

מסדי נתונים – 89-281

תרגול 5 - DBS, מודלים, אלג' יחסים 1

עמיעד רוזנברג

The human mind ordinarily operates at only ten 10% of its capacity, the rest is overhead for the operating system.

הגדרות

- שדה – Field

אוסף תווים בעל קשר לוגי. מדובר על יחידת הנתונים הקטנה ביותר (שם, ת"ז, וכו').
קיימת אפשרות להגביל את אורכו של שדה (למשל: אורך שדה המכיל ת"ז יוגבל ל-9 ספרות).

- רשומה – Record

אוסף של שדות בעלי קשר לוגי (למשל, אוסף של שדות המייצגים פרטי סטודנט).
לדוגמא:

שם פרטי	שם משפחה	ת"ז	טלפון	משכורת
10 תווים	10 תווים	9 ספרות	10 ספרות	4 ספרות (☹)

הגדרות

• קובץ - File

אוסף של רשומות.

נזכיר שאמרנו מקודם שניתן להגדיר אורך לכל שדה. אבל, הנתונים שלנו הם לא בגודל אחיד (השם "דן" הוא באורך 2, השם "אלי" הוא באורך 3 והשם "מתושלח" הוא באורך 6. מכיוון שהגדרנו את השדה "שם פרטי" באורך 10, עלינו להוסיף רווחים ע"מ להתאים את עצמנו למבנה הקובץ).

עבודה עם מערכת קבצים

- מערכת הקבצים היא בעצם השיטה הישנה (מאוד) לאחסון נתונים. הנתונים היו נשמרים בקבצים שונים בצורה סדרתית.
- לדוגמא :
נזכיר את הרשומה שראינו מקודם :

שם פרטי	שם משפחה	ת"ז	טלפון	משכורת
10 תווים	10 תווים	9 ספרות	10 ספרות	4 ספרות (☹)

- רשומה במבנה הזה תיראה כך :
`Amiad#####Rozenberg#12345678903987654321000`

בעיות בעבודה עם מערכת קבצים

- נניח שנרצה לשמור את המידע שלנו במערכת קבצים. למשל – נייצר קבצים שונים בהתאם לסוגי המידע השונים ונשמור בקבצים אלו את המידע. אילו בעיות יכולות להיות לנו?
- כל תוכנית שתראה לעבוד עם הקובץ תידרש להכיר את המבנה המדויק של הקובץ.
- בעייתיות בתמיכה בשינויים במבנה הקובץ (כל שינוי במבנה יצטרך להיות מעודכן בכל התוכנות המשתמשות עם הקבצים).
- בעייתיות בהגדרת מבנה הקובץ. (למשל, מה יסמן סוף שדה? סוף רשומה?)
- ע"מ לאתר רשומה (או לגלות שלא קיימת אחת כזו) יש לעיתים לסרוק את כל הקובץ (איטיות).
- אבטחת מידע (מה יקרה אם לא נרצה להציג את כל השדות או את כל הרשומות לחלק מהאנשים?).
- כפילויות (מידע המופיע פעמיים ב-2 קבצים שונים).
- אי אמינות הנתונים (עדכון השדה "שם פרטי" שבוצע רק בחלק מהקבצים ולא בכולם).
- בעייתיות באיחוד מידע הקיים במספר קבצים, בשיתוף מידע ביניהם והסקת מסקנות מהמידע המשותף החדש (למשל, קובץ שמות עם קובץ הזמנות ע"מ לגלות את שמות האנשים שביצעו הזמנה מסוימת).
- בעייתיות בסנכרון בין מספר משתמשים שונים הנוגעים במידע. (שני אנשים רוצים לעדכן את הקובץ)
- בעייתיות במיון רשומות לפי שדות שונים. (כל מיון מצריך קובץ משל עצמו)

הפיתרון:

- עבודה עם מערכת בסיס נתונים!
- הרעיון : המבנה הפיסי של הנתונים וצורת ההתנהלות את מול המבנה הפיסי הופכים לשקופים עבור למשתמש.

עוד (קצת) הגדרות

- בסיס \ מסד נתונים – DataBase
אוסף של נתונים מסוגים שונים המקושרים ביניהם ע"י קשר לוגי כלשהו.
- מערכת ניהול בסיס נתונים – DataBase Management System
מערכת (תוכנה \ אפליקציה) לניהול בסיסי נתונים המאפשרת :
יצירה, תחזוקה ושימוש בבסיס הנתונים תוך התמודדות עם ריבוי משתמשים במקביל.
- מערכת מסד נתונים – DataBase System
שילוב של DB + DBMS.
אמור לפתור לנו את הבעיות שדיברנו עליהם מקודם.

דרישות מ-DBMS

- תמיכה במודל נתונים כלשהו (נלמד מייד מהו מודל נתונים).
- יכולות אחסון ושליפת נתונים בצורה נוחה (תמיכה בשפה עילית כלשהי המאפשרת עבודה נוחה למשתמש, בעוד שביצוע הפעולות מול מערכת הקבצים בפועל נעשה בצורה השקופה למשתמש).
- אבטחת מידע (רק בעל הרשאה מתאימה יכול לגשת אל הנתונים).
- תמיכה בשימוש ע"י מספר משתמשים שונים בו זמנית.
- ביצוע פעולות אוטומטיות (גיבוי, שחזור, העתקה, כיווץ וכו').

מודלים של בסיסי נתונים

- מודל נתונים הוא בעצם הצורה להגדיר את הקשרים בין הנתונים השונים.
- ישנם 3 מודלי נתונים :
 - המודל ההיררכי
 - המודל הרישתי
 - המודל היחסי / טבלאי / רלציוני

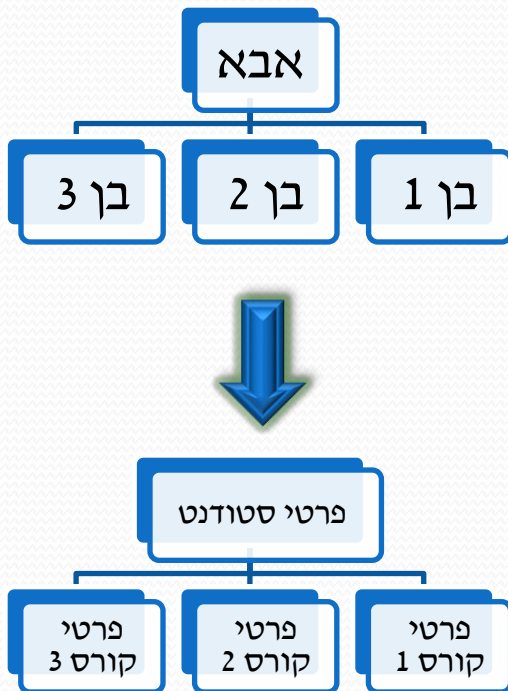
המודל ההיררכי

הגדרה:

- מודל שבו הנתונים מסודרים במבנה של עץ.
- קדקודי העץ מציינים את הרשומות והקשתות מציינות את הקשרים בין הרשימות.

כללים:

- לכל בן (רשומה) יש אב אחד בלבד.
- לא קיימים קשרים בין אחים.
- יש קשר של יחיד לרבים בין האבא לבן. (אפשר יותר מבן אחד).



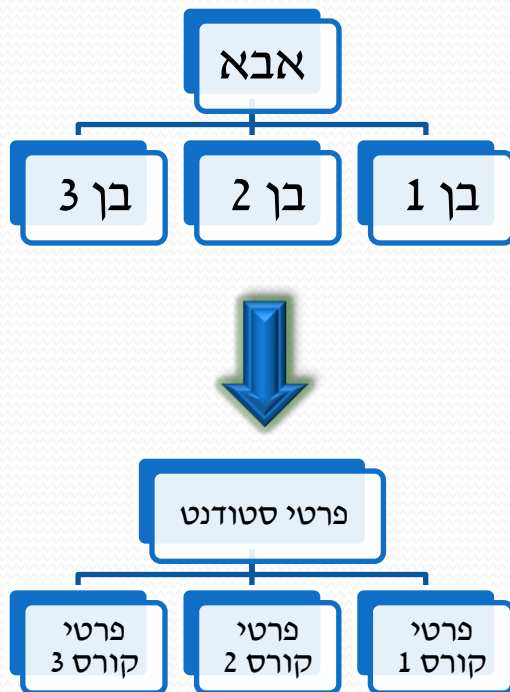
המודל ההיררכי

שאלתא:

- השאלתא היא בעצם הליכה על מסלול עד לרשומה מסוימת בעץ.

בעיות:

- עלולה להיווצר לנו כפילות נתונים :
מתרגל סטודנט
עמיעד ← 89281 משה ← 89281
הרשומה "89281" (על כל פרטיה) מופיעה פעמיים.
עדכון פרטי הקורס יצטרך להתבצע ב-2 מקומות.
- לא ניתן להוסיף רשומת בן עד שהיא לא מוכלת ברשומת אב (למשל, קורס חדש שנפתח שעדיין אף סטודנט לא נרשם אליו).
- המודל אינו יודע לתאר קשרים מסוג של רבים לרבים.



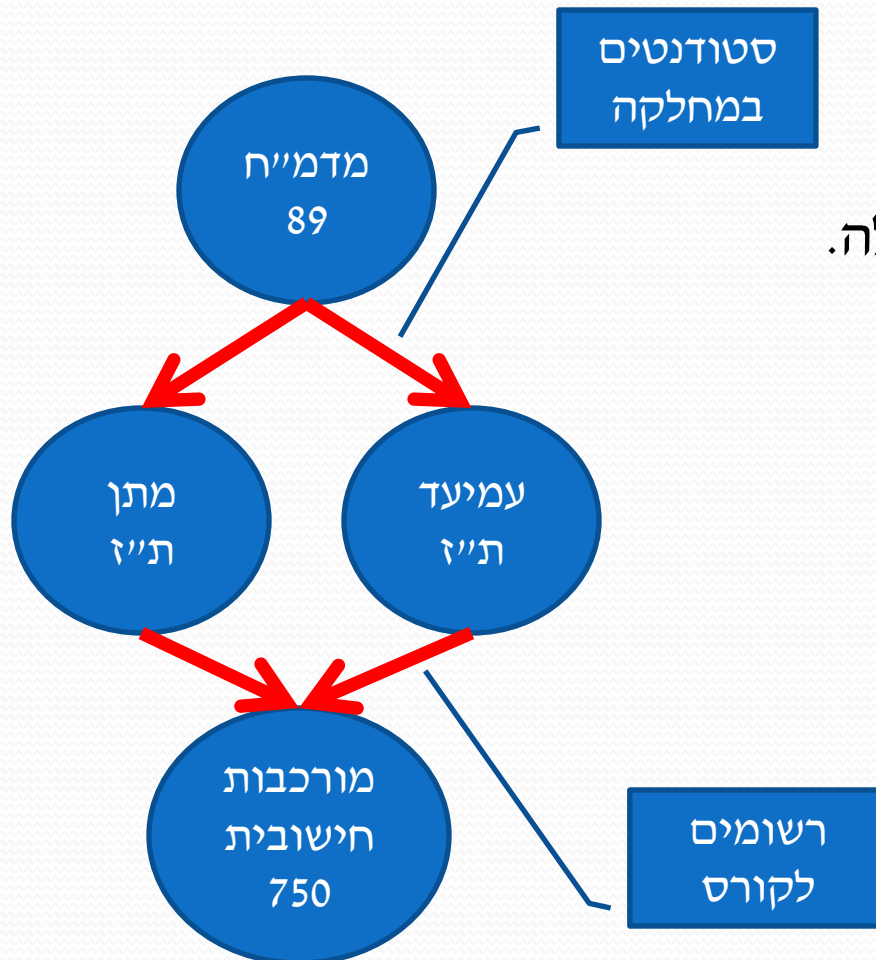
המודל הרישתי

הגדרה:

- רשת של קדקודים וקשתות ללא הגבלה.
- הנתונים מוגדרים בקודקודים.
- הקשרים מוגדרים בקשתות.

כללים:

- יכול להיות קשר בין כולם לכולם. בעצם, מחליפים את העץ מהמודל הרישתי בגרף.



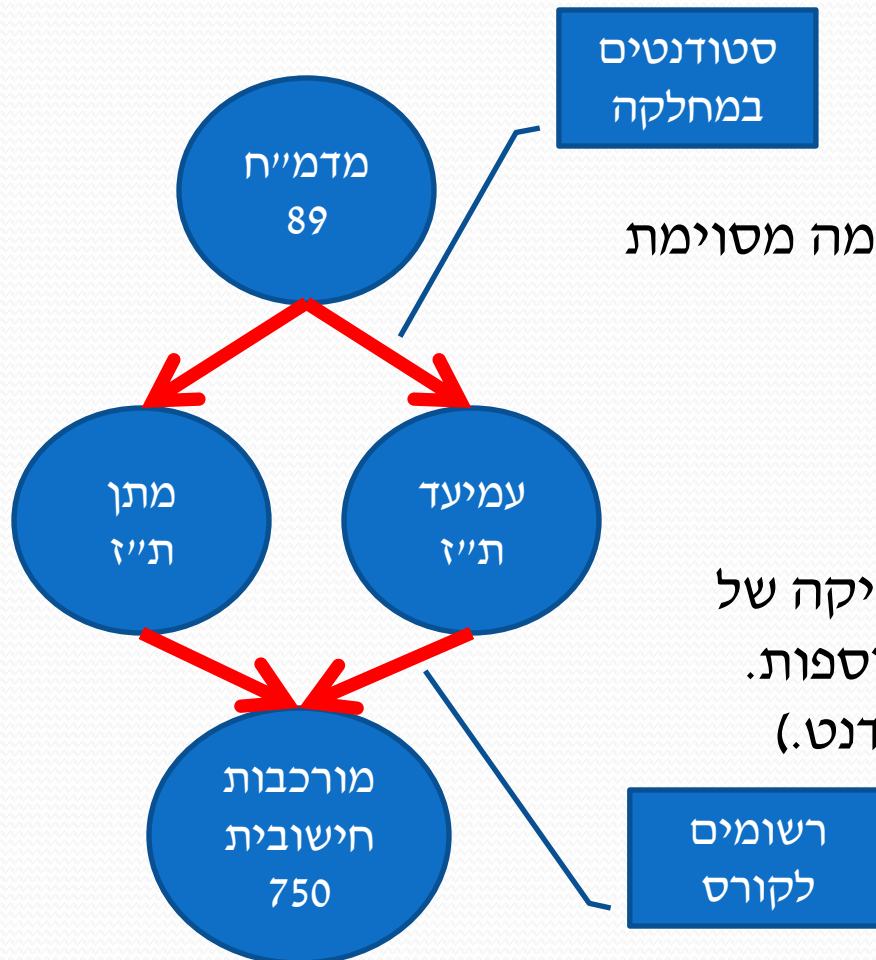
המודל הרישתי

שאלתא:

- השאלתא היא בעצם מסלול עד לרשומה מסוימת בגרף.

בעיות:

- מורכבות גבוהה מאוד!
- קשה מאוד לבצע שינויים במידע. (מחיקה של רשומה עלולה לגרום לפעולות רבות נוספות. חישוב על מצב שבו נרצה למחוק סטודנט.)



המודל היחסי \ טבלאי

הגדרה:

- בסיס נתונים הבנוי מטבלאות.

כללים:

- מאפשר לתאר קשרים רב-רב-ערכיים.
- סדר השורות אינו חשוב.
- מס' השורות אינו מוגבל.
- מס' העמודות – סופי.
- סדר העמודות לא חשוב.
- שימו לב!
טבלאות עם סדר עמודות שונה שקולות,
אך לא שוות!
- טבלה = יחס (relation).
- עמודה = תכונה (attribute).

מס' מחלקה	שם מחלקה
89	מדמ"ח
88	מתמטיקה

ID	מחלקה	משפחה	פרטי
1	89	קידר	מתן
2	89	רוזנברג	עמיעד

המודל היחסי \ טבלאי - הגדרות

מס' מחלקה	שם מחלקה
89	מדמ"ח
88	מתמטיקה

ID	מחלקה	משפחה	פרטי
1	89	קידר	מתן
2	89	רוזנברג	עמיעד

- סכמה – מבנה בסיס הנתונים. סדרת תכונות במדל הטבלאי.
- תחום (Domain) – אוסף הערכים שתכונה יכולה לקבל. (דוגמא: אוסף התווים ב-א"ב באורך 10, עבור תכונת "שם פרטי")
- מופע של יחס R – תת קבוצה במכפלה הקרטזית $D_1 \times D_2 \times \dots \times D_n$
- סימון: R(S) – יחס R עם סכמה S.
- Tuple (n-יה) – ווקטור של תכונות המצביע על שורה מסויימת.
 $t_2 = \{2, 89, \text{Rozenberg}, \text{Amiad}\}$
בצורה הבאה יש משמעות לסדר העמודות:
 $t_2[f_name] = \text{'Amiad'}$
 $t_2[l_name] = \text{'Rozenberg'}$

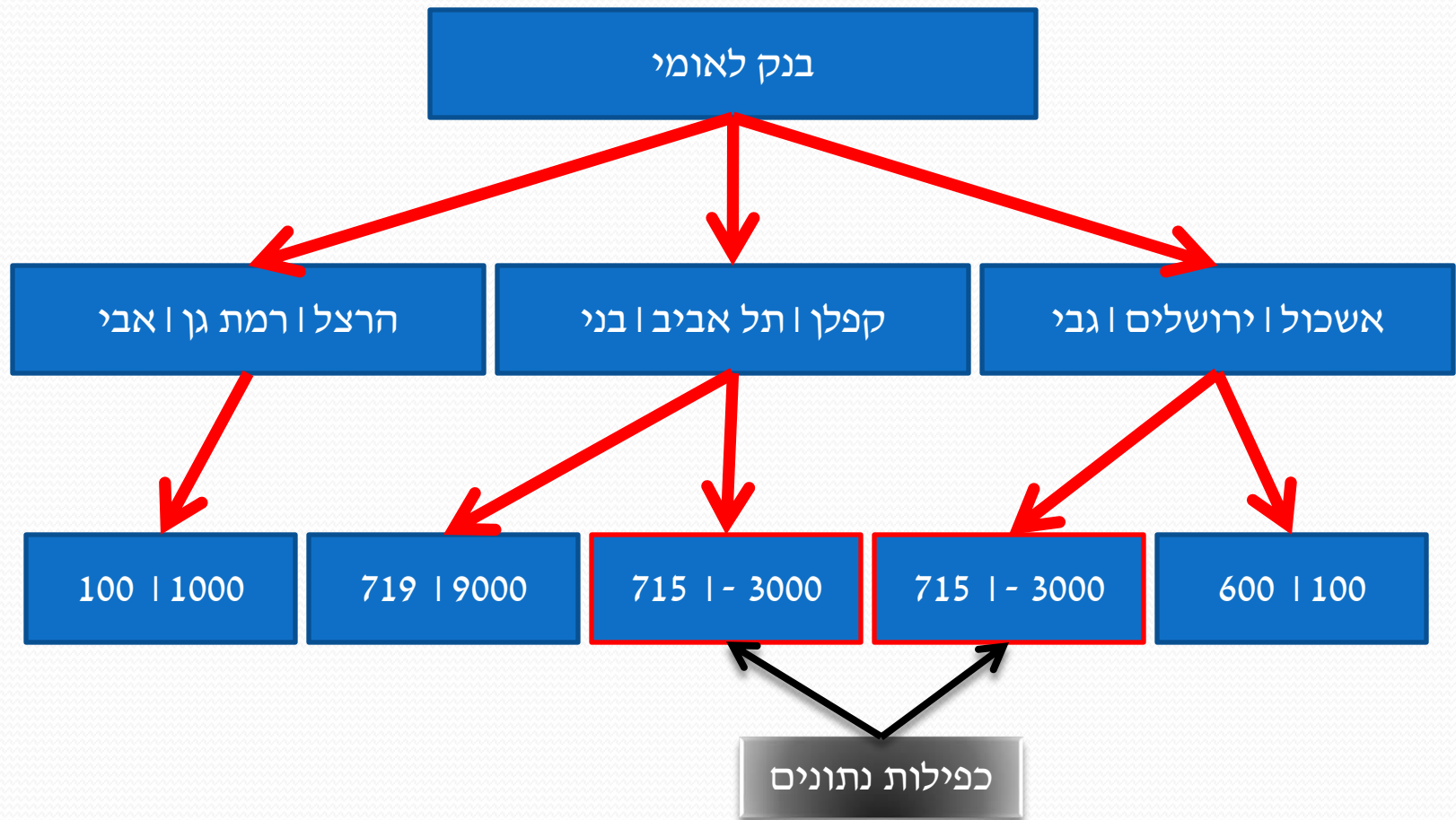
המודל היחסי \ טבלאי - הגדרות

מס' מחלקה	שם מחלקה
89	מדמ"ח
88	מתמטיקה

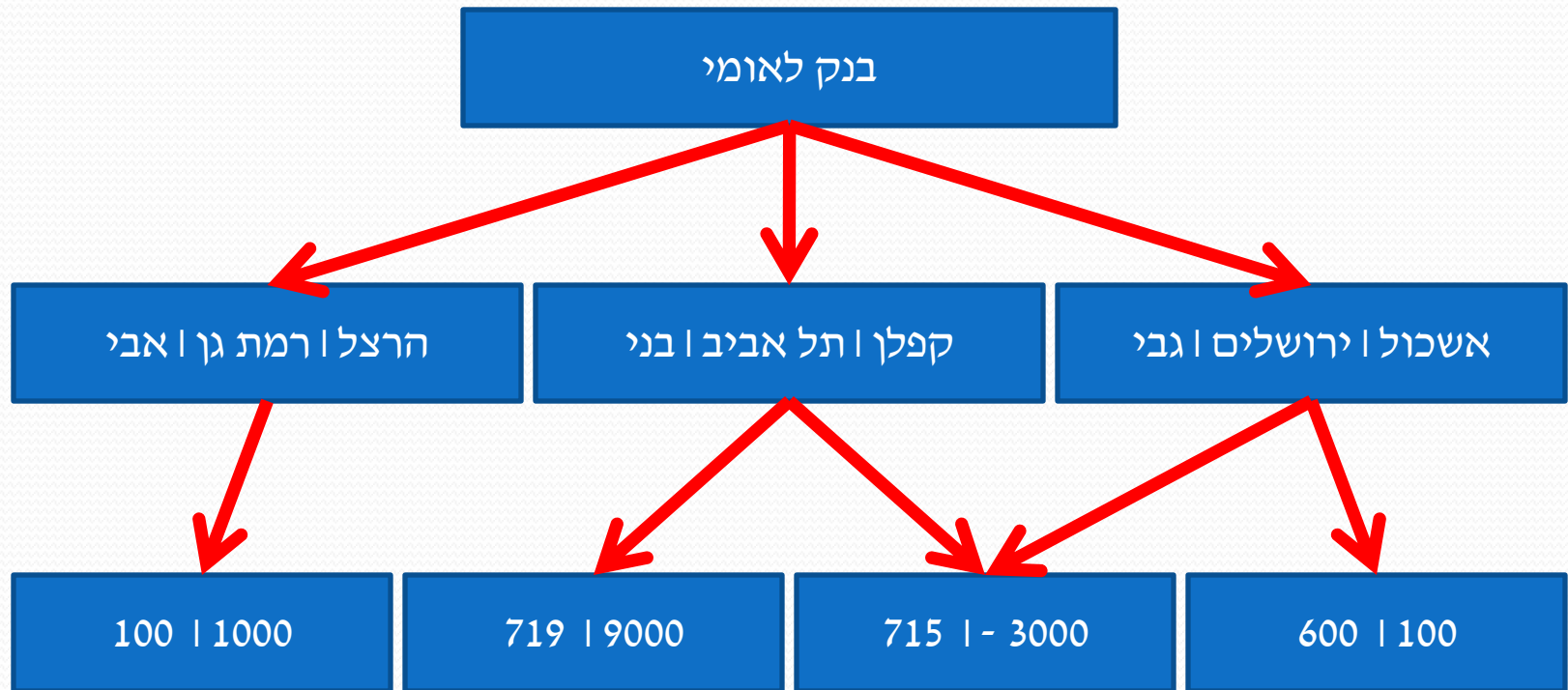
ID	מחלקה	משפחה	פרטי
1	89	קידר	מתן
2	89	רוזנברג	עמיעד

- מפתח - שדה \ מס' שדות המזהה באופן חד משמעי את הרשומה. (אין בטבלה שתי רשומות שונות עם אותו המפתח!) קשרים בין רשומות המופיעות בטבלאות שונות מתבצעים בעזרת שדות המפתח.
- שם טבלה - כל טבלה מזוהה בצורה חד משמעית ע"י שם ייחודי המשמש את בסיס הנתונים ע"מ לאתר את הטבלה. ע"מ להשתמש בטבלה יש לדעת את שמה בלבד.
- הערה: במודל הטבלאי ניתן לשמור בטבלאות גם מידע לניהול בסיס הנתונים עצמו (למשל: הרשאות).

דוגמא – מודל היררכי



דוגמא – מודל הרישתי



אם נרצה למחוק את בני, נצטרך למחוק את החשבונות שלו
אך לשמור על החשבונות שלו הקשורים לאנשים אחרים.

דוגמא – מודל יחסי \ טבלאי

טבלת לקוחות			
שם	עיר	רחוב	מס' חשבון
אבי	רמת גן	הרצל	100
בני	תל אביב	קפלן	719
בני	תל אביב	קפלן	715
גבי	ירושלים	אשכול	715
גבי	ירושלים	אשכול	600

טבלת חשבוניות	
מס' חשבון	יתרה
100	1000
719	9000
715	-3000
600	100

נוכל למחוק שורה מכל טבלה
בלי קשר לתלות של אותה השורה בטבלה השניה.

אלגברת יחסים (רלציונית)

- כאמור, המודל הנמצא בשימוש כיום הוא המודל היחסי \ טבלאי.
- כעת נראה כיצד נשלף מידע מתוך היחסים (טבלאות) במודל.
- שליפת המידע תתבצע ע"י פעולות לוגיות ממשפחת אלגברת היחסים (רלציונית).
- האלגברה הרלציונית מתבססת על תורת הקבוצות ומהווה את **המודל המתמטי** עבור ביצוע פעולות על רלציות (יחסים \ טבלאות).
- סגורה תחת סט הפעולות שנלמד (כלומר – פעולות האופרטורים מופעלות על טבלאות והתשובות המתקבלות הם טבלאות).

אלגברת יחסים (רלציונית) - טבלאות

חשבונות בנק - account		
branch-name	account-number	balance
Downtown	A-101	500
Mianus	A-215	700
Perryridge	A-102	400
Round Hill	A-305	350
Brighton	A-201	900
Redwood	A-222	700
Brighton	A-217	750

מפקידים - Depositor	
customer-name	account-number
Johnson	A-101
Smith	A-215
Hayes	A-102
Turner	A-305
Johnson	A-201
Jones	A-217
Lindsay	A-222

סניפים - Branch		
branch-name	branch-city	assets
Downtown	Brooklyn	9000000
Redwood	Palo Alto	2100000
Perryridge	Horseneck	1700000
Mianus	Horseneck	400000
Round Hill	Horseneck	8000000
Pownal	Bennington	300000
North Town	Rye	3700000
Brighton	Brooklyn	7100000

לקוחות - Customer		
customer-name	customer-street	customer-city
Jones	Main	Harrison
Smith	North	Rye
Hayes	Main	Harrison
Curry	North	Rye
Lindsay	Park	Pittsfield
Turner	Putnam	Stamford
Williams	Nassau	Princeton
Adams	Spring	Pittsfield
Johnson	Alma	Palo Alto
Glenn	Sand Hill	Woodside
Brooks	Senator	Brooklyn
Green	Walnut	Stamford

הלוואות - loan		
branch-name	loan-number	amount
Downtown	L-17	1000
Redwood	L-23	2000
Perryridge	L-15	1500
Downtown	L-14	1500
Mianus	L-93	500
Round Hill	L-11	900
Perryridge	L-16	1300

לווים - borrower	
customer-name	loan-number
Jones	L-17
Smith	L-23
Hayes	L-15
Jackson	L-14
Curry	L-93
Smith	L-11
Williams	L-17
Adams	L-16

בחירה $\sigma_P(R)$ - Select

- פעולה אונארית הבוחרת מתוך יחס R את ה- n -יות (tuples) המקיימות פרדיקט (משפט לוגי) נתון P .
- הפעולה מחזירה יחס בעל סכימה זהה לזו של R . מספר השורות ביחס המוחזר **קטן או שווה** למספר השורות ביחס R .
(נשים לב לכך שפעולה מופעלת על יחס ומחזירה יחס)
- הפרדיקט P יכול להכיל את:
 - הסימנים $< > \geq \leq \neq =$
 - הקשרים הלוגיים $\vee \wedge \neg$

$\sigma_p(R)$ - בחירה Select

• דוגמא 1:

כל ההלוואות בהן הסכום גדול מ-1200

• $\sigma_{\text{amount} > 1200}(\text{loan})$

loan		
branch-name	loan-number	amount
Downtown	L-17	1000
Redwood	L-23	2000
Perryridge	L-15	1500
Downtown	L-14	1500
Mianus	L-93	500
Round Hill	L-11	900
Perryridge	L-16	1300



loan		
branch-name	loan-number	amount
Redwood	L-23	2000
Perryridge	L-15	1500
Downtown	L-14	1500
Perryridge	L-16	1300

$\sigma_p(R)$ - בחירה Select

• דוגמא 2 :

כל ההלוואות בהן הסכום גדול מ-1200
שבוצעו בסניף Downtown

• $\sigma_{\text{amount} > 1200 \wedge \text{branch-name} = \text{'Downtown'}}(\text{loan})$

loan		
branch-name	loan-number	amount
Downtown	L-17	1000
Redwood	L-23	2000
Perryridge	L-15	1500
Downtown	L-14	1500
Mianus	L-93	500
Round Hill	L-11	900
Perryridge	L-16	1300



loan		
branch-name	loan-number	amount
Downtown	L-14	1500

$\sigma_p(R)$ - בחירה Select

• דוגמא 3 :

כל ההלוואות בהן הסכום גדול מ-1200
שבוצעו בסניף Downtown – דרך נוספת.

• $\sigma_{\text{amount} > 1200} (\sigma_{\text{branch-name} = \text{'Downtown'}}(\text{loan}))$

loan		
branch-name	loan-number	amount
Downtown	L-17	1000
Redwood	L-23	2000
Perryridge	L-15	1500
Downtown	L-14	1500
Mianus	L-93	500
Round Hill	L-11	900
Perryridge	L-16	1300



loan		
branch-name	loan-number	amount
Downtown	L-17	1000
Downtown	L-14	1500



loan		
branch-name	loan-number	amount
Downtown	L-14	1500

הטלה $\pi_A(R)$ - Project

- פעולה אונארית המחזירה יחס המכיל רשימת תכונות נבחרות (A) מתוך התכונות של יחס R.
- בתוצאה המוחזרת יהיה מספר שורות זהה או קטן ממספר השורות ב R.
- הסיבה שיכולה להתקבל תוצאה עם מס' שורות קטן יותר היא במקרה שהתקבלו שורות זהות.
- מכיון שיחס היא קבוצה, שורות כפולות מופיעות רק פעם אחת.

הטלה $\pi_A(R)$ - Project

• דוגמא 1:

הצג את שמות הלקוחות והערים בהם הם גרים.

• $\pi_{\text{customer-name, customer-city}}(\text{customer})$

customer		
customer-name	customer-street	customer-city
Jones	Main	Harrison
Smith	North	Rye
Hayes	Main	Harrison
Curry	North	Rye
Lindsay	Park	Pittsfield
Turner	Putnam	Stamford
Williams	Nassau	Princeton
Adams	Spring	Pittsfield
Johnson	Alma	Palo Alto
Glenn	Sand Hill	Woodside
Brooks	Senator	Brooklyn
Green	Walnut	Stamford



customer	
customer-name	customer-city
Jones	Harrison
Smith	Rye
Hayes	Harrison
Curry	Rye
Lindsay	Pittsfield
Turner	Stamford
Williams	Princeton
Adams	Pittsfield
Johnson	Palo Alto
Glenn	Woodside
Brooks	Brooklyn
Green	Stamford

הטלה $\pi_A(R)$ - Project

• דוגמא 2 :

הצג את שמות הסניפים בהם בוצעו הלוואות בסכום הגבוה מ- 1200.

• $\pi_{\text{branch-name}}(\sigma_{\text{amount} > 1200}(\text{loan}))$

loan		
branch-name	loan-number	amount
Downtown	L-17	1000
Redwood	L-23	2000
Perryridge	L-15	1500
Downtown	L-14	1500
Mianus	L-93	500
Round Hill	L-11	900
Perryridge	L-16	1300



loan		
branch-name	loan-number	amount
Redwood	L-23	2000
Perryridge	L-15	1500
Downtown	L-14	1500
Perryridge	L-16	1300



loan		
branch-name		
Redwood		
Perryridge		
Downtown		

שימו לב לכך שמספר הסניפים הצטמצם, למרות שלא ביקשנו להוריד רשומות. הסיבה לכך היא שהייתה לנו כפילות בשם הסניף, ומכיוון שיחס הוא בעצם קבוצה – אנחנו לא מכילים רשומות כפולות.

הטלה $\pi_A(R)$ - Project

• דוגמא 2 :

הצג את שמות הסניפים בהם בוצעו הלוואות בסכום הגבוה מ-1200.
האם ניתן להפוך את סדר הפעולות?

• $\sigma_{\text{amount} > 1200} (\pi_{\text{branch-name}}(\text{loan}))$

loan		
branch-name	loan-number	amount
Downtown	L-17	1000
Redwood	L-23	2000
Perryridge	L-15	1500
Downtown	L-14	1500
Mianus	L-93	500
Round Hill	L-11	900
Perryridge	L-16	1300



loan		
branch-name		
Downtown		
Redwood		
Perryridge		
Mianus		
Round Hill		



איחוד Union - U

- פעולה בינארית היוצרת יחס המכיל את כל ה n -יות המופיעות לפחות באחד משני היחסים.
- סימון: $R_1 \cup R_2$
- לפני שמבצעים איחוד, יש לוודא שהיחסים אותם רוצים לאחד הם תואמים (*compatible*)- כלומר מתקיימים התנאים הבאים:
 - לשני היחסים יש אותה דרגה, כלומר אותו מספר תכונות.
 - תחום התכונה ה i -ית ב R_1 זהה לתחום התכונה ה i -ית ב R_2 .
- שמות הכותרת של השדות עליהם מתבצע האיחוד לא חייבות להיות זהות.
- בתוצאה המוחזרת:
 - לא מופיעות שורות כפולות
 - ליחס המוחזר, במקומות שבהם שמות העמודות שונות – אין סכימה, כלומר אין שמות לעמודות ואי אפשר לבצע עליו פעולות (מלבד איחוד והפרש).

איחוד Union - U

• דוגמא 1 :

מצא את שמות כל הלקוחות שהפקידו כסף בבנק, או לקחו הלוואה או גם וגם.

• $\pi_{\text{customer-name}}(\text{depositor}) \cup \pi_{\text{customer-name}}(\text{borrower})$

depositor
customer-name
Johnson
Smith
Hayes
Turner
Jones
Lindsay

U

borrower
customer-name
Jones
Smith
Hayes
Jackson
Curry
Williams
Adams

=

customer-name
Johnson
Smith
Hayes
Turner
Jones
Lindsay
Jackson
Curry
Williams
Adams

הפרש Difference - “-”

- פעולה בינארית המחזירה את ה n -יות הנמצאות ביחס אחד, אך לא באחר.
- סימון: $R_1 - R_2$
- בדומה לאיחוד, לפני שמבצעים הפרש - יש לוודא ששני היחסים תואמים.
- בתוצאה המוחזרת:
- לא מופיעות שורות כפולות
- ליחס המוחזר, במקומות שבהם שמות העמודות שונות – אין סכימה, כלומר אין שמות לעמודות ואי אפשר לבצע עליו פעולות (מלבד איחוד והפרש).

הפרש Difference - “-”

• דוגמא:

מצא את שמות כל הלקוחות של הבנק שלא לקחו הלוואה.

• $\pi_{\text{customer-name}}(\text{customer}) - \pi_{\text{customer-name}}(\text{borrower})$

customer
customer-name
Jones
Smith
Hayes
Curry
Lindsay
Turner
Williams
Adams
Johnson
Glenn
Brooks
Green

=

borrower
customer-name
Jones
Smith
Hayes
Jackson
Curry
Williams
Adams

=

customer-name
Lindsay
Turner
Johnson
Glenn
Brooks
Green