# Data Types

## טיפוסי נתונים:

טיפוס נתונים הוא מושג בשפת תכנות המתאר את סוגו של משתנה ומגדיר אילו ערכים יוכל לקבל ובאילו דרכים.

במסדי נתונים כל עמודה הינה משתנה ולכל אחת **טיפוס נתונים** (מוגדר בעת יצירת טבלה ועמודותיה) אשר מגדיר את סוגי הנתונים שיכולים להתקבל לעמודה זו, צורת עבודה זו מאפשרת לDB לשמור על **אמינות המידע** (לדוגמה: למנוע כניסה של שם משפחה לעמודה של גיל) ולעבוד בצורה יעילה (כל טיפוס הנותנים יודעים מראש ומעובדים האופטימלית ביותר).

ב–PostgreSQL יש לנו כמה משפחות של טיפוסי נתונים, טיפוסי הנתונים בכל משפחה דומים אחד לשני בסוג המידע איתו הם עובדים אך כל אחד בעל מטרה שונה.

טיפוסי נתונים הקיימים ב PostgreSQL:

* Boolean
* Character
* Numeric
* Temporal

\*ישנם טיפוסי נתונים נוספים ב – PostgreSQL אך לא נעמיק בהם.

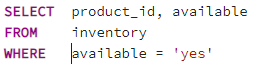
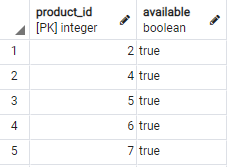
**Boolean**

בוליאני – ביטוי בעל לוגיקה של כן/לא. טיפוס בוליאני סטנדרטי יכול לקבל שלושה ערכים:

.TRUE, FALSE, NULL

זאת אומרת הוא יכול להיות כן, לא או ריק (במידה ולא הוגדר אחרת(.

\*טיפוס בוליאני בPostgreSQL- גמיש יותר ומקבל גם ערכים כמו t/f, 1/0, yes/no אך ה - DB יתרגם אותם לTrue וFalse וכך גם יציג אותם.



**Character**

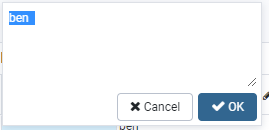
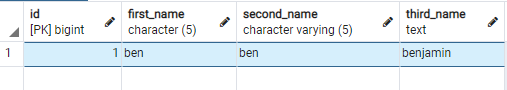
PostgreSQL מציע לנו שלושה טיפוסים עיקריים להתמודדות עם טקסט: CHAR, VARCHAR וTEXT - .

**CHARACTER VARYING(n) –** טיפוס זה מקבל כל טקסט וניתן להגבלה על ידי n, במידה ולא צוין n, לא חלה הגבלה על כמות הטקסט המתקבל. במידה ויש רווח לאחר הטקסט שעובר את ההגבלה, PostgreSQL יחתוך את הרווח כך שהטקסט יעמוד בהגבלה.

**CHARACTER(n) –** טיפוס זה מקבל כל טקסט וניתן להגבילו על ידי n, במידה ולא צוין n, חלה הגבלה של תו 1. במידה ויש רווח לאחר הטקסט שעובר את ההגבלה, PostgreSQL יחתוך את הרווח כך שהטקסט יעמוד בהגבלה, במידה ואורך הטקסט קצר מn-, PostgreSQL יוסיף רווח בסוף הטקסט שיתקבל כך שיהיה באורך של n.

**TEXT –** טיפוס זה מקבל כל טקסט ללא הגבלה.

**\*אין הבדלי ביצועים בין שלושת הטיפוסים**



**Numeric**

תחת הטיפוסים הנומריים ישנם שתי משפחות, integers וfloats:

**INTEGERS** – מספרים שלמים:

* SMALLINT – טווח מספרים בין 32,768- ל 32,767+
* INTEGER – טווח מספרים בין 2,147,483,648- ל 2,147,483,647+
* BIGINT – טווח מספרים בין 9223372036854775808- ל 9223372036854775807+
* SERIAL – מספר הגדל אוטומטית משורה לשורה בטווח של 1 עד 2,147,483,647
* BIGSERIAL - מספר הגדל אוטומטית משורה לשורה בטווח של 1 עד 9,223,372,036,854,775,807

**\*ההבדלים בין הטיפוסים השונים הינם בטווח המספרים אותו הם מאפשרים וכיוצא בזה נפח זיכרון והביצועים.**

**\*הטיפוס INTEGER מציע את האיזון הטוב ביותר בין טווח המספרים ולנפח והביצועים.**

**FLOATS** –מספרים עשרוניים:

* REAL – מקבל מספר עשרוני וחותך אותו 6 ספרות לאחר הנקודה
* DOUBLE PRECISION – מקבל מספר עשרוני וחותך אותו עד 15 ספרות לאחר הנקודה
* NUMERIC – מספר עשרוני עד 131,072 ספרות לפני הנקודה

ו16383 ספרות אחרי הנקודה. טיפוס זה מקבל 2 פרמטרים numeric(p,s)

הפרמטר P מגדיר את כמות הספרות לפני ואחרי הנקודה בעוד הפרמטר S מגדיר

את כמות הספרות אחרי הנקודה.

**דוגמה**: נכניס את המספר 21332.232 אל טיפוס numeric(9,2) ß 21332.23.

במידה ולא הוגדר S וP הטיפוס מקבל מספרים עם S ו P שונים.

**\*real וdouble precision מייצגים קירוב למספר בעוד numeric מתייחס למספר מדויק.**

**\*numeric מתאים לחישובים מדויקים אך פעולות מתמטיות הינן איטיות יותר ביחס לטיפוסי נותנים אחרים.**

**Temporal**

טיפוסים זמניים, מאפשרים לנו להתמודד עם מידע המתייחס למימד זמן כמו תאריכים, שעות, פרקי זמן.

PostgreSQL מציע לנו 4טיפוסי נתונים זמניים מרכזים.

* + - **DATE –** טיפוס זה מקבל תאריך בכל מיני פורמטים שונים, של

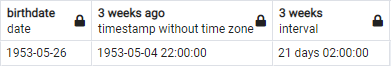
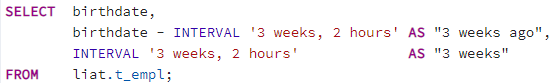
יום ספציפי, הDB יפרמט את התאריך שיתקבל לyyyy-mm-dd, ניתן להציג את המידע בפורמט שונה בעזרת פונקציית TO\_CHAR.

* + - **TIME –** טיפוס המקבל זמן ביום (ללא התאריך) בפורמטים שונים, הDB יפרמט את השעה ל HH:MM:SS
    - **TIMESTAMP –** טיפוס המתאר תאריך ושעה, למעשה צירוף של DATE וTIME
    - **INTERVAL –** טיפוס המאפשר לנו לעבוד עם פרקי זמן הנשמרים בהיררכיה של שנים,

חודשים, ימים, שעות והלאה, לדוגמה 2 years, 5 months, 3 days, 8 hours, 12 minutes. הטיפוס interval שימושי לפעולות חישוב זמנים.

נכריז על interval בצירוף (+ -, לשם ביצוע הפעולה) ולאחר מכן נרשום מספר המתייחס

לכמות בצרוף של יחידת מידה, לדוגמה:



כל אחד הוא טיפוס

צוות מסדי נתונים.