



ניתוח מערכת - מדרישות למודלים

חלק ב'

בשבוע שעבר...

- התחלנו את שלב הניתוח בתהליך פיתוח תוכנה
- 3 גישות ניתוח
- דיאגרמות:
 - DFD (לא בשימוש כיום)
 - Use Case Diagram
 - ERD (מודל סוג 1)
 - Activity Diagram
 - State Machine Diagram

מערכת לניהול חשבונות בנק

מערכת לניהול בנקים- סניפים, לקוחות, חשבונות
והלוואות:

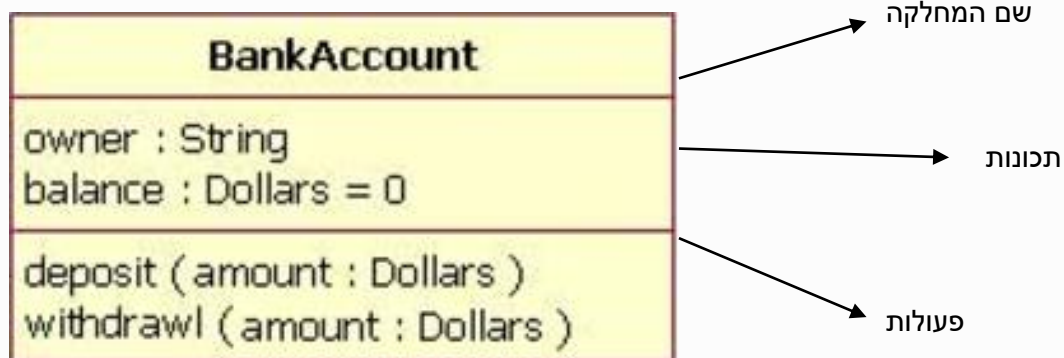
- בכל בנק יש הרבה סניפים.
- בכל סניף יש הרבה חשבונות.
- חשבון בנק יכול להיות מ-2 סוגים- חשבון חיסכון וחשבון עו"ש
- ללקוח יש חשבון בנק מאחד הסוגים או את שניהם
- בכל חשבון יכול להיות רשום רק לקוח אחד
- הלוואות משויכות ללקוח ולסניף בנק (בנפרד מחשבונות הבנק)

Class Diagram

- מודל של מבנה המערכת ע"י מידול המחלקות, התכונות והפעולות.
- UML class diagram הוא תכנית האב של המחלקות הדרושות לבניית התוכנה
- מתכנתים מפתחים את המערכת בהתבסס על מודל המחלקות שהוגדר
- זהו המודל הנפוץ ביותר ב-UML

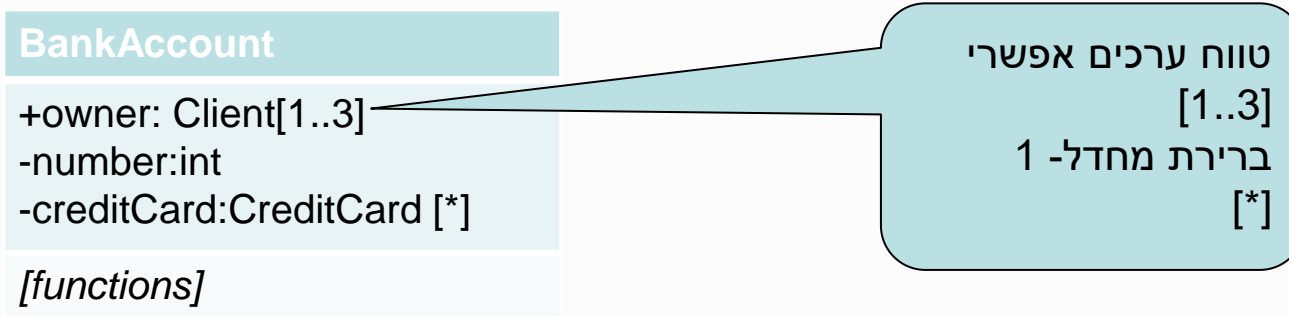
סימון מחלקה

- A simple class:



Multiplicity

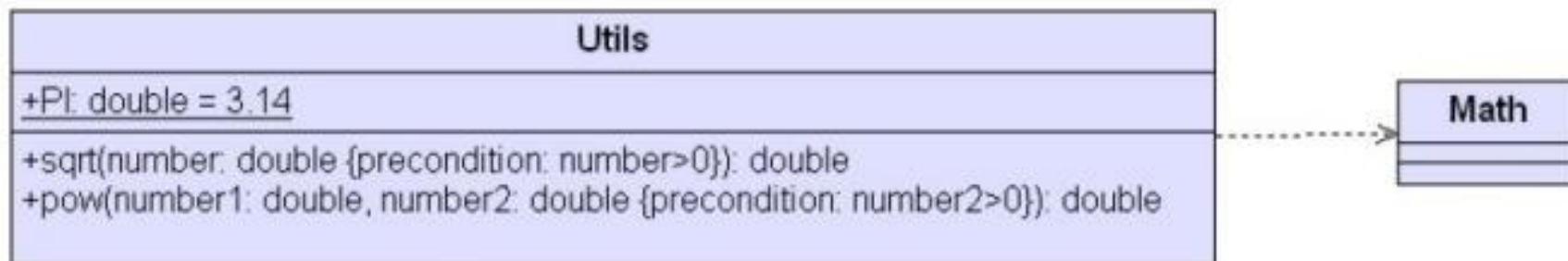
- מייד לאחר ציון ה- type של המשתנה ניתן לציין את ה- multiplicity שלו.



קשרים בין מחלקות

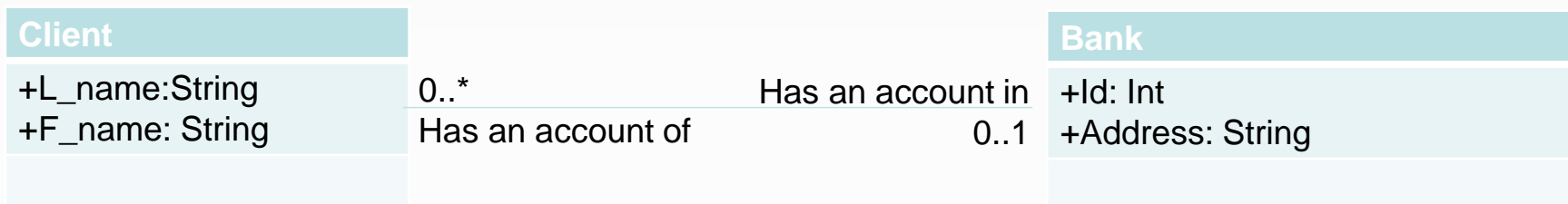
- Dependency
 - Association
 - Aggregation
 - Composition

קשר של Dependency



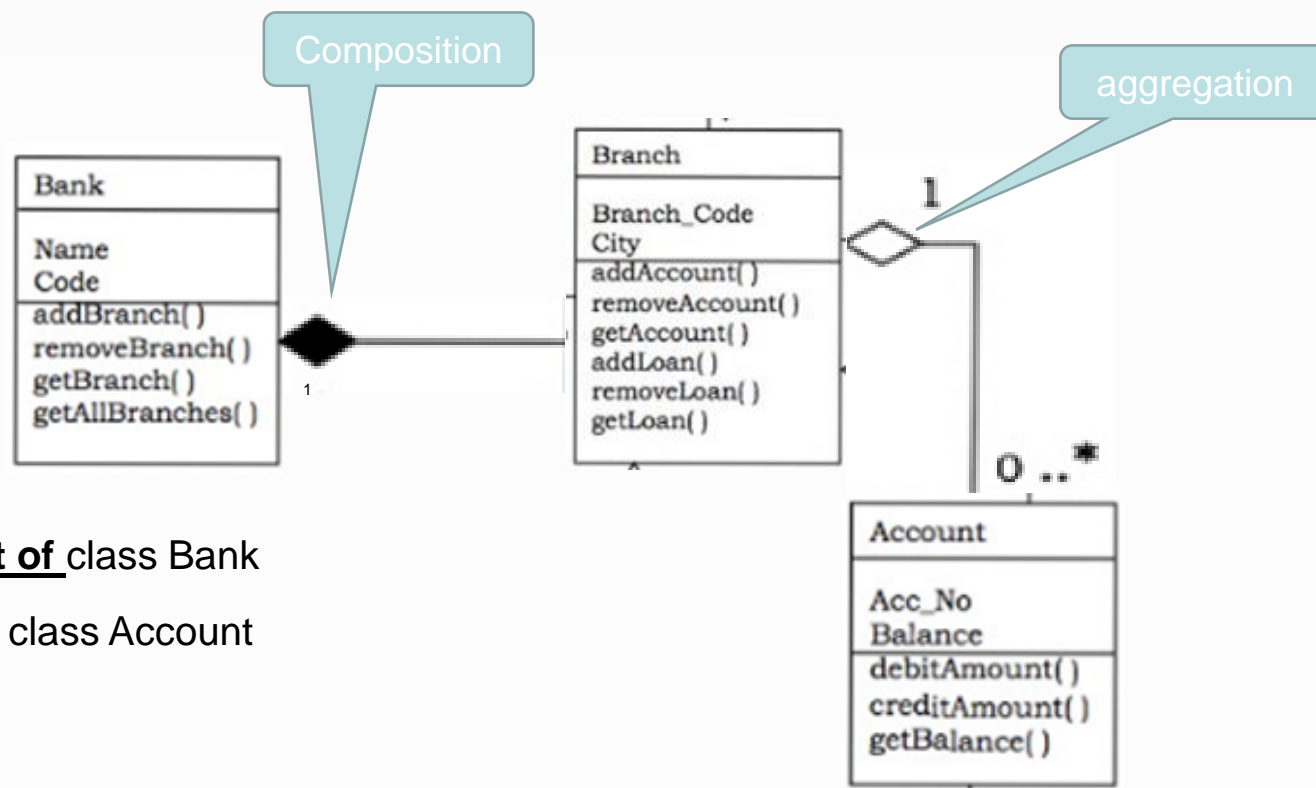
class Utils uses class Math

קשר של Association



class Bank has class Client

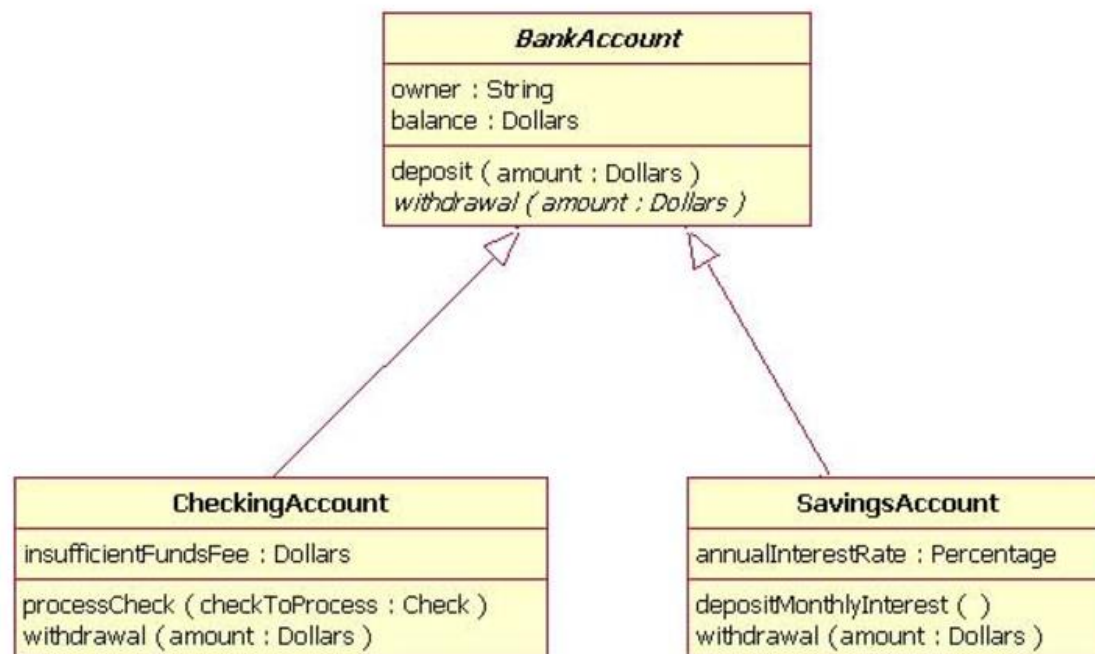
קשרים של Aggregation ו- Composition



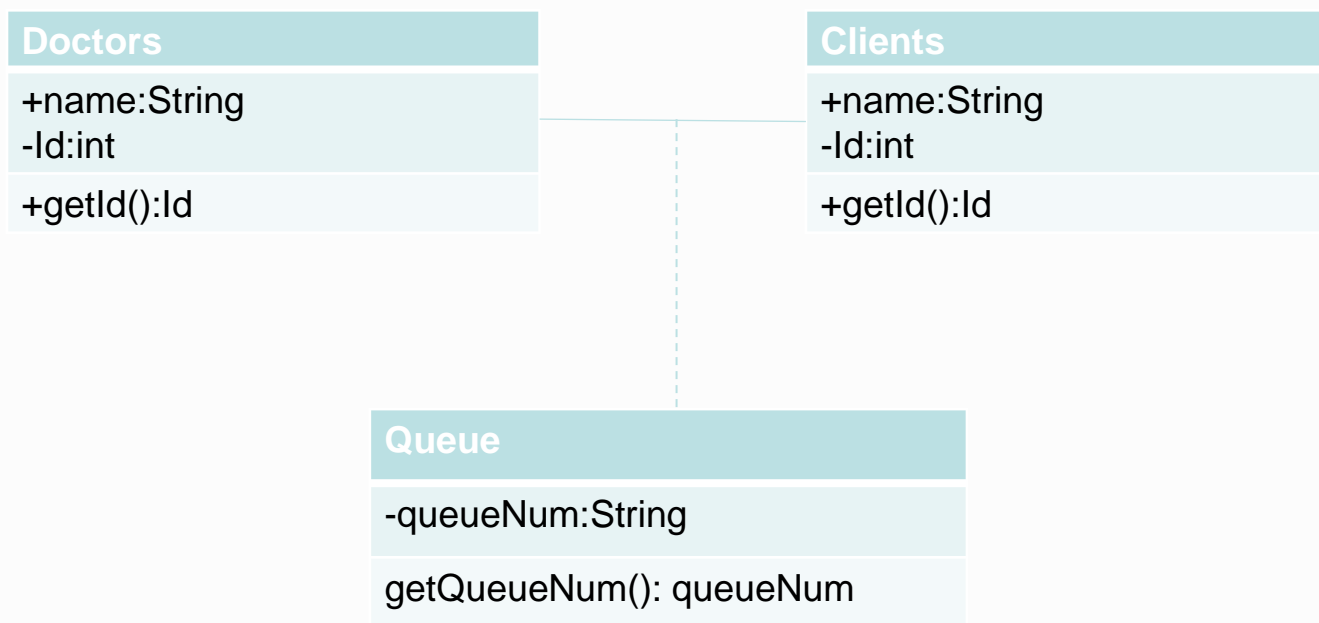
class Branch **is part of** class Bank

Class Branch **owns** class Account

הורשה



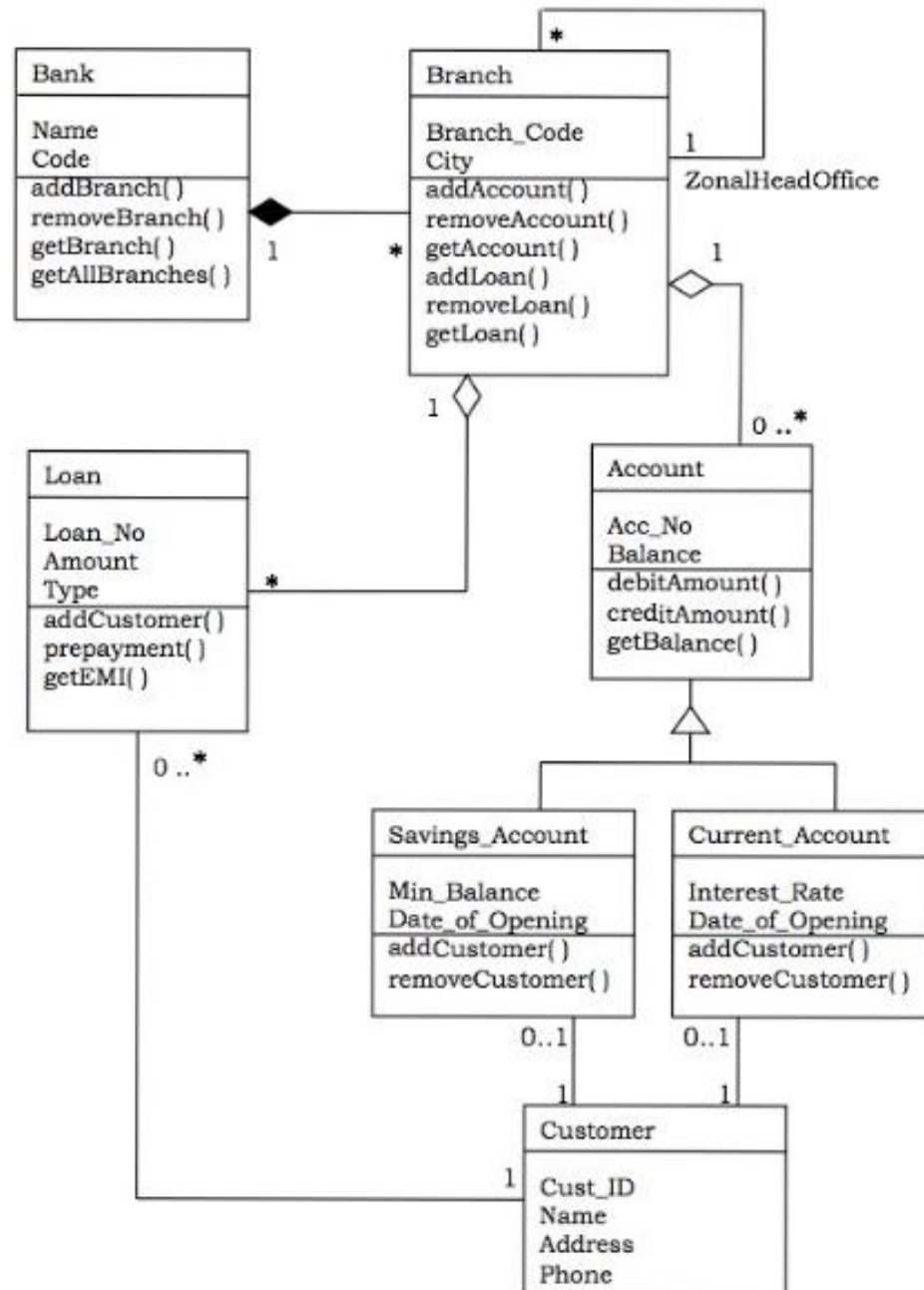
מחלקת קשר



תיאור Interfaces ומחלקות שמיישמים אותם



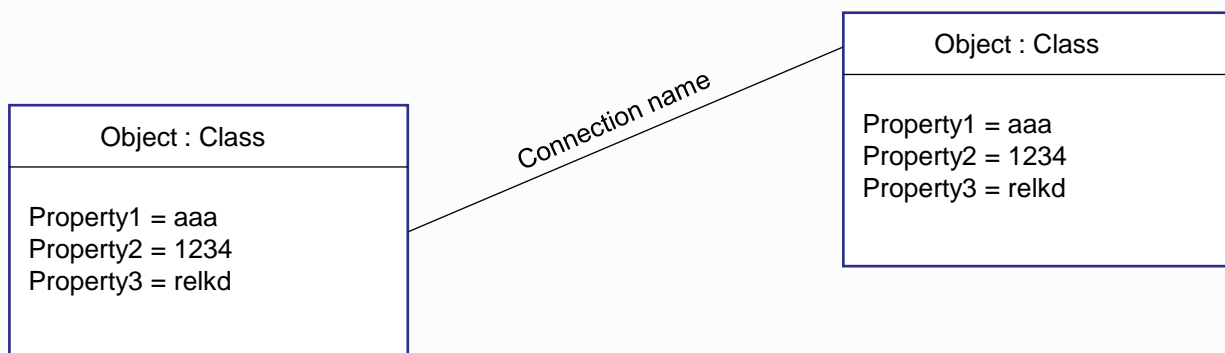
Class Diagram



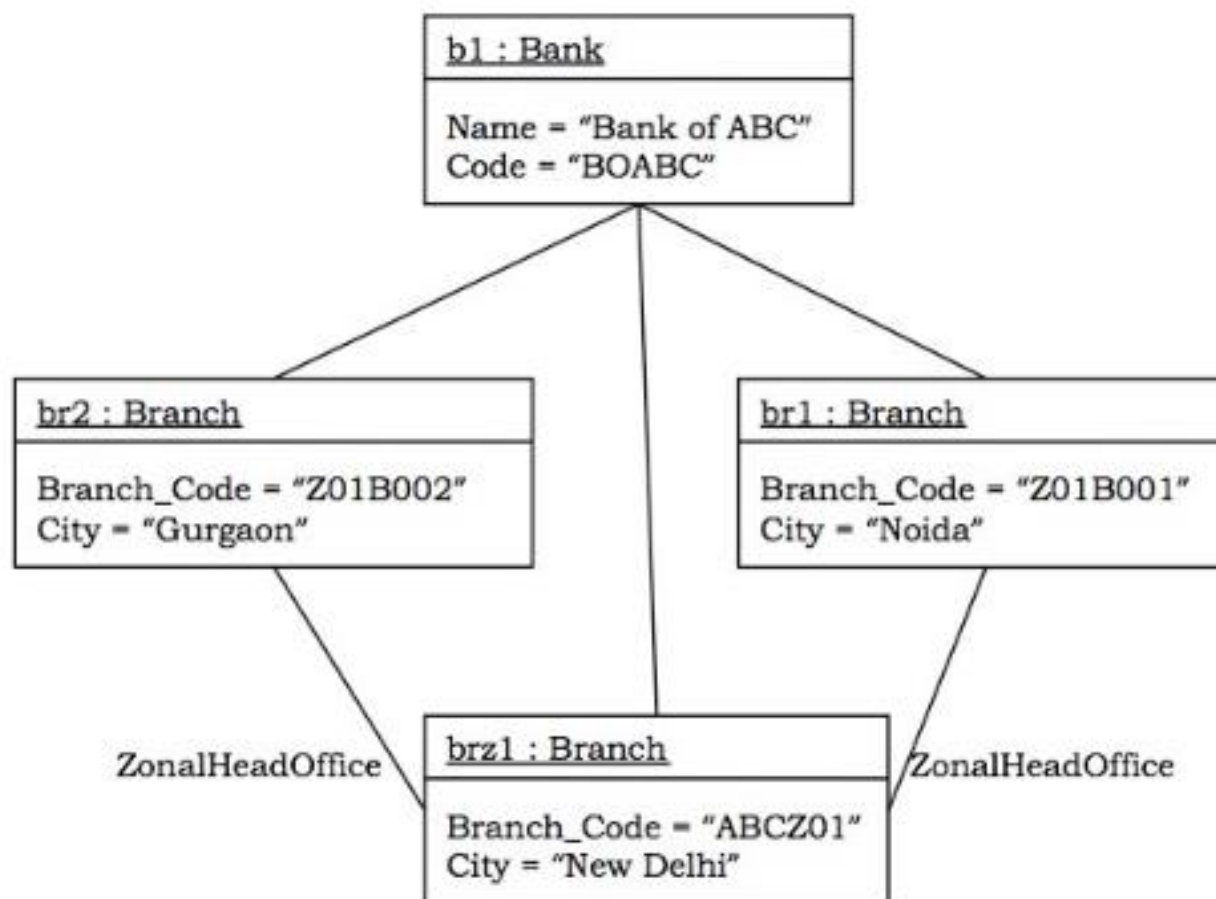
Object Diagram

- דיאגרמת אובייקטים מתארת מופע של מודל המחלקות, בדומה לתרחישים אמיתיים שעל בסיסם בונים את המערכת.
- דיאגרמת אובייקטים היא מודל סטטי (בדומה למודל המחלקות)
- השימוש במודל האובייקטים דומה למודל המחלקות רק שהם מאפשרים בניית אב טיפוס של המערכת מנקודת מבט מעשית

שיטת סימון



Object Diagram



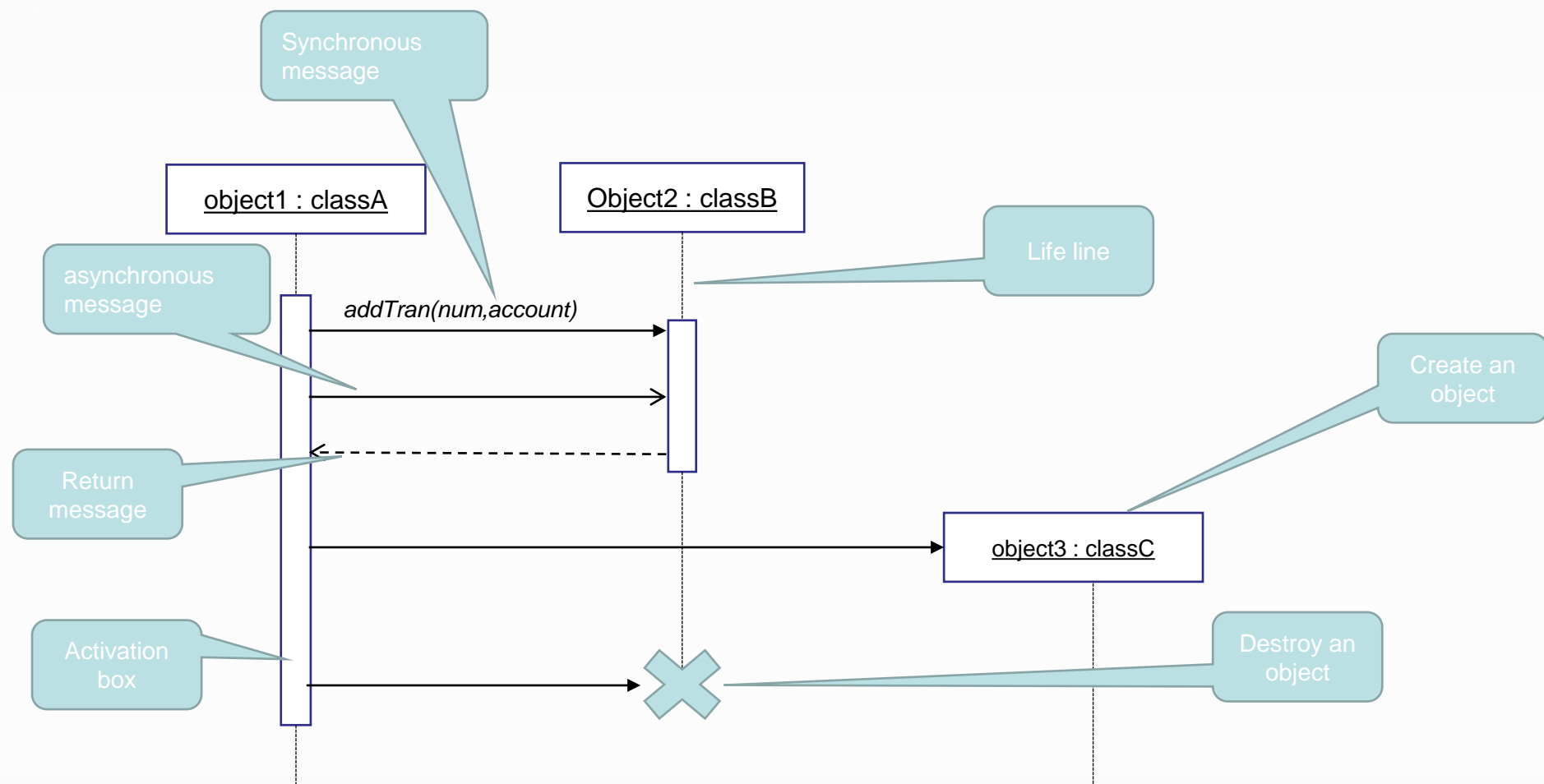
Class Diagram Vs. Object Diagram

Object diagram	Class diagram	
מכיל 2 חלקים: שם ורשימת מאפיינים	מכיל 3 חלקים: שם, רשימת תכונות ורשימת פעולות	מבנה
הפורמט מורכב משם האובייקט + נקודותיים + שם המחלקה (Tom:Employee)	שם המחלקה עומד בפני עצמו בחלק של שם המחלקה	שם המחלקה
מגדיר את הערך הנוכחי של כל תכונה	מגדיר את התכונות של המחלקה	רשימת התכונות
לא כלולות	כלולות	רשימת הפעולות
מוגדר שם הקשר, אבל לא הכמות (לא רלוונטי כשמדברים על ישות בודדת)	מוגדר הקשר בין מחלקות- שם הקשר וכמות הקשר.	קשרים

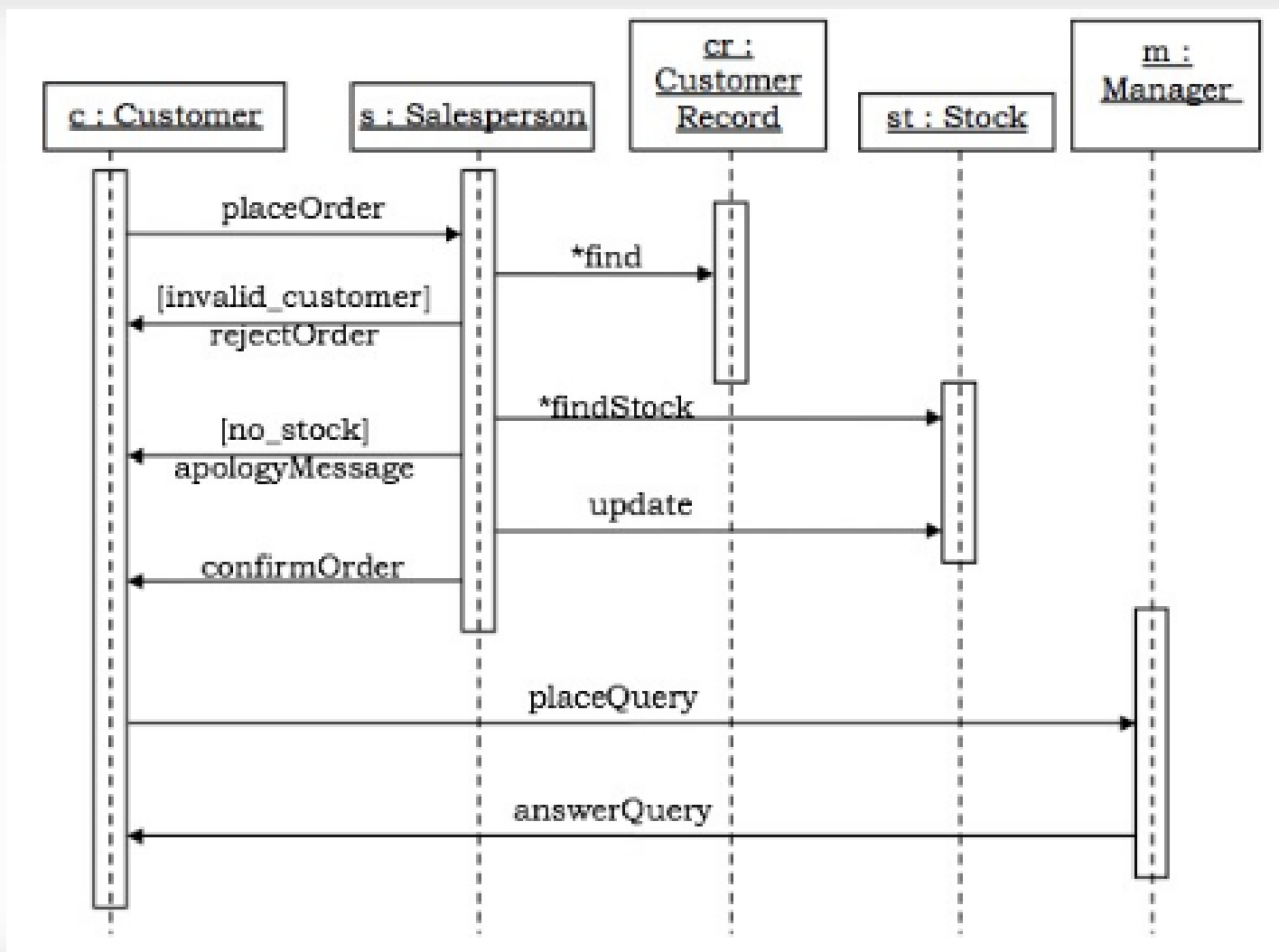
Sequence Diagram

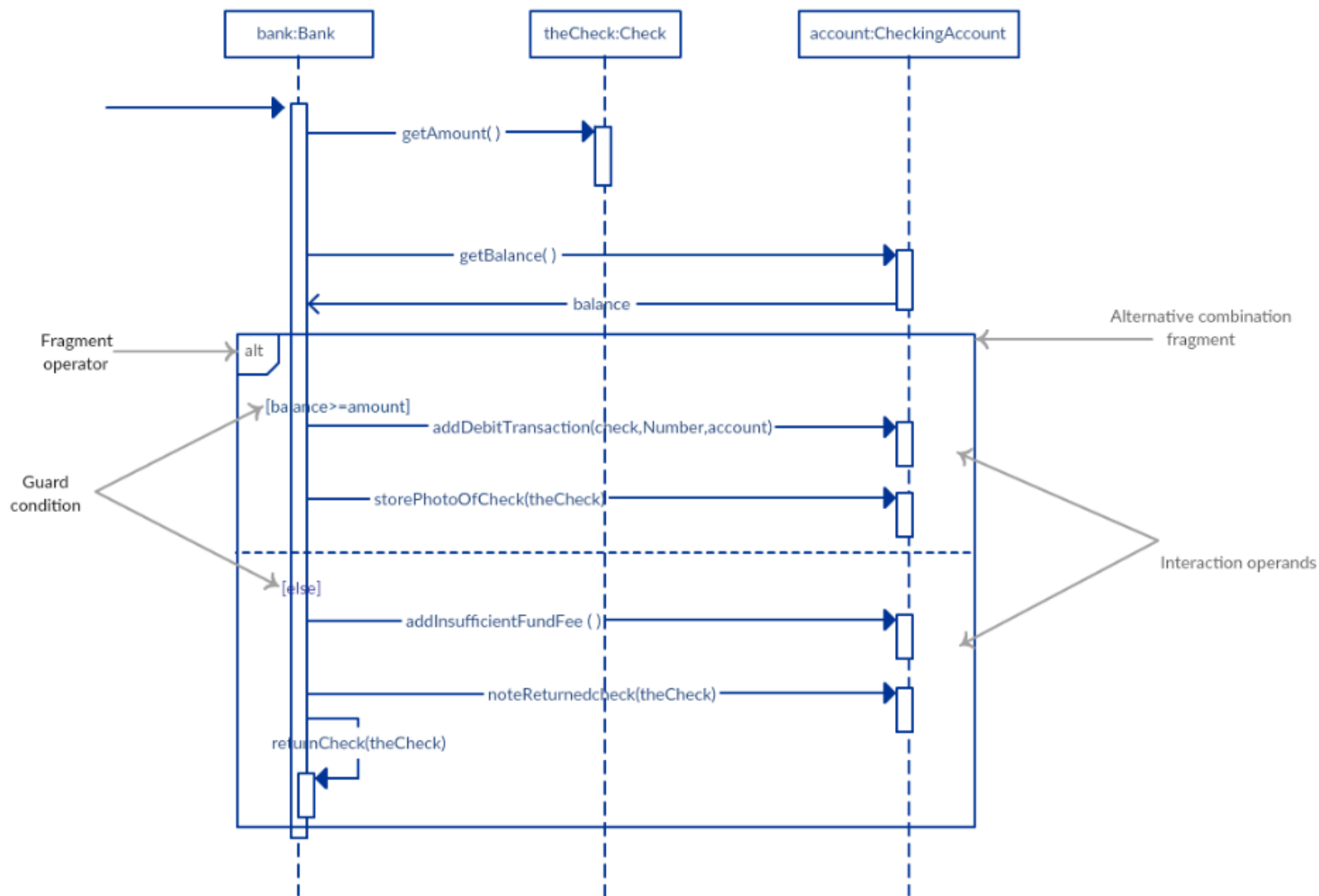
- תרשים רצף הממחיש את סדר הפעילויות ברצף של הזמן.
- תרשים רשת נכתב בצורה של תרשים דו מימדי
 - על ציר ה- X נמצאים האובייקטים
 - על ציר ה- Y ממוקמות ההודעות שהאובייקטים האלה שולחים
- לכל פונקציונליות נכין תרשים רצף נפרד

שיטת הסימון



Sequence Diagram





Activity Diagram Vs. Sequence Diagram

Activity	Sequence
דיאגרמת התנהגות	דיאגרמת אינטראקציה
מתמקד בתהליך של האובייקטים ונותן דגש לרצף ולתנאים שיש בתהליך	מתמקד בהודעות שמועברות בין ישויות במערכת.
סדר ביצוע הפעולות לא מודגש	נותן דגש לסדר ביצוע הפעולות
כללי יותר ופחות מדויק	התהליך יותר מדויק ומפורט

דיאגרמות ה-UML

ניתן לחלק את הדיאגרמות הקיימות ב-uml ל-2:

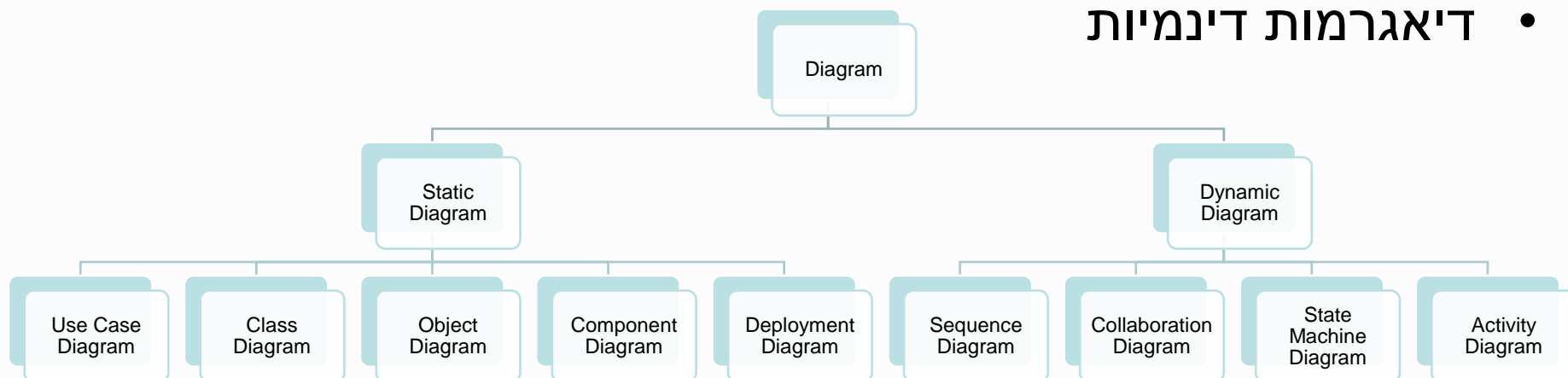
- דיאגרמות המתארות מבנה
- דיאגרמות המתארות התנהגות ואינטראקציה



דיאגרמות ה-UML

חלוקה נוספת:

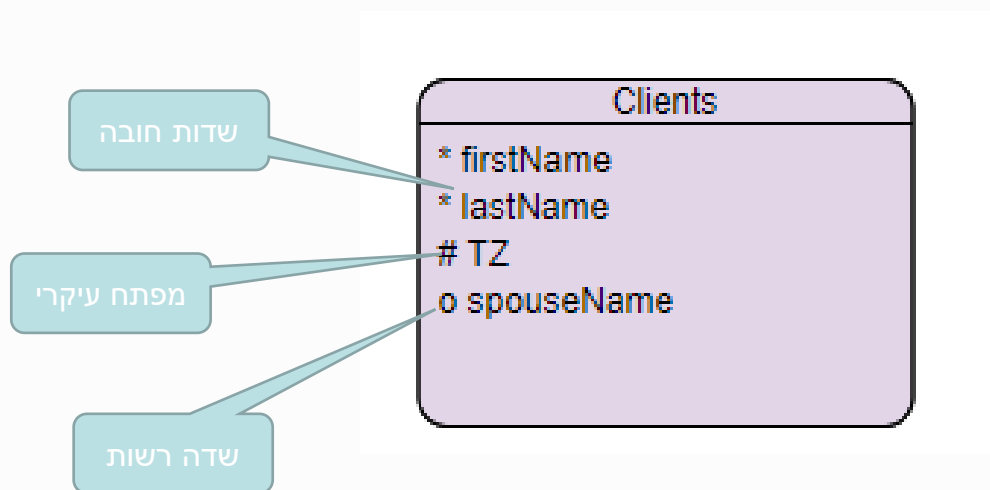
- דיאגרמות סטטיות
- דיאגרמות דינמיות





ERD

הצגת יישות במודל



קשרים בין ישויות - Relationship

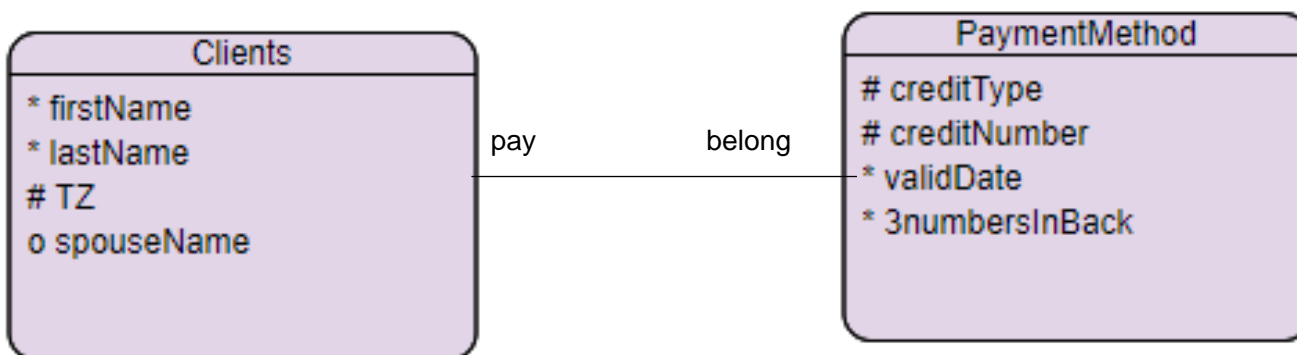
קשר מוגדר כיחס בעל משמעות בין ישויות שונות.
פונקציונליות הקשר- מוגדרת כסוג המיפוי הקיים בין הקבוצות
המשתתפות בקשר. הפונקציונליות יכולה להיות מסוג- $1:M$, $1:1$,
 $M:N$

דוגמאות-

- עבור הישויות – מרפאה ולקוח קיים קשר- לקוח שייך למרפאה אחת, במרפאה יש הרבה לקוחות.
- עבור הישויות- מזכירה ועמדות עבודה קיים קשר- מזכירה עובדת בעמדת עבודה ועמדת עבודה שייכת למזכירה.

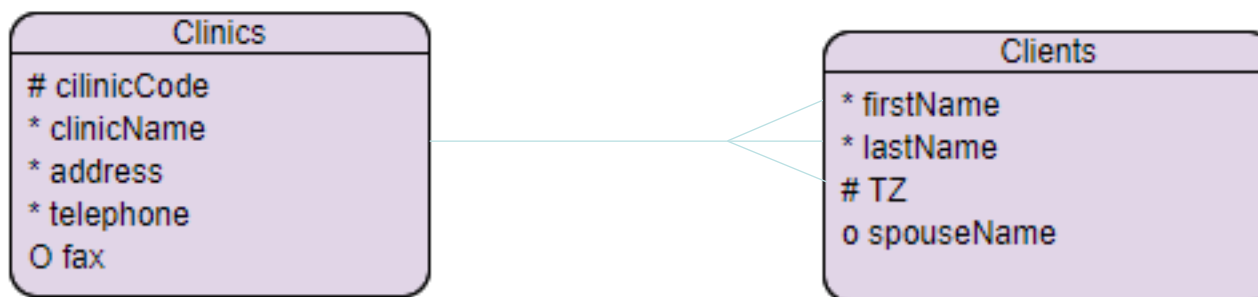
קשר חד חד ערכי (1:1)

- ישות אחת מתאימה לישות אחת אחרת
לדוגמא-
• לכל לקוח מוגדר אמצעי תשלום יחיד וכל אמצעי תשלום שייכת ללקוח אחד בלבד



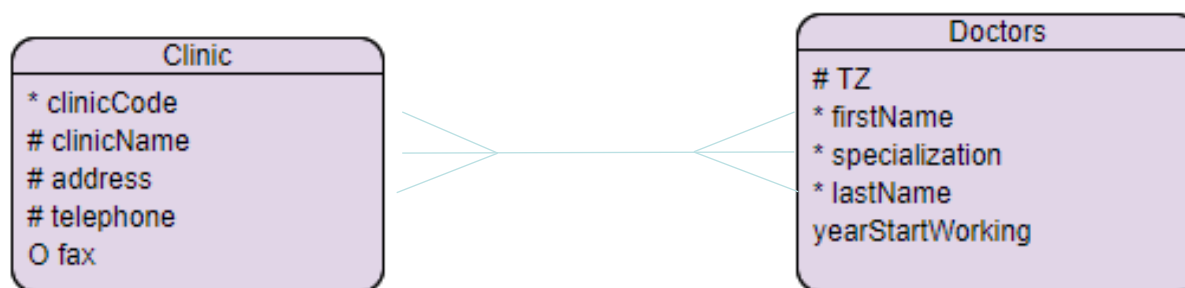
קשר חד רב ערכי (1:M)

- ישות אחת מתאימה לכמה ישויות בקבוצה אחרת
לדוגמא-
• כל לקוח משויך למרפאה אחת ובכל מרפאה יש הרבה לקוחות



קשר רב רב ערכי (M:N)

- ישות אחת מתאימה לכמה ישויות בקבוצה אחרת, וישות בקבוצה האחרת מתאימה לכמה ישויות בקבוצה הראשונה.
- לדוגמא -
רופא יכול לעבוד בכמה מרפאות ובכל מרפאה עובדים הרבה רופאים



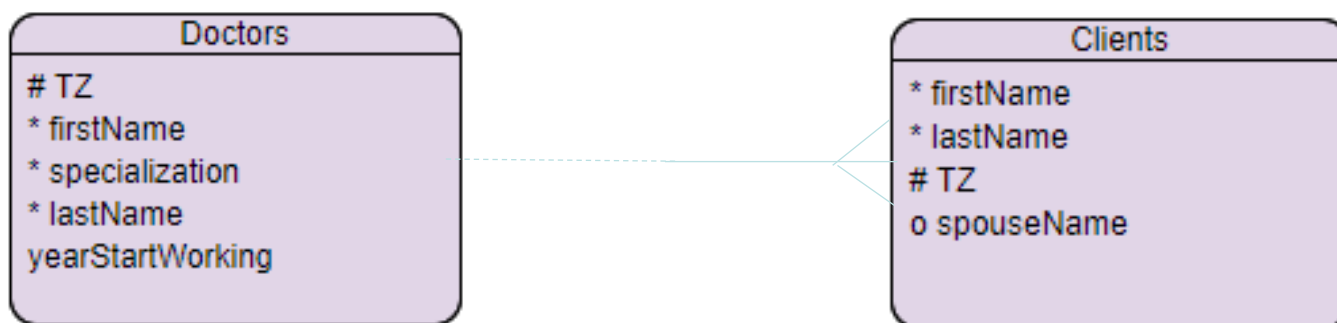
קרדינליות הקשר

Relationship cardinality

קרדינליות הקשר מוגדרת כמספר הישיות המינימלי והמקסימלי בקבוצת ישויות A הקשורות לישות אחת בקבוצת ישויות B.

לדוגמא-

- רופא משפחה יכול להיות לטפל בעד 100 לקוחות.

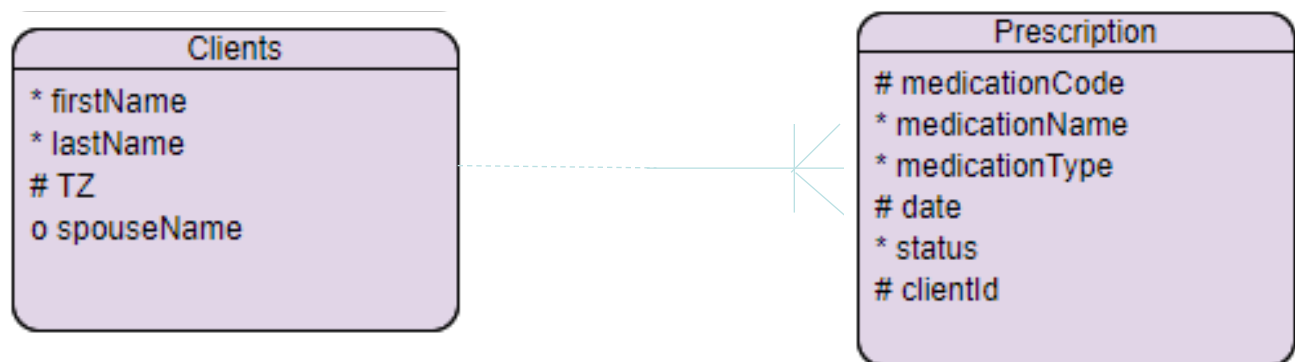


תלות קיומית existence dependence

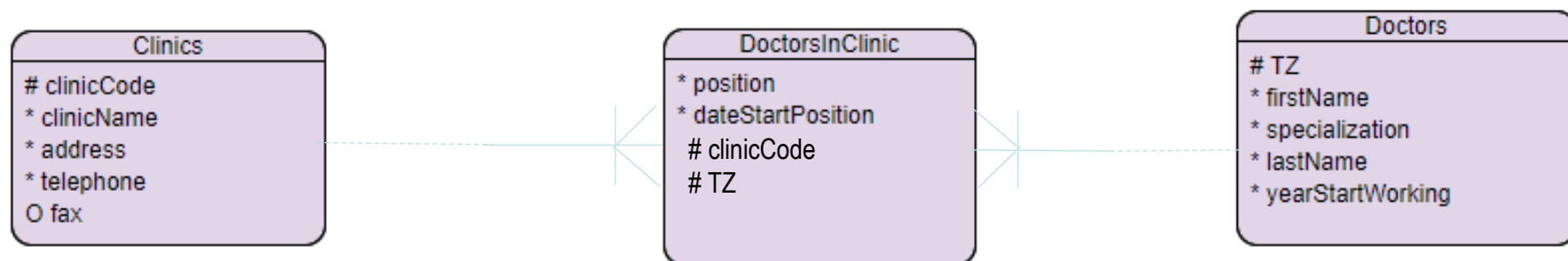
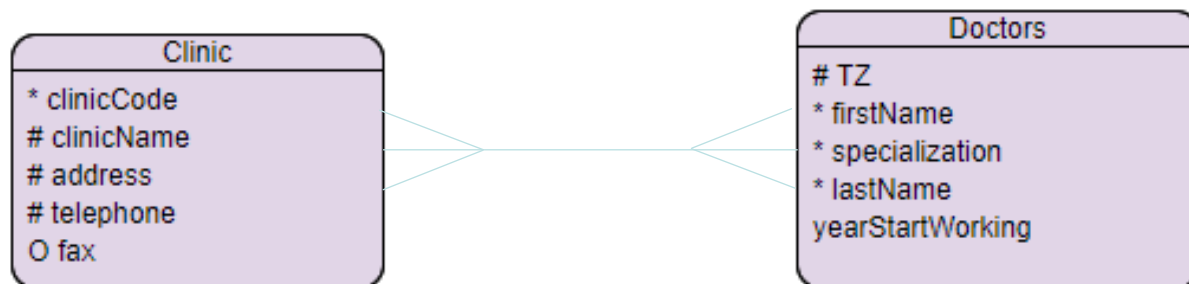
תלות קיומית בין קבוצה A לקבוצה B מוגדרת כמצב שבו קיום ישות בקבוצת ישויות A מותנה בקיום ישות בקבוצת ישויות אחרת B

לדוגמא-

- לטבלה "מרשמים ללקוח" אין משמעות כשהלקוח עזב את הקופה וכבר לא מבוטח שלה



From m:n to m:1 + n:1

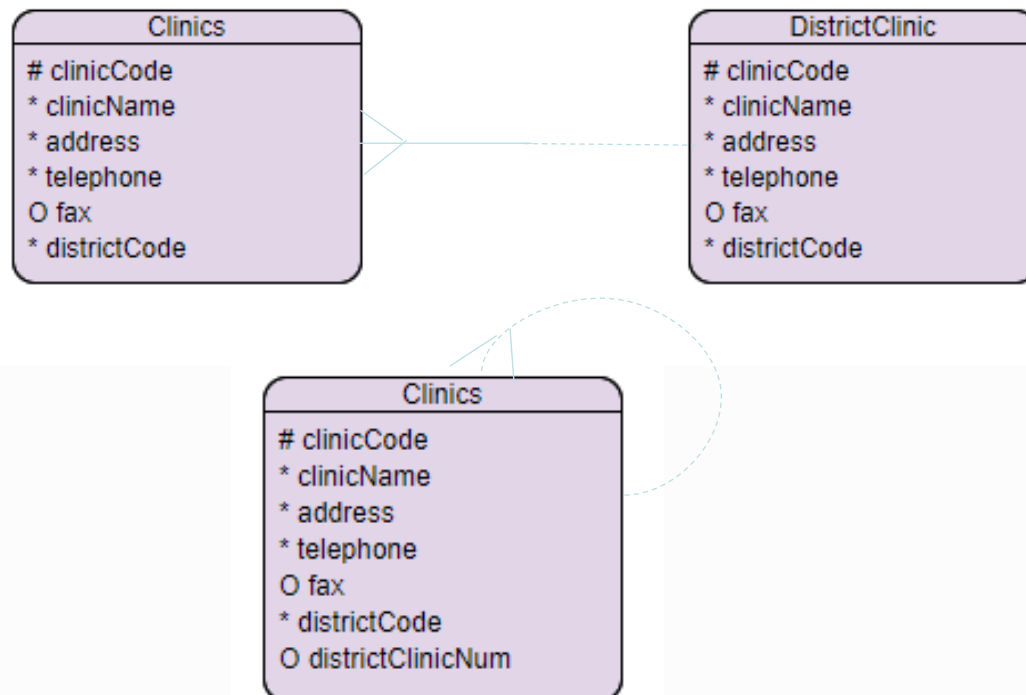


קשר רקורסיבי

קשר רקורסיבי מוגדר כקשר המחבר בין קבוצה A לעצמה

לדוגמא-

• מרפאה מחוזית

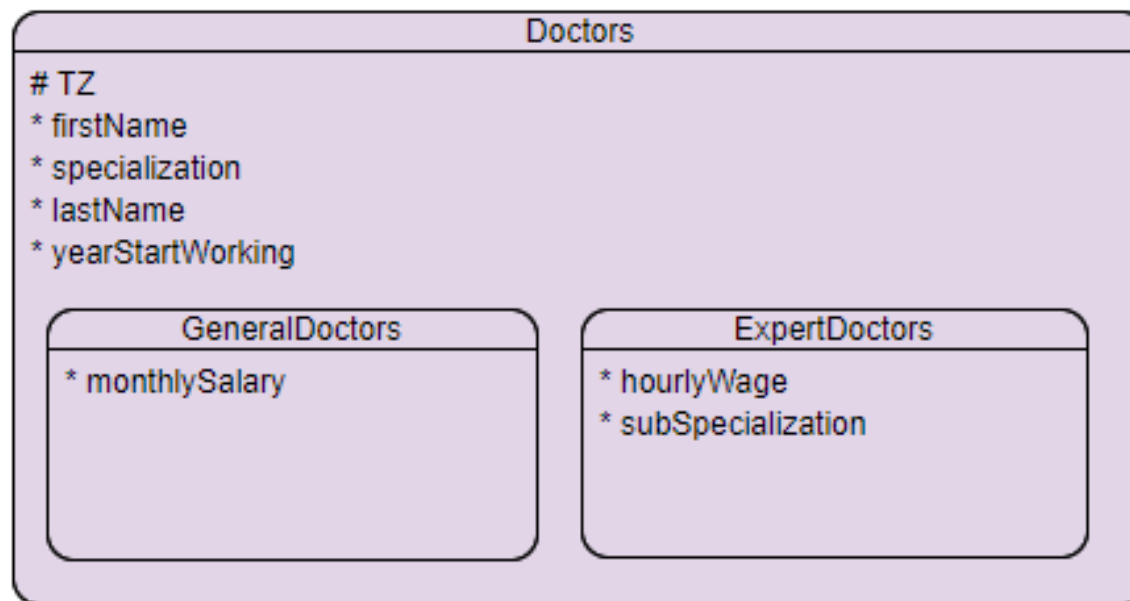


ישות על ותת ישות

תת ישות הינה סוג של ישות אשר יורשת את מאפייני ישות העל

לדוגמא-

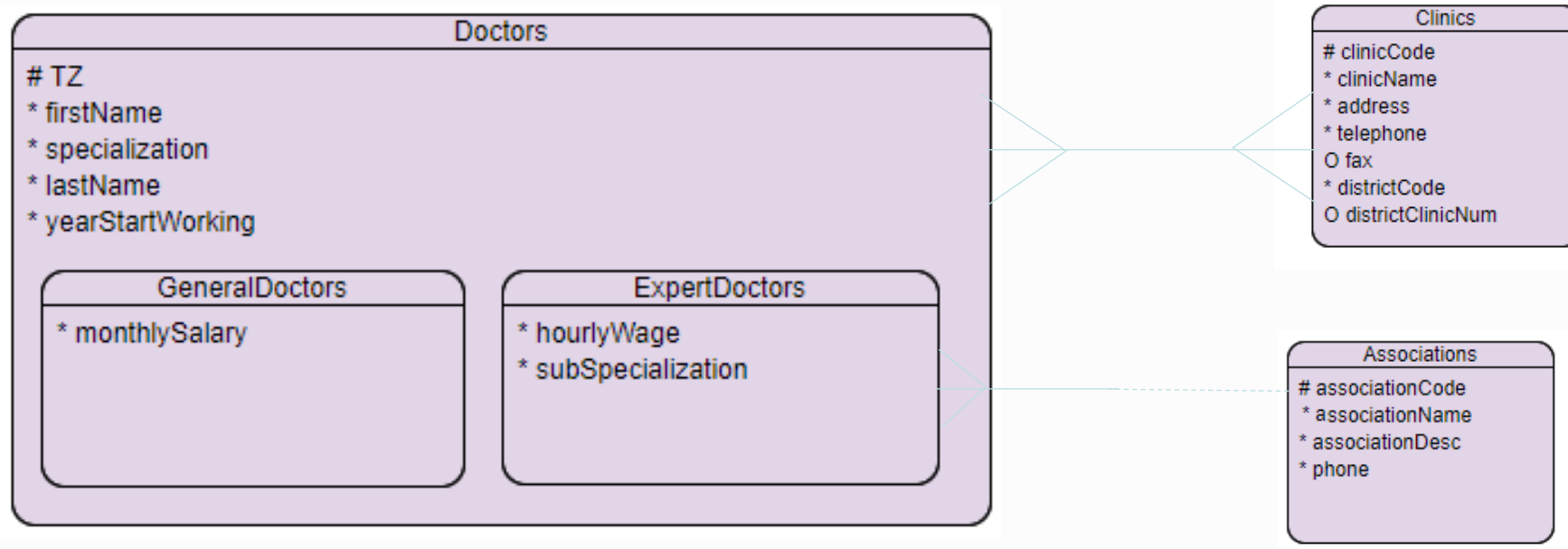
- כל רופא חייב להיות או רופא כללי או רופא מומחה



קשרים, ישויות על ותתי ישויות

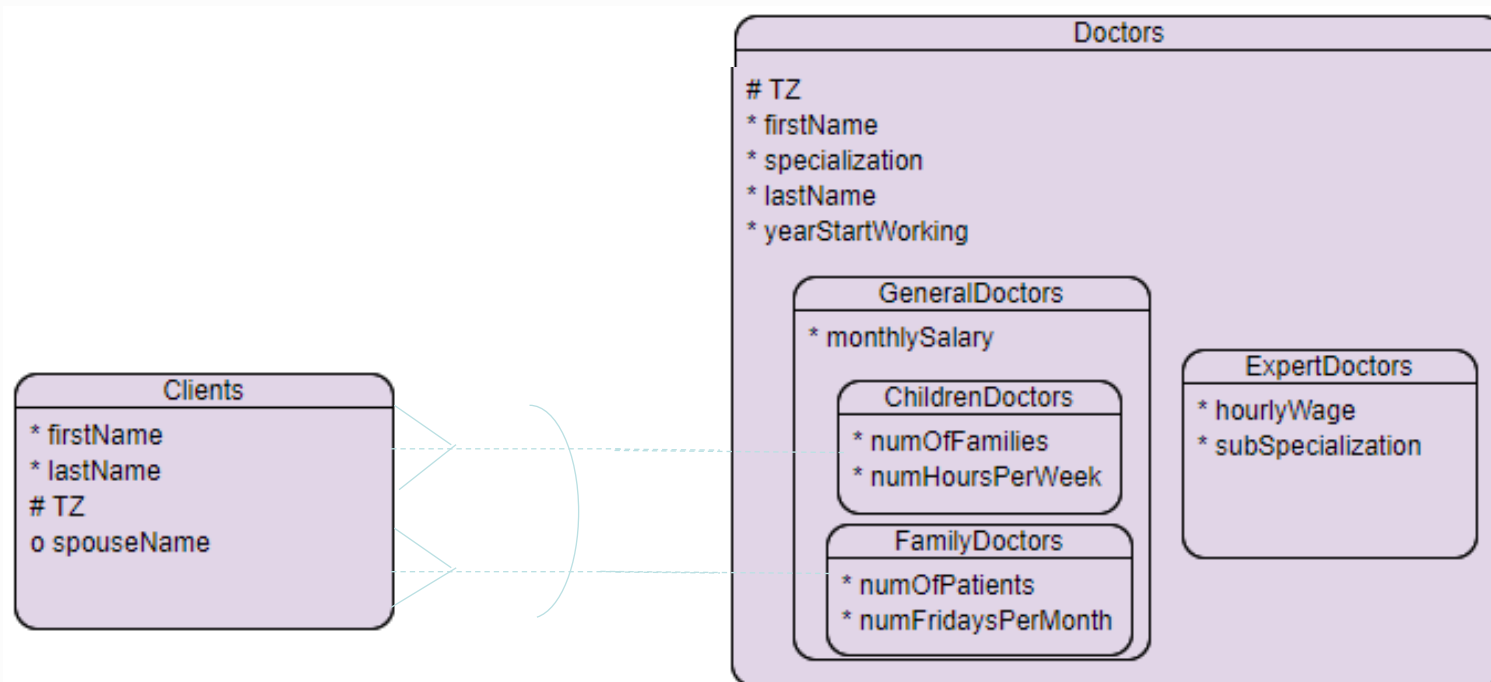
קשר של ישות העל הינו גם קשר של תת הישויות.
קשר של תת ישות מהווה קשר שלה עצמה בלבד.

• דוגמא-

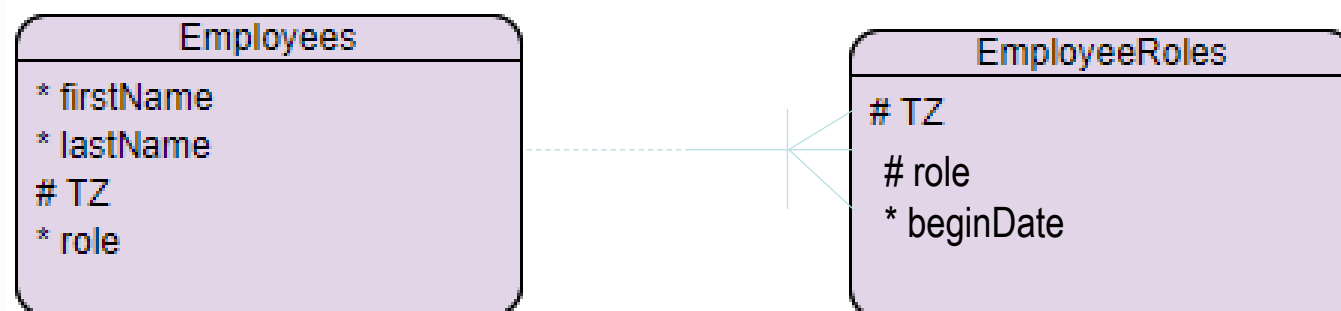
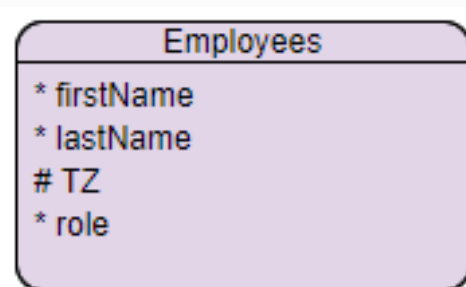


יחס בחירה בין קשרים

יחס בחירה בין מספר קשרים מוגדר כבחירה בקיום קשר אחד בלבד מתוך מס' אפשריים



חיי ישות לאורך זמן



שלבים ביצירת מודל ה-ER שלנו

- בהינתן תיאור המערכת הרצויה / רשימת הדרישות:
- זיהוי הישויות
 - הגדרת הקשרים וסוגי הקשרים בין הישויות
 - הגדרת קרדינליות הקשר
 - זיהוי התכונות וסוג כל תכונה ותכונה
 - יצירת ERD ראשוני
 - בדיקת רמת הנירמול של המודל ותיקונו עד קבלת מודל ברמת הנירמול המבוקשת