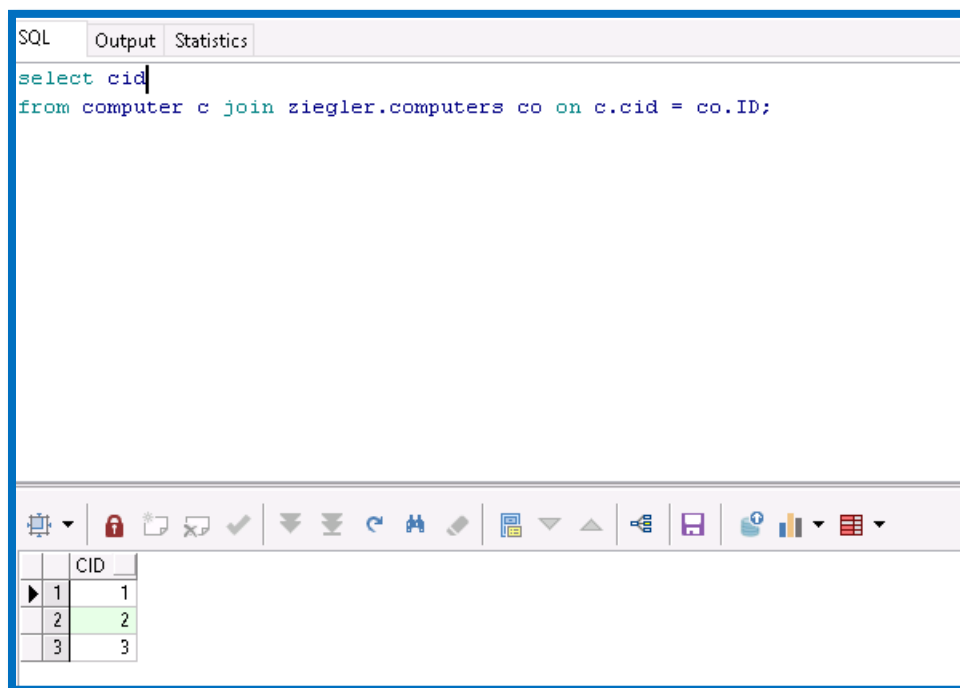


שאלות שמשתמשות במידע החדש מהטבלאות של המשתמש ZIEGLER:

שאלת ראשונה

מעוניינים לקבל את הרשימה של כל המחשבים ששייכים לעובדים וגם משתמשים בהם בספרייה:

הקוד והתוצאה:



The screenshot shows a SQL query editor with three tabs: SQL, Output, and Statistics. The SQL tab is active, displaying the following query:

```
select cid  
from computer c join ziegler.computers co on c.cid = co.ID;
```

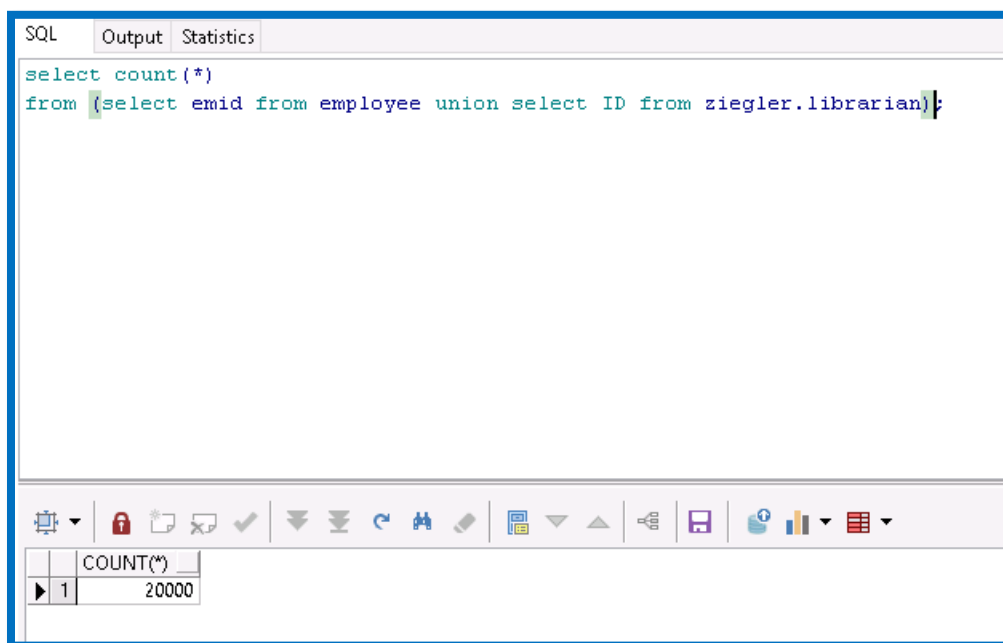
Below the query editor is a toolbar with various icons for database operations. At the bottom, a results table is displayed with the following data:

	CID
1	1
2	2
3	3

שאלתה שניה

רוצים להוציא לגיבוש את עובדי הID ואת הספרנים יחד ורוצים לדעת מה הכמות שלהם

הקוד והתוצאה:



The screenshot shows a SQL IDE window with three tabs: SQL, Output, and Statistics. The SQL tab is active, displaying the following query:

```
select count(*)  
from (select emid from employee union select ID from ziegler.librarian);
```

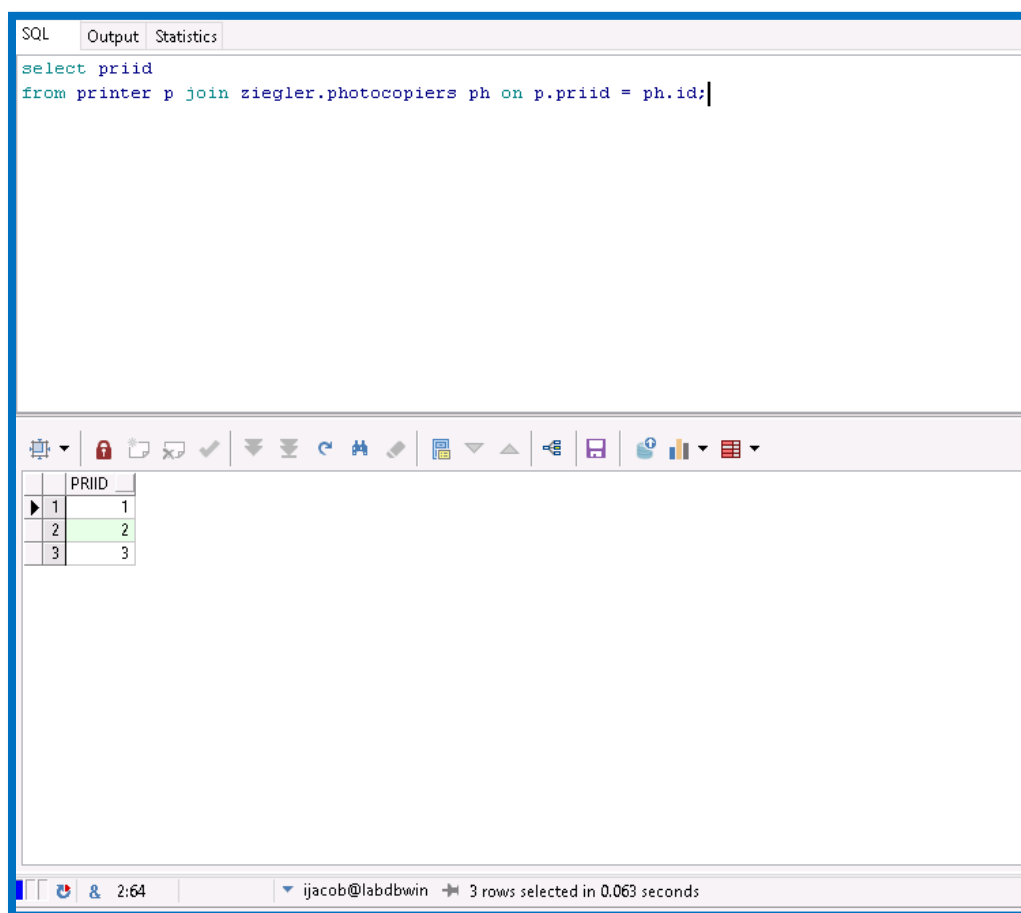
Below the query editor is a toolbar with various icons for database operations. At the bottom, the result set is displayed in a table format:

	COUNT(*)
1	20000

שאלתה שלישי

רוצים למדר בין המדפסות של ה-IT למדפסות של הספרייה ולכן רוצים לדעת האם ישנם מדפסות משותפות.

הקוד והתוצאה:



The screenshot shows a SQL query execution window with three tabs: SQL, Output, and Statistics. The SQL tab is active, displaying the following query:

```
select priid
from printer p join ziegler.photocopiers ph on p.priid = ph.id;
```

Below the query is a toolbar with various icons for execution, saving, and viewing. The Output tab shows the results of the query in a table format:

	PRIID
1	1
2	2
3	3

The status bar at the bottom indicates that 3 rows were selected in 0.063 seconds.

שלב ד'

Views

1. נרצה ליצור טבלה עם פרטים חלקיים של עובדים ומידע על התפקיד והצוות שלהם.

הקוד:

```
create view EMPLOYEE_DETAILS as
(select emid, emname, dateofbirth, rolename, tename
from employee natural join team natural join role);
```

התוצאות:

SQL

Output

Statistics

select t.* from EMPLOYEE_DETAILS t

	EMID	EMNAME	DATEOFBIRTH	ROLENAME	TENAME
1	291134595	Kimberly Danger	12/12/1980	Analyst Koua	Santana do parna?ba
2	291137936	Dylan Hynde	21/05/1991	Developer V?xj?	Zafferana Etnea
3	291137937	Kirsten Diehl	22/06/1992	Administrator Baiima	Sao caetano do sul
4	291137938	Nina Bancroft	23/07/1993	Manager Mentougou	Silverdale
5	291137939	Tobey Hagerty	24/08/1994	Manager Newark	Oosterhout
6	291137940	Miles Larter	25/09/1995	Analyst Ejidal	Boulogne
7	291137941	Angela Patton	26/10/1996	Developer Krasnoye	Leipzig
8	291137942	Steve Lorenz	27/11/1997	Administrator Almalagu?s	Mogliano Veneto
9	291137943	Matthew Heche	28/12/1998	Manager Yuanba	Adamstown
10	291137944	Vince Sisto	01/01/1999	Designer Sada	M?nchen
11	291137945	Hector Cockburn	02/02/2000	Administrator Tafo	Gaithersburg
12	291137946	John Anderson	03/03/2001	Developer Casal	Rothenburg
13	291137947	Ani Applegate	04/04/2002	Manager Al Had?	Blue bell
14	291137948	Buddy Lawrence	05/05/2003	Manager Perpignan	Portsmouth
15	291137949	Marc Fiennes	06/06/2004	Analyst Arneiro	Zafferana Etnea
16	291137950	Will Lynch	07/07/1960	Developer Kabir	Kobe
17	291137951	Leonardo Michael	08/08/1961	Manager Villa Castelli	Kingston
18	291137952	Owen Blades	09/09/1962	Designer Norton	Pompeia
19	291137953	Warren Purefoy	10/10/1963	Developer Wudian	Cherepovets
20	291137954	Neil Pesci	11/11/1964	Manager Kukulu	Boise
21	291137955	Richie Martinez	12/12/1965	Administrator Angra dos Reis	Bellevue
22	291137956	Chubby Moody	13/01/1966	Analyst Paris 05	Arlington
23	291137957	Trace Bryson	14/02/1967	Developer Dz??nbulag	Ciudad del Este
24	291137958	Kurtwood Mattea	15/03/1968	Developer Puerto Deseado	Deerfield
25	291137959	Lee Flanagan	16/04/1969	Designer Lazarevskoye	Streamwood
26	291137960	Jonathan Rhymes	17/05/1970	Administrator Almalagu?s	Pompeia

1:35

0:07

jjacob@labdbwin

20000 rows selected in 7.369 seconds

שאלת 1

תיאור: נרצה את הפרטים של כל העובדים שבתפקיד מפתח.

הקוד:

```
select *
from EMPLOYEE_DETAILS t
where rolename like 'Developer%';
```

התוצאות:

	EMID	EMNAME	DATEOFBIRTH	ROLENAME	TENAME
1	291137936	Dylan Hynde	21/05/1991	Developer V?g?	Zafferana Etnea
2	291137941	Angela Patton	26/10/1996	Developer Krasnoye	Leipzig
3	291137946	John Anderson	03/03/2001	Developer Casal	Rothenburg
4	291137950	Will Lynch	07/07/1960	Developer Kabir	Kobe
5	291137953	Warren Purefoy	10/10/1963	Developer Wudian	Cherepovets
6	291137957	Trace Bryson	14/02/1967	Developer Dz??nbulag	Ciudad del Este
7	291137958	Kurtwood Mattea	15/03/1968	Developer Puerto Deseado	Deerfield
8	291137966	Stephen Ingram	23/11/1976	Developer Okahandja	Lecanto
9	291137973	Tim Niven	02/06/1983	Developer Fushun	Purley
10	291137974	Mika Cruise	03/07/1984	Developer Pizarro	Kopavogur
11	291137978	Tara Carrington	07/11/1988	Developer V?g?	Pompeia
12	291137979	Mary Buckingham	08/12/1989	Developer Layo	Gen?ve
13	291137985	Clea Edmunds	14/06/1995	Developer Taquarituba	Redhill
14	291137994	Mel Marley	23/03/2004	Developer San Francisco	Bretzfelf-Waldbach
15	291137999	Louise Shorter	28/08/1964	Developer Kabir	Vancouver
16	291138004	Miguel Benet	05/01/1969	Developer Hod HaSharon	Berlin
17	291138005	Phoebe Brooks	06/02/1970	Developer Peh?evo	Morioka
18	291138012	Holly Trejo	13/09/1977	Developer Piotr?w Kujawski	Boucherville
19	291138018	Sophie O'Donnell	19/03/1983	Developer Shreveport	Harsum
20	291138026	Mary Beth Nash	27/11/1991	Developer Culianin	Dalmine
21	291138029	Kelly Merchant	02/02/1994	Developer Zhouzai	Ljubljana
22	291138031	Bill Levert	04/04/1996	Developer Argelia	Yokohama
23	291138042	Molly Dunn	15/03/1962	Developer Ibaan	Bloomington
24	291138044	Coley Brandt	17/05/1964	Developer ?Al?qahd?r? G?l?n	Rotterdam
25	291138045	Isabella Gore	18/06/1965	Developer San Pedro Sacatep?quez	Carlingford
26	291138054	Stanley Holbrook	27/03/1974	Developer Qarg?n	Lund

שאלתה 2

תיאור: רוצים למצוא את כל העובדים שמעל גיל 60 שעובדים בצוות בשם 'Lyon'.

הקוד:

```
SQL Output Statistics
select *
from EMPLOYEE_DETAILS e
where (sysdate - e.dateofbirth)/365 >= 60 and e.tenname like 'Lyon';
```

התוצאות:

	EMID	EMNAME	DATEOFBIRTH	ROLENAME	TENAME
1	291137095	Patrick Brandt	20/04/1960	Developer Ipubi	Lyon
2	111115882	Miko Vinton	07/11/1961	Manager Las Heras	Lyon
3	291138176	Mac Sizemore	09/05/1961	Administrator Ar Ru?ays	Lyon
4	291136376	Ice Spacey	01/05/1961	Designer San Luis	Lyon
5	111112235	Beth Margolyes	05/12/1952 23:24:43	Administrator Qiandian	Lyon
6	111135372	Bobbi McDowall	29/05/1925 03:51:15	Designer Al Ma?all??	Lyon

2. נרצה ליצור טבלה שמורכבת מפרטים על פרויקט ועל הקומה שבה יושב הצוות שעובד עליו.

הקוד:

```
SQL Output Statistics
create view PROJECT_FLOOR as
select proid, proname,duedate, teid, floor
from Project natural join team natural join room;
```

התוצאות:

	PROID	PRONAME	DUEDATE	TEID	FLOOR
1	1	Perez	01/01/2016	943	9
2	2	McGill	02/02/2017	765	5
3	3	Goldberg	03/03/2018	892	4
4	4	O'Neill	04/04/2019	233	7
5	5	Fox	05/05/2020	796	7
6	6	Kinnear	06/06/2021	510	9
7	7	Cash	07/07/2022	696	5
8	8	McNarland	08/08/2023	940	9
9	9	de Lancie	09/09/2024	837	1
10	10	Oszajca	10/10/2025	924	2
11	11	Hershey	11/11/2026	103	3
12	12	Sossamon	12/12/2027	818	3
13	13	Gagnon	13/01/2028	469	1
14	14	Cusack	14/02/2029	272	5
15	15	Guest	15/03/2030	576	6
16	16	Perez	16/04/2031	685	3
17	17	Bedelia	17/05/2016	550	7
18	18	Gunton	18/06/2017	600	4
19	19	Merchant	19/07/2018	745	7
20	20	Ledger	20/08/2019	601	1
21	21	Curtis	21/09/2020	546	2
22	22	Epps	22/10/2021	825	2
23	23	Bridges	23/11/2022	503	6
24	24	Maxwell	24/12/2023	971	5
25	25	Kramer	25/01/2024	226	1
26	26	Cronin	26/02/2025	847	1
27	27	Fox	27/03/2026	469	8

1:8 | jacob@labdbwin | 2000 rows selected in 0.768 seconds

שאלתה 1

תיאור: בקומה 3 עומדים לעשות שיפוצים ולכן יהיו שיבושים בעבודה של הצוותים שיושבים בקומה זו. לצורך כך, חצים רשימה של כל הפרויקטים שהצוותים שלהם יושבים בקומה 3.

הקוד:

```
SQL      Output  Statistics
select proid, pronomame, duedate
from PROJECT_FLOOR
where floor = 3;
```

התוצאות:

	PROID	PRONAME	DUE DATE
1	11	Hershey	11/11/2026
2	12	Sossamon	12/12/2027
3	16	Perez	16/04/2031
4	32	Pacino	04/08/2031
5	33	Laws	05/09/2016
6	51	McCabe	23/03/2018
7	66	Travolta	10/06/2017
8	68	Herrmann	12/08/2019
9	80	Scaggs	24/08/2031
10	81	Reinhold	25/09/2016
11	84	Danger	28/12/2019
12	85	Cornell	01/01/2020

11:1 | ijacob@labdbwin | 226 rows selected in 0.423 seconds

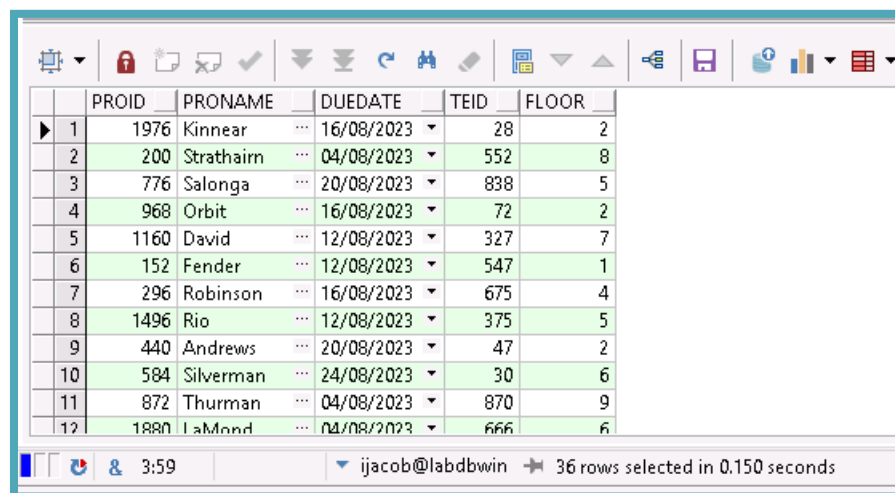
שאלתה 2

תיאור: בכל קומה הצוותים מסייעים אחד לשני. רוצים לפזר בין הקומות את הפרויקטים שתאריכי היעד שלהם קרובים כדי שיהיה לצוות שלהם כמה שיותר עזרה. לשם כך צריכים את כל הפרויקטים שתאריכי היעד שלהם הינם בעוד פחות מ-3 חודשים.

הקוד:

```
SQL Output Statistics
select *
from PROJECT_FLOOR
where (duedate - sysdate) < 90 and (duedate - sysdate) > 0;
```

התוצאות:



	PROID	PRONAME	DUE DATE	TEID	FLOOR
1	1976	Kinnear	16/08/2023	28	2
2	200	Strathairn	04/08/2023	552	8
3	776	Salonga	20/08/2023	838	5
4	968	Orbit	16/08/2023	72	2
5	1160	David	12/08/2023	327	7
6	152	Fender	12/08/2023	547	1
7	296	Robinson	16/08/2023	675	4
8	1496	Rio	12/08/2023	375	5
9	440	Andrews	20/08/2023	47	2
10	584	Silverman	24/08/2023	30	6
11	872	Thurman	04/08/2023	870	9
12	1880	LaMond	04/08/2023	666	6

פונקציות

1. הערכה אחרונה של עובד.

תיאור: בהינתן מספר עובד נרצה לדעת מהו הציון האחרון שקיבל העובד.

הקוד:



```
SQL Output Statistics
CREATE OR REPLACE FUNCTION GetLastGrade(EmID_param IN INT)
RETURN INT IS
    last_grade INT;
    CURSOR grade_cursor IS
        SELECT Grade
        FROM Evaluation
        WHERE EmID = EmID_param
        ORDER BY Year1 DESC
        FETCH FIRST 1 ROW ONLY;
BEGIN
    OPEN grade_cursor;
    FETCH grade_cursor INTO last_grade;
    CLOSE grade_cursor;
    RETURN last_grade;
EXCEPTION
    WHEN NO_DATA_FOUND THEN
        RETURN NULL;
END;
```

וגמא:


נראה שאכן הציון 5 הוא הציון האחרון

SQL

Output

Statistics

```
select *  
from evaluation  
where emid = 111112729  
order by year1;
```

Test script	DBMS Output	Statistics	Profiler	Trace
				
1	begin			
2	-- Call the function			
3	:result := getlastgrade(emid_param => :emid_param);			
4	end;			

<input type="checkbox"/>	Variable	Type	Value
<input checked="" type="checkbox"/>	result	Float	5
<input checked="" type="checkbox"/>	emid_param	Float	111112729
<input type="checkbox"/>	307134723	String	

2. פרויקט קרוב של צוות.

תיאור: בהינתן צוות נרצה לדעת מה הפרויקט שלו עם התאריך יעד הקרוב ביותר.

הקוד:

```
SQL Output Statistics
CREATE OR REPLACE FUNCTION get_earliest_due_date_project(
    team_id IN INT
)
RETURN PROJECT%ROWTYPE
IS
    v_earliest_project PROJECT%ROWTYPE;
    v_current_date DATE := SYSDATE;

    CURSOR c_projects IS
        SELECT p.*
        FROM Project p
        INNER JOIN Team t ON p.TeID = t.TeID
        WHERE t.TeID = team_id
        ORDER BY p.DueDate ASC;
BEGIN
    -- Open cursor and fetch projects
    OPEN c_projects;

    -- Fetch the first project from the cursor
    FETCH c_projects INTO v_earliest_project;

    -- Close cursor
    CLOSE c_projects;

    -- Return the earliest project
    RETURN v_earliest_project;
END;
```

דוגמא:

נראה שאכן הפרויקט עם תאריך היעד הקרוב של צוות 836 הינו 792

SQL Output Statistics

```
select *
from project
where teid = 836
order by duedate;
```

	PROID	PRONAME	DUEDATE	COST	PROFIT	TEID
1	482	Baranski	06/02/2017	11356	4490	836
2	38	Dafoe	10/02/2021	10659	27028	836
3	1479	Loring	23/03/2022	13203	20942	836
4	599	Pride	11/11/2022	7452	3245	836
5	1288	Copeland	20/01/2023	12544	23583	836
6	792	Applegate	08/12/2023	2684	14382	836
7	281	Pleasence	01/03/2024	14096	17596	836
8	1358	Quinlan	14/02/2029	10943	23666	836

Test script DBMS Output Statistics Profiler Trace

```
1 begin
2   -- Call the function
3   :result := get_earliest_due_date_project(team_id => :team_id);
4 end;
```

Variable	Type	Value
result	Float	792
team_id	Float	836

פרוצדורות

1. חיבור מחשב למדפסת

תיאור: בהינתן מספר מזהה של מחשב ומספר מזהה של מדפסת נרצה להוסיף שורה בטבלה connectedTo שתחבר בין המחשב למדפסת.

הקוד:

```
SQL      Output  Statistics
CREATE OR REPLACE PROCEDURE AddComputerToPrinter
(
  p_ComputerID IN INT,
  p_PrinterIDs IN INT
)
AS
  -- Check if the computer exists
  l_ComputerCount INT;
BEGIN
  SELECT COUNT(*)
  INTO l_ComputerCount
  FROM Computer
  WHERE CID = p_ComputerID;

  IF l_ComputerCount = 0 THEN
    RAISE_APPLICATION_ERROR(-20001, 'Computer does not exist.');
```

```
    RETURN;
  END IF;

  -- Check if the printer exists
  l_ComputerCount := 0;
  SELECT COUNT(*)
  INTO l_ComputerCount
  FROM Printer
  WHERE PriID = p_PrinterIDs;

  IF l_ComputerCount = 0 THEN
    RAISE_APPLICATION_ERROR(-20002, 'Printer with ID ' || p_PrinterIDs || ' does not exist.');
```

```
    RETURN;
  END IF;

  -- Check if the computer is already connected to the printer
  l_ComputerCount := 0;
  SELECT COUNT(*)
  INTO l_ComputerCount
  FROM ConnectedTo
  WHERE CID = p_ComputerID AND PriID = p_PrinterIDs;

  IF l_ComputerCount > 0 THEN
    RAISE_APPLICATION_ERROR(-20003, 'Computer is already connected to Printer with ID ' || p_PrinterIDs || '.');
```

```
    RETURN;
  END IF;

  -- Connect the computer to the printer
  INSERT INTO ConnectedTo (PriID, CID)
  VALUES (p_PrinterIDs, p_ComputerID);
END;
```

```

1 begin
2   -- Call the procedure
3   addcomputertoprinter(p_computerid => :p_computerid,
4                       p_printerids => :p_printerids);
5 end;

```

Variable	Type	Value
p_computerid	Float	30
p_printerids	Float	50

1:1 | ijacob@labdbwin | Executed in 0.01 seconds

נראה שאכן נוספה השורה

SQL | Output | Statistics

```

select *
from connectedto
where cid = 30 and priid = 50;

```

	PRIID	CID
1	50	30

2. מחיקת פרויקטים ישנים.

תיאור: נרצה למחוק את כל הפרויקטים שעבר תאריך היעד שלהם.

הקוד:

```
SQL Output Statistics
CREATE OR REPLACE PROCEDURE DeleteExpiredProjects
AS
BEGIN
    DELETE FROM Project
    WHERE (DueDate - sysdate) < 0;

    COMMIT;
EXCEPTION
    WHEN OTHERS THEN
        DBMS_OUTPUT.PUT_LINE('Error occurred: ' || SQLERRM);
        ROLLBACK;
END;
```

דוגמה:

```
Test script DBMS Output Statistics Profiler Trace
1 begin
2 -- Call the procedure
3 deleteexpiredprojects;
4 end;
```

נראה שאכן נמחקו כל הפרויקטים הישנים

```
SQL Output Statistics
select *
from project
where (sysdate - dueDate) > 0;
```

PROID	PRONAME	DUEDATE	COST	PROFIT	TEID
-------	---------	---------	------	--------	------

טריגרים

1. מחיקת מדפסת.

תיאור: כאשר אנחנו מוחקים מדפסת מהטבלה printer, נרצה למחוק את כל השורות בטבלה connectedTo שמופיעה בהם המדפסת שנמחקה.

דקוד:

```
SQL Output Statistics
CREATE OR REPLACE TRIGGER delete_connectedto_trigger
AFTER DELETE ON Printer
FOR EACH ROW
BEGIN
DELETE FROM ConnectedTo
WHERE PriID = :old.PriID;
END;
```

דוגמה:

נראה שכל השורות הנ"ל בטבלה
connectedTo, נמחקו

```
SQL Output Statistics
select *
from Connectedto
where priid = 45
```

	PRIID	CID

נמחק את מדפסת 45

```
SQL Output Statistics
delete from printer
where priid = 45
```

בטבלה connectedTo קיימים 18
שורות עם מדפסת מספר 45

	PRIID	CID
1	45	25
2	45	85
3	45	92
4	45	98
5	45	176
6	45	204
7	45	316
8	45	360
9	45	497
10	45	501
11	45	511
12	45	512
13	45	644
14	45	682
15	45	688
16	45	707
17	45	761
18	45	903

2. הוספת מחשב

תיאור: כאשר מוסיפים מחשב, נרצה שתהיה מותקנת עליו תוכנה מספר 1 שהיא תוכנה בסיסית.

הקוד:

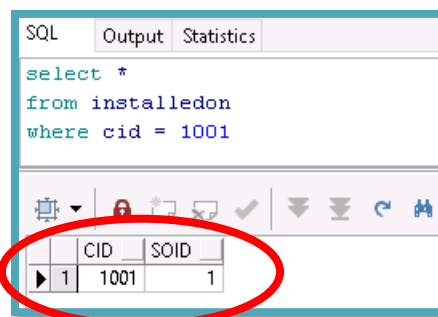
```
CREATE OR REPLACE TRIGGER insert_installedOn_trigger
AFTER INSERT ON Computer
FOR EACH ROW
BEGIN
    INSERT INTO installedon (CID, soid)
    VALUES (:NEW.CID, 1);
END;
```


דוגמה:

נוסיף מחשב

```
insert into computer values(1001, 'i7', 16, 512, 2023)
```

נבדוק שהשורה המבוקשת
קיימת בinstalledOn



SQL			Output	Statistics
<pre>select * from installedon where cid = 1001</pre>				
				
		CID	SOID	
▶ 1	1001	1		

שלב ה'

דוחות של שאילתות עם פרמטרים

שאילה ראשונה עם פרמטרים

תיאור:

כל פרטי העובדים שיש להם סוג מסוים של תפקיד. יוצגו הפרטים הרלוונטיים בלבד, שם העובד, תפקידו, שם הצוות שלו ומספר המחשב האישי שלו.

שימושי עבור אדם המעוניין רק בעובד עם סוג תפקיד ספציפי, יקל את החיפוש.

הוספנו משתנה שבזמן הריצה יקבל את ערכו והוא הקטגוריה הרצויה:

דקוד:

```
select emname "employee name",rolename "role name", tename "team name", cid "computer id"
from employee, role, team
where employee.roleid = role.roleid
and team.teid = employee.teid
and role.rolename like (&<name= "role category"
list="Administrator, Analyst, Designer, Developer, Manager"
type="string"> || '%')
order by emname;
```

וגמא:

הדוח שהתקבל:

בחירת משתנים:

role category = Designer			
employee name	role name	team name	computer id
Aaron Padillo	Designer Postes	Rafaelica	169
Adrian Caine	Designer Gelin	McDonalder	232
Aida McKellen	Designer K/Team	Siddhach	90
Annee Standiford	Designer Santana	Cleveland	779
Al Austin	Designer Latta	Lipshman	542
Al Peniston	Designer Orogbo	Bloomington	23
Al Schreiber	Designer Pengadounan	Cape town	130
Alan Wilcox	Designer Oslo	Waco	603
Aleahh Humes	Designer Celica	Regensburg	779
Aleahh Lange	Designer Gamba	Moorestown	295
Alec Seaguy	Designer Santana	Amherst	954
Alexandro Cocker	Designer Dr/Tridling	Draper	384
Alex Andrews	Designer Puerto Tojeda	Brashear	147
Alex Epps	Designer Taken	Brashear	557
Alex Galtst	Designer Orogbo	San Mateo	

Name	Value
role category	Administrator Analyst Designer Developer Manager

שאלה שניה עם פרמטרים

תיאור:

שאלתה המחזירה את פרטי הפרויקטים שהרווח מהם נע בין שני סכומים, סכום מינימלי וסכום מקסימלי. יופיע שם הפרויקט, תאריך היעד שלו, העלות והרווח.
ניתן למפעיל השאלתה לבחור מה יהיו הסכומים.

דקוד:

```
select pronaame "project name", due date "due date", cost "cost", profit "profit"
from project
where profit between <name="minimum profit" type=integer>
and <name="maximum profit" type=integer>
order by due date;
```

דוגמא:

הדוח שהתקבל:

בחירת משתנים:

minimum profit = 10000 maximum profit = 10500				
project name	due date	cost	profit	
Orbit	16/08/2023	17777	10324	
Tambor	09/05/2024	14470	10329	
MacIsaac	21/09/2024	7533	10166	
Quatro	25/09/2024	1351	10103	
Apple	10/06/2025	18868	10124	
Mahood	18/10/2025	5354	10050	
Harfield	27/11/2026	8275	10365	
Collie	04/04/2027	15303	10045	
Derydas	24/12/2027	2569	10011	
Guanini	05/01/2028	16565	10072	
Brody	13/09/2028	2260	10182	
Colen	07/03/2030	2594	10409	
Hammon	07/07/2030	17502	10200	
Ozajca	11/07/2030	2089	10428	
Belin	07/11/2030	15002	10161	

גרפים

שאלתה ראשונה + גרף

תיאור:

נריץ שאלתה המחזירה את כמות הפרויקטים שבוצעו על ידי כל צוות, ולאחר מכן נבנה גרף מתאים. עוזר לארגון נכון של הפרויקטים והחלוקה בין הצוותים.

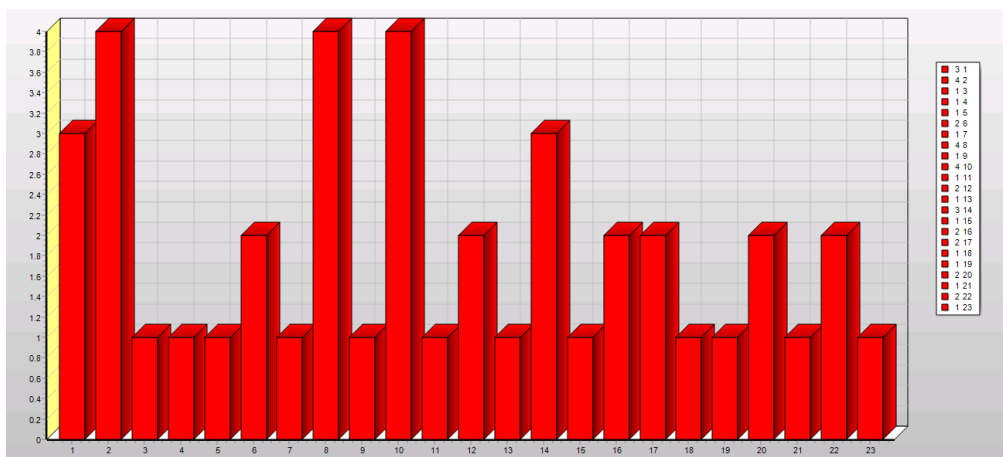
הקוד והתוצאות:

```
select count(*)
from project
group by teid
order by teid;
```

	COUNT(*)
1	3
2	4
3	1
4	1
5	1
6	2
7	1
8	4
9	1
10	4
11	1

הגרף המתקבל לאחר עיצוב:

כמות הפרויקטים שמתבצעים על ידי כל צוות



שאלתה שניה + גרף

תיאור:

נרץ שאלתה המחזירה את כמות הצוותים בכל קומה ולאחר מכן בנה גרף מתאים. עוזר לארגון נכון של איזון העומס בחדרים.

הקוד והתוצאות:

```
select count(*)
from room r, team t
where r.roomnumber = t.roomnumber
group by floor;
```

	COUNT(*)
1	102
2	104
3	218
4	95
5	162
6	78
7	99
8	35
9	107

הגרף המתקבל לאחר עיצוב:

מספר הצוותים בכל קומה

