**[Error! Hyperlink reference not valid.]( )**

**הצוות:**

* **ידידיה בקורדזה: 332461854**
* **מאיר קרומבי: 214736688**

**על הארגון:**

ברוכים הבאים למלון "אל הוטל לה פארמז'אן" מלון 5 כוכבים מפואר בלב מזרח ירוחם מקום פסטורלי בהחלט!  
  
בקובץ זה נתאר את הליך בניית בסיסי הנתונים למחלקת ניהול העובדים במלון.  
בסיס הנתונים יכלול את כל המידע על:  
1. העובדים במלון – עובדים, מנהלי המחלקות, מגייסים וכו'  
2. מחלקות המלון – ניקיון, אוכל, פרסום, לוגיסטיקה, הנהלת חשבונות, הנהלה וכו'  
3. אנשים שמגישים מועמדות לעבוד באחת ממחלקות המלון.  
4. משמרות – כמה משמרות כל עובד לקח ואיזה משמרות יש וכו'

**מיני"פ בסיסי נתונים – תשפ"ד**

**שלב א** – תיאור הארגון, מודל ERD, מודל DSD, סקריפטי SQL ואכלוס מידע:

**הגדרת הישויות:**

בסיס נתונים זה יכלול 7 ישויות כי הגדלנו ראש ואילו הן:

* **איש** - ישות אב הכוללת תכונות בסיסיות של הומוספיאנס שאנחנו.  
  ישות זו מכילה את התכונות הבאות:

1. **מספר זהות** – מפתח.
2. שם משפחה.
3. שם פרטי.
4. כתובת מגורים.
5. עיר מגורים.
6. אימייל
7. יום הולדת

* **עובד** – ישות המתארת את עובדים המלון, מנהלים מנקים ומה שביניהם  
  ישות זו יורשת מישות האיש את תכונותיה, ובנוסף:

1. מספר זהות – ירושה מישות "איש" – מפתח ראשי (זר)
2. שכר שעתי.
3. תפקיד.
4. תאריך הצטרפות לצוות המלון.

* **מועמד** – ישות המתארת את האנשים שהגישו מועמדות למשרה לאחת ממחלקות המלון.  
  עובדים אלה יכולים להגיש מועמדות למשרה בעבודה לבדם או על ידי המגייס שלהם (באם גוייסו על ידי אחד).  
  כל מועמד רשאי להגיש מועמדות למשרה אחת או יותר, לבד או עם מגייס ובלבד שלא יגיש ליותר ממשרה אחת למחלקה מסוימת!

1. מספר זהות - ירושה מישות "איש" – מהווה מפתח ראשי לטבלה זו.
2. [אופציונלי] מספר הזהות של המגייס – תכונה זו אפשרית ולכן ניתנת להיות בעלת ערך null-י הואיל וכל מועמד ראשי להגיש את עצמו לעבוד.
3. תאריך התמודדות (נמצא בקשר "Willing\_To\_Work\_A")

* **מנהל** - ישות המתארת את האנשים היושבים על הארגזים, אחראים על מחלקות ובעיקר פה כדי לקבל 50% יותר משכורת מכולם 🥺  
  ישות זו יורשת את תכונותיה מישות העובד ובנוסף מקבלת מפתח זר של "מספר מחלקה".
* **מגייס** - ישות המתארת את העובדים האחראים על גיוס עובדים חדשים לצוות המלון.  
  ישות זו יורשת את תכונותיה מישות העובד.
* **מחלקה** - ישות המתארת את המחלקות השונות במלון כגון מחלקת חשבונות ניקיון וכו'  
  לכל מחלקה מנהל אחד או יותר. (לפי הקשר שיוצג בתמונה בהמשך 😁)  
  **תכונות הישות:**

1. **מספר מחלקה** - מפתח.
2. שם מחלקה.

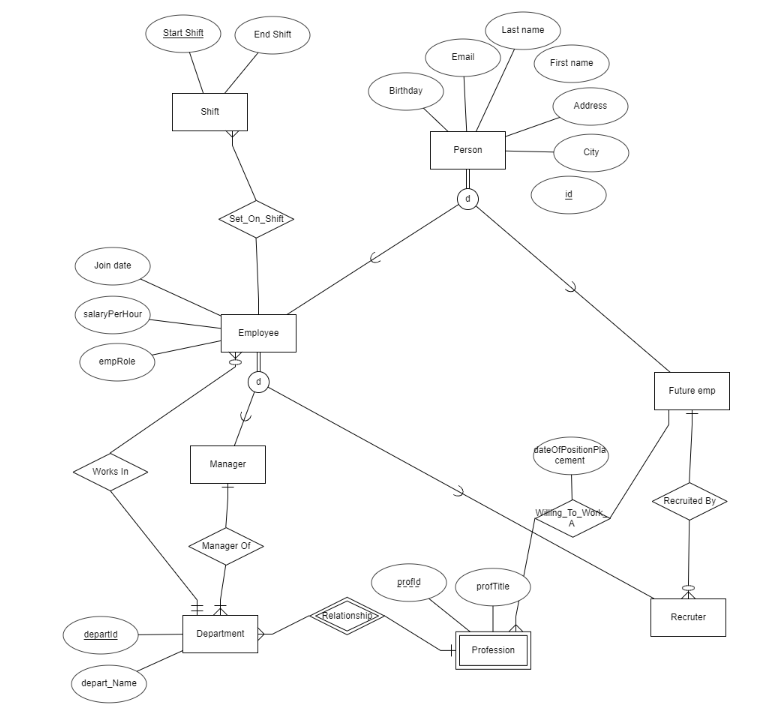
* **משמרות** - ישות המתארת את כל המשמרות שעובד במלון יכול לקחת.  
  **תכונות הישות:**

1. **שעת התחלה** – מפתח.
2. שעת סיום.

* **התמחות** – ישות המתארת את תחום העבודה של כל עובד במלון (כגון רואה חשבון ומזכירה)  
  **תכונות הישות:**

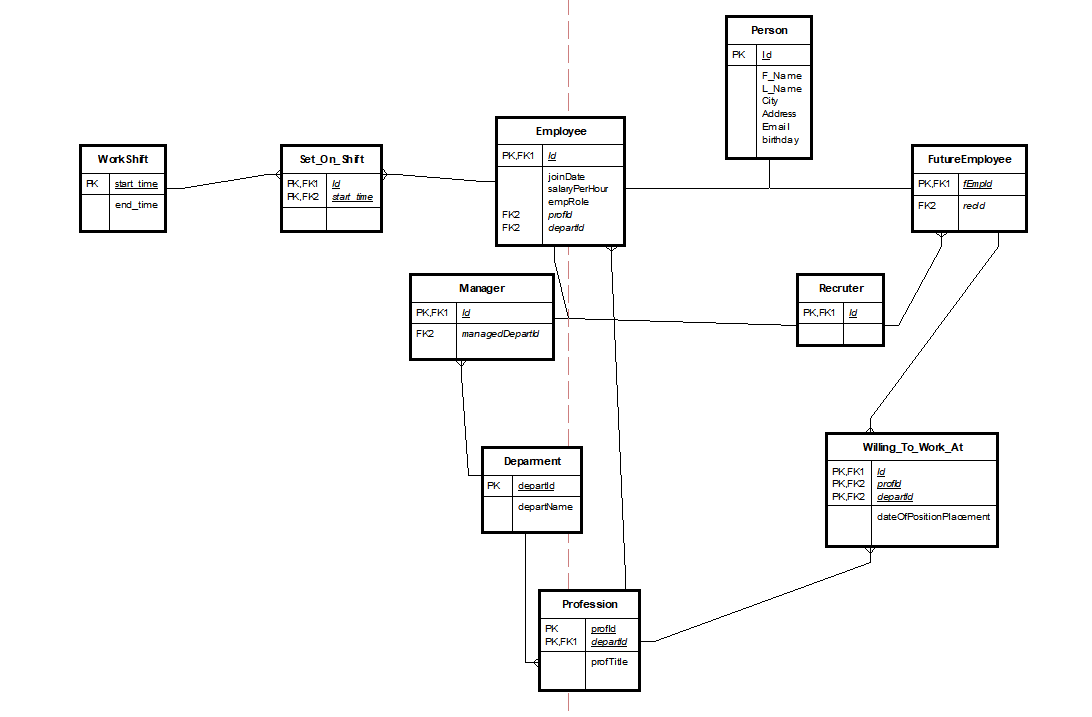
1. מספ' תפקיד – מפתח חלש לעומת ישות "מחלקה" (אפשר לראות בדיאגרמת ERD).
2. תיאור תפקיד..

**דיאגרמת – ERD**  
הדיאגרמה נעשתה בתוכנת lite DDS – כמו שהמרצה אמר לנו.  
זו בעצם דיאגרמה שמתארת בצורה ויזואלית את כל מה שהוסבר על הישויות בחלק הקודם.



**דיאגרמת DSD –**

תיאור הטבלאות שאנו ניצור במערכת בסיס הנתונים שלנו – ניתן לראות את המפתחות הראשיים בכל טבלה וכמו כן ניתן להבחין כי חלק מהקשרים מטבלת הERD אינם מתוארים כטבלאות בפני עצמם אלא מתוארים כמפתחות זרים בטבלאות אחרות.



שינוי נוסף שלא ניתן לעשותו בתוכנה היא קביעת מספר תכונות כמפתח ראשי לטבלה.

במקרה דנן רצינו ליצור את טבלת FutureEmployee כך שהמפתח הראשי לטבלה זו תהיה ת"ז העובד העתידי (fEmpID) וגם מספר המחלקה שבה הוא ירצה לעבוד departId)) – שכן עובד לא יגיש מועמדות לאותו התפקיד יותר מפעם אחת.  
אך ללא הצלחה נאלצנו להשאיר זאת לשלב כתיבת הסקריפטים.

**נרמול טבלאות – NF3**  
כל הטבלאות שלנו מנורמלות לNF3, נרמול טבלאות לצורה הזאת הוא מאוד חשוב מהסיבות הבאות:

1. מבטל יתירות על ידי פירוק טבלאות ליחידות אטומיות קטנות יותר.
2. גורם לשיפור שלימות הנתונים על ידי הפחתת הסיכון לחריגות.
3. מקל על מדרגיות וגמישות של מסד הנתונים על ידי מתן בסיס איתן לשינויים.
4. משפר את עקביות הנתונים על ידי אחסון נתונים באופן לא מיותר.

**לסיכום –**

נורמליזציה של טבלאות ל – NF3 משפרת את איכות הנתונים, מפחיתה יתירות וחריגות, מבטיחה שלמות ועקביות נתונים, מפשטת את התחזוקה ותומכת במדרגיות ובגמישות של מסד הנתונים.

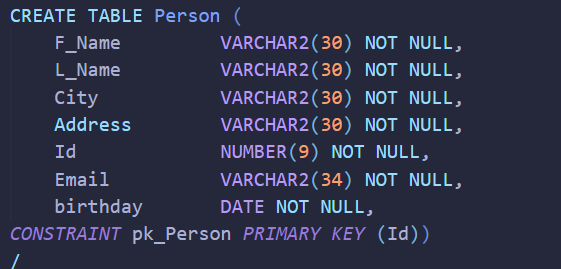
**יצירות טבלאות –**

בשלב זה נתאר את הטבלאות שיצרנו עבור בסיס הנתונים שלנו.  
עבור כל טבלה אנו נתאר ונביא את תיאור הטבלה כפקודת create table ב-SQL, את התהליך בה הכנסו את הנתונים ודוגמית קטנה של הנתונים.

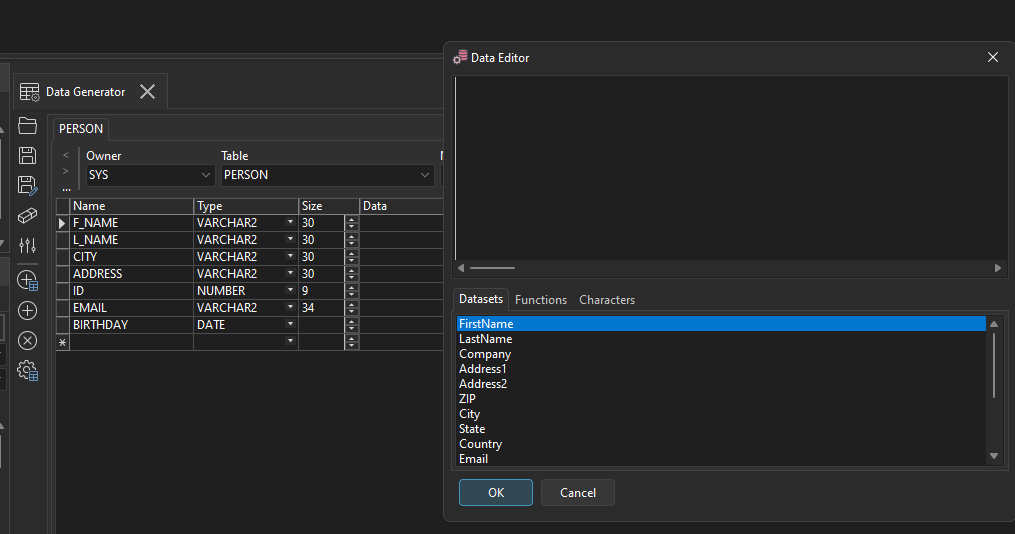
יאללה מתחילים –

**איש איש מה אתה מרגיש – טבלת האיש Person:**

טבלת האיש (Person בלע"ז) מכילה בתוכה אוסף של כל האנשים שקיימים בבסיס הנתונים שלנו.  
אוסף האנשים כולל בין היתר את העובדים, המועמדים וכאלה שהם סתם אנשים (סתם יחסית לתפקידים – אף אדם אין "סתמי") ולכן היא מהווה ישות אב עבור ישויות בן שיורשות ממנו את תכונותיה.  
את החלוקה וישויות הבן היורשות נתאר בהמשך הטבלאות, כעת נחזור לישות האב –

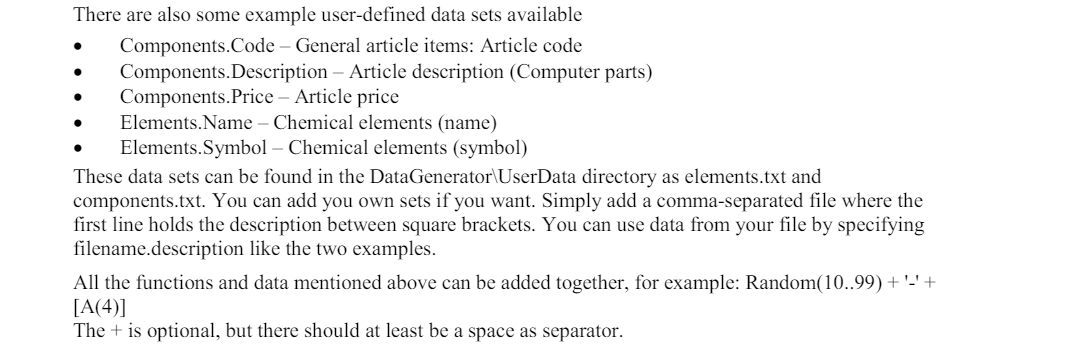
כפי שניתן לראות ישות ה-"איש" מכילה בתוכה תכונות בסיסיות של כל אדם, שמו ושם משפחתו, העיר והכתובת בה מתגורר, תאריך יום הולדתו וכתובת דואר האלקטורנית (כן, זה המצב כיום שהמייל מתאר את האדם נו שוין).

**אז נעבור לתכלס – איך הכנסנו את הנתונים?** ובכן, ל-PLSQL יש מחולל נתונים מוטמע בתוכנה (Data Generator) ובו קיימים מערכי נתונים מוגדרים מראש שנפוצים כגון: שם, שם משפחה, כתובת וכו –



דבר אחד שהיה חסר לנו היה מאגר מידע של תאריכים עבור תאריכי לידה של כל אדם מקבוצת הישויות "איש" – דבר זה לא היה קיים במערכת ולכן נאלצנו כמו כל מפתח טוב לרדת לעמקי השאול, אל הלא נודע, אל המעמקים – "אל התיעוד (Documentation)" (אל מלשון לכיוון ולא משלום ישות שמימית אלוהית כזאת או אחרת, למרות שכמה מפתחים מתייחסים אליו כאחד כזה).

אז הנה אנחנו בעיצומו של צאת יום הזיכרון יושבים ושוברים את הראש במאבק למצוא את הטריק, את הדרך בה נוכל לממש את זה.  
ואז ראינו את האור –

****כפי שניתן לראות בחלק השני של הקטע הזה (עמוד 196) – נאמר בו שאם נרצה להוסיף מאגרי מידע למחולל הנתונים אז עלינו ליצור קובץ עם הערכים כאשר הם מופרדים על ידי פסיק. ובנוסף, בראש הקובץ והעמודה עלינו לתת לו כותרת לערכים שבאותו העמודה מה שיכול לתת לנו את האופציה ליצור כמה עמודות באותו הקובץ – אבל אנחנו מעונינים רק באחד מהם אז הכל בסיידר.

אז עכשיו נשאל השאלה – הכיצד ניצור מאגר מידע שכזה?  
אני שמח ששאלתם כי לנו יש תשובה לזה והוא כולל את החית מחמד האהובה עלינו –

A snake eating a mouse

Description automatically generated

לא הפייתון הזה יא גאון הפייתון השני –

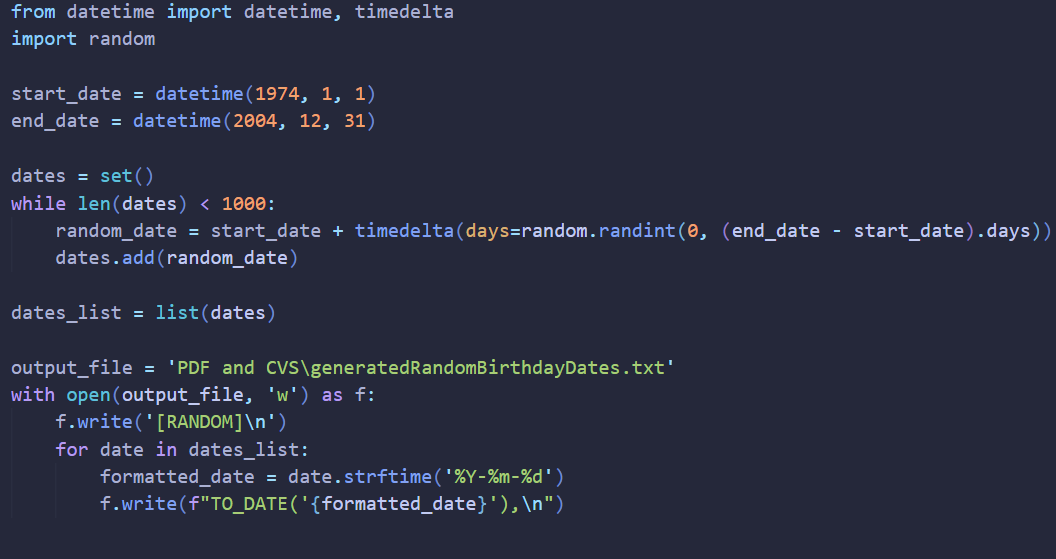


או שכוייח בדיוק מה שרצינו.  
אז ככה מה שאנחנו נעשה הוא ניצור סקריפט בפייתון שיג'נרט (מלשון Generate, המילה "יחולל" לא נשמעת לי כל כך טובה) לנו מלא תאריכי לידה בין השנים 1974 עד 2004  
הסקריפט מייצר קובץ csv שבו נמצאים ערכים רנדומלים של תאריכים כפי שציינו.  
בנוסף הוא ישמור את הערכים בתבנית –

TO\_DATE(‘date’)

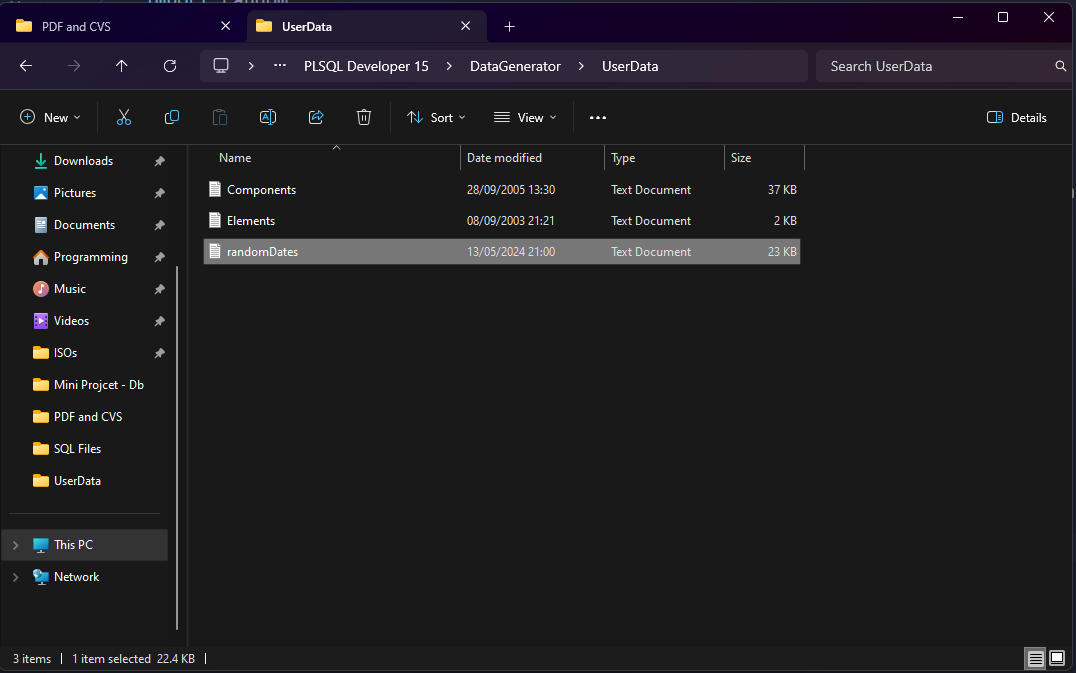
כך שכאשר המוחלל הנתונים יקרא את זה הוא יפרש את זה כערך מסוג DATE ולא מסוג NUMBER או VARCHAR2.

שנראה את הסקריפט? נו ברור אחרת לא תתנו לנו ציון לא ככה? –



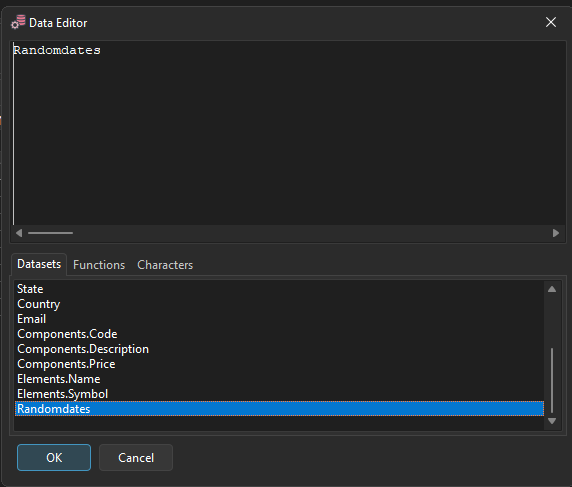
אךךךךך התענוג כשאתה לא צריך לעשות הרבה עבודה ... אמריקה

אז כפי שניתן לראות הוא מייצר את הקובץ עם הערך [RANDOM] בתחילת הקובץ המהווה שם שרירותי עבור העמודה של הערכים.

כעת כל מה שנותר לנו הוא ללכת לתקייה הראשית של המחשב – אל תיקייה Program Files לחפש את התיקייה של Plsql Developer ולהיכנס אל תיקיית UserData ולהכניס לשם את הקובץ המג'נורט –

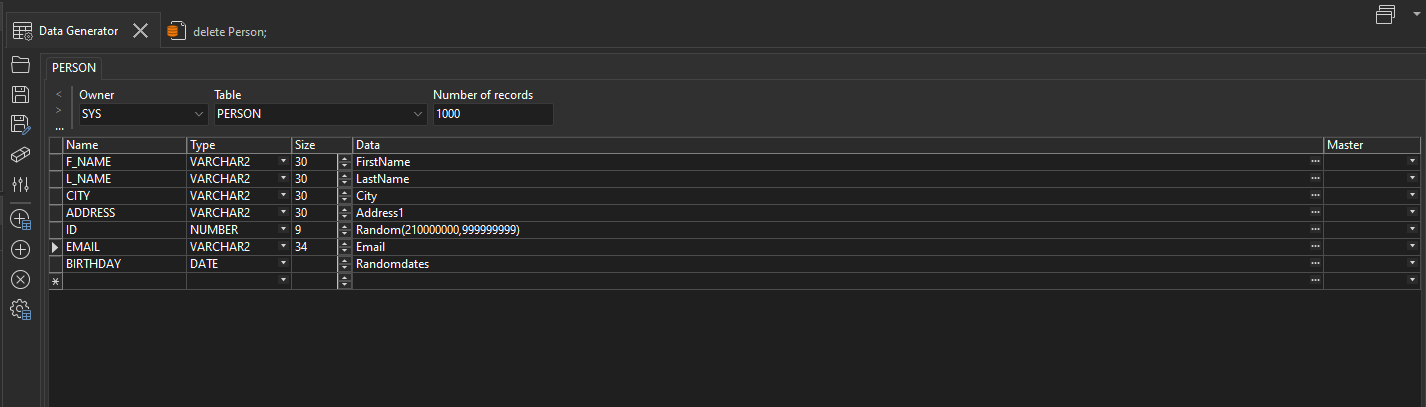
ניתן לשנות את השם על מנת לקבל תיאור יותר מינימלי במחולל הנתונים על הערכים שהקובץ הזה מכיל.

להפעיל מחדש את הPLSQL והפלא ופלא –

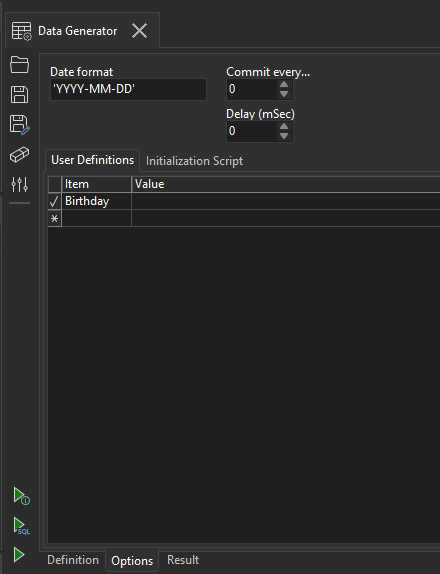
כעת ניתן לראות את הקובץ במחולל הנתונים וליצור שורות של אנשים עם שימוש בתאריכים האלו.

נכון שמגניב? חכו להמשך.

כעת סוף סוף אפשר ליצור את הערכים עבור הטבלה Person בקלי קלות –

בשורת הת"ז (id) נבקש שייצר לנו מספר אקראי בין 210000000 ל- 999999999

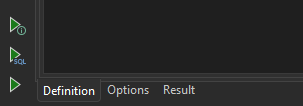
נבחר את המאגרי המידע הנחוצים לנו לכל אחת מהשורות ונבקש מהמחולל היקר שיחולל עבורינו סך של 1000 שורות



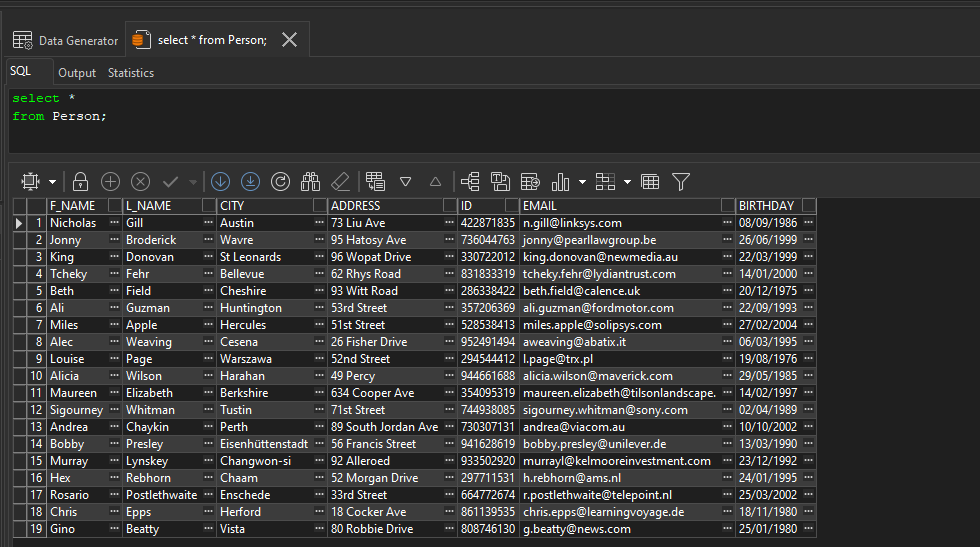
לאחר מכן נלחץ על הכפתור Options, נכתוב את הפורמט –

‘YYYY-MM-DD’ ונוסיף את השורה birthday כפי שמיוצג פה

על מנת להגיד למחולל הנתונים שהנתונים במיקום של birthday הם מהפורמט שרשמנו למעלה.

****

נלחץ על הכפתור הירוק העליון כדי לבדוק איך המידע יראה (לא להילחץ מזה שבעמודה של התאריך לידה המידע לא נראה כמו DATE זה ישתנה במהלך הג'ינרוט של המידע) ולאחר מכן על הכפתור האחרון כדי להריץ את הכל ו....

****

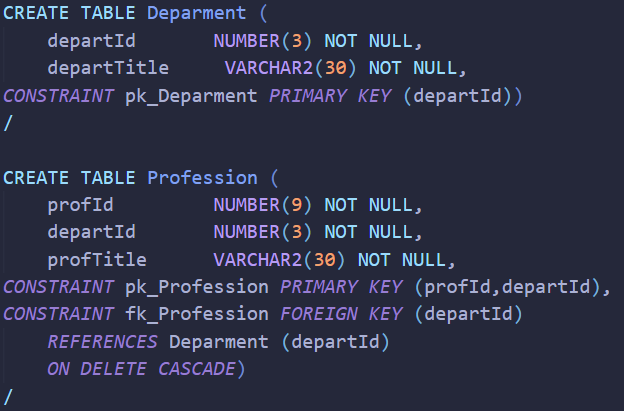
אאוריקה! יש לנו את זה חברים יקרים הצלחנו לג'נרט את המידע דרך מחולל הנתונים.

כעת נעבור לטבלה הבאה.

**מחלקות ולא בקטע של OOP – טבלת המחלקות והתפקידים Department, Profession:**

כמו שלכל סיר יש מכסה – לכל מנהל יש משועבד ולכל משועבד יש משהו שהוא משתעבד אליו.

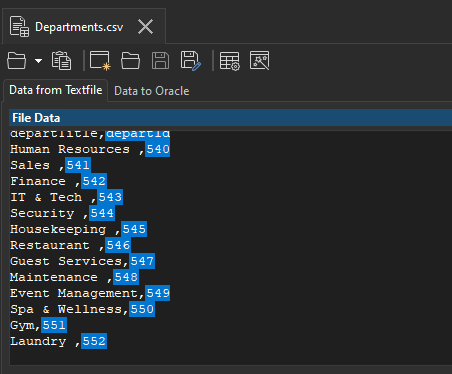
אנחנו ניצור כעת טבלה של מחלקות ותפקידים לכל מחלקה – את הקבצים האלו ימרנו בעצמינו אז בבקשה מחיאות כפיים כי יש פה ליטרלי 1500 פלוס שורות שנעשו באופן ידני!

אז ככה –

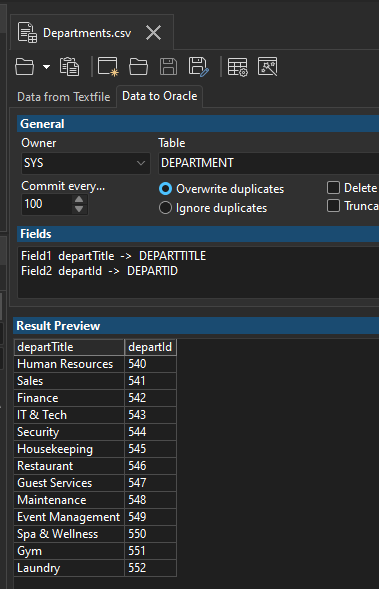
כפי שזכור לכם או שאתם רואים את זה כעת מול עיניכם – לכל מחלקה יש עובדים השייכים למחלקה הזאת ולכן הקבוצה של התפקידים היא ישות חלשה יחסית לקבוצה של המחלקות, זה מתבטא בכך שלקבוצה של התפקידים יש מפתח ראשי שמורכב מהמפתח שלו ושל הטבלה של המחלקות.

לאחר שיצרנו את הטבלאות הבא ונכניס לתוכה את הערכים שלה – בריפו תוכלו למצוא את הקבצי csv שמכילים את המחלקות ואת התפקידים (Departments.csv, professions.csv).

נשתמש בשיטה השניה של הכנסת נתונים והיא על ידי שימוש בקבצי csv ושימוש בכלי Text Importer שיש לנו בPlsql.



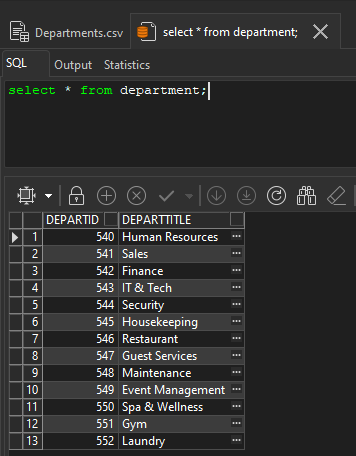
נבחר את הקובץ Departments.csv על ידי לחיצה על הכפתור של התיקייה.



לאחר מכן נבחר את הטבלה שנרצה להכניס אליה את הנתונים – במקרה דנן Department

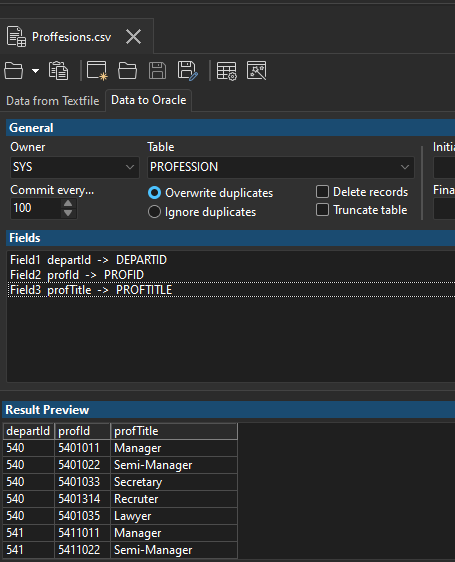
שימו לב שהתוכנה מזהה את ההקשר בין העמודות של הטבלה לעמודות של הקובץ היות ונתנו להם את אותם השמות.  
למטה ניתן לראות איך הטבלה תראה – אל דאגה יש לנו טבלאות יותר מאסיביות

נלחץ על הכפתור import ונייבא את המידע



והנה לנו המידע שלנו

וכעת ניתן להמשיך הלאה אל התפקידים של כל מחלקה  
הקובץ Professions.csv גם הוא ניתן לייבוא אל תוך בסיס הנתונים ונעשה את זה באותה הדרך



גם פה נבחר את הקובץ המקור ואת הטבלת יעד שלנו

ניתן לראות גם פה דוגמית מתוך כל הנתונים הקיימים

נלחץ על ייבוא ונקווה שהמחשב לא יקרוס (שוב) כתוצאה מכך

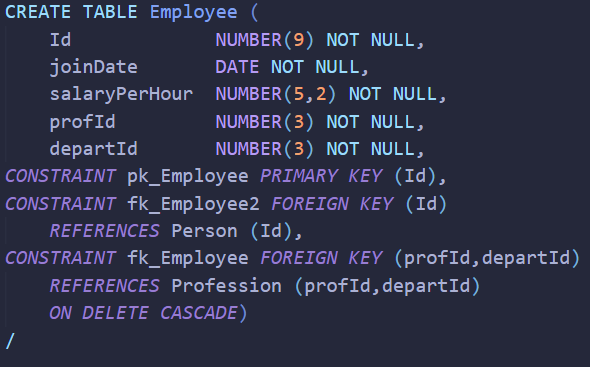
A black background with white text

Description automatically generated

וכפי שניתן לראות הכנסנו 1544 שורות של ערכים.

כעת נלך לשעבד לנו כמה אנשים.

**מוכר במדונלדס עוקץ המבורגר,ברמן בבר מבריח בקבוק – מה מבריחים המנקים? בקבוק אקונומיקה? – טבלת העובדים (Employee):**

כעת ניצור את טבלת העובדים –

כידוע לכם כל עובד מכיל בתוכו מפתח זר מישות האב ("איש") – המספר זהות שמתפקד כמפתח ראשי בסכמה זו

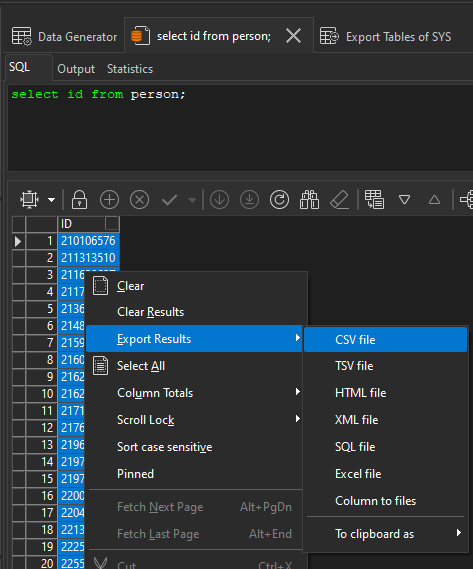
כמו כן ישות העובד מכילה בתוכו 2 מפתחות זרים נוספות – מספר מחלקה ומספר התפקיד שהוא עובד בה במלון.

לכן על מנת שניצור בסיס נתונים איכותי ואמיתי עלינו לוודא שהערכים שהעובדים מקבלים במפתחות זרים – הם בהתאם לערכים שבאמת קיימים בבסיס הנתונים שלנו ולא כל מיני ערכים אקראיים שמהווים אין ואפס ושום דבר. לכן עלינו לוודא שהם באמת מקבלים ערכים אמיתיים והגיוניים.

איך נעשה את זה? לשם כך נצטרך דבר ראשון לקבל מאגר מידע שמכיל את כל מספרי הזהות שכבר רשומים במערכת.

הנה איך נעשה את זה, נפתח את הPLSQL ונרשום את השאילתא שתתן לנו את כל האנשים שקיימים במערכת –

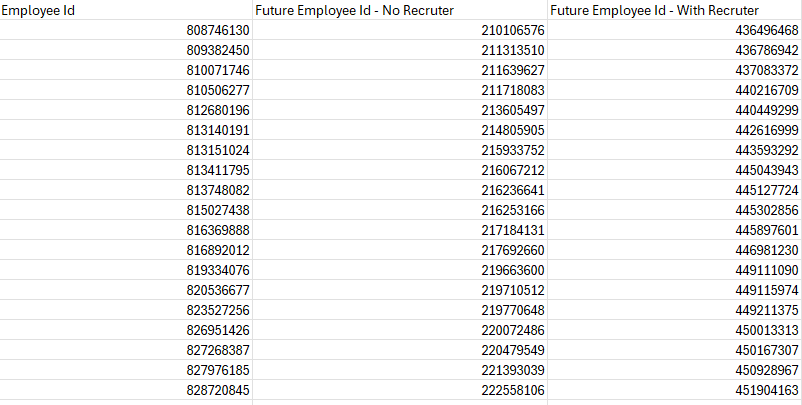
על מנת לקבל את כל התוצאות מבסיס הנתונים תלחצו על הכפתור הזה שמופיע בשורה מעל פאנל התוצאות.



ולאחר מכן נלחץ על העמודה ID וזה יסמן את כל שורות התוצאה של השאליתא.  
לחיצה ימנית על השורות המסומנות ולחיצה על Export Results ובחירה בקובץ csv תייצר לנו את הקובץ המבוקש, עם תוספת של השם שנבחר שהמערכת תבקש מאיתנו כמו כן גם המיקום בה נרצה לשמור את הקובץ נקבל את הקובץ המיוחל

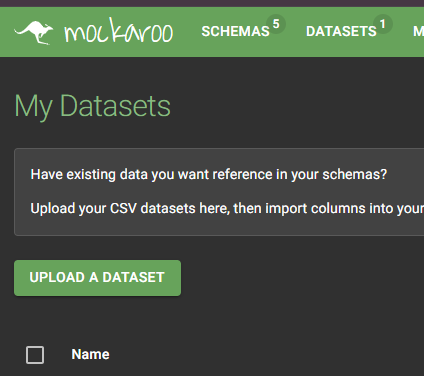
עכשיו כשיש לנו את הקובץ בואו נחלק את הערכים שלה לכמה חלקים, לחלק אחד שיציין את העובדים שכבר קיימים במערכת בתור עובדים, אנשים שעומדים להגיש מועמדות עם מגייס ואנשים שעומדים להגיש מועמדות ללא נציג.

הזה נוכל לייצר מידע שמתואם עם המציאות הקיימת.  
איך נעשה את זה? נשתמש באתר mockaroo הידוע –

נחלק את הערכים שקיימים לנו מבסיסי הנתונים בצורה הבאה –

* מספרי זהות הראשונים ילכו לאנשים שכבר עובדים במלון.
* הבאים הם יהיו האנשים שמועמדים לעבדות במלון והגיעו ללא עזרה של מגייס.
* ומי שנשאר יהיו המועמדים שהגיעו לעבודה עם עזרת המגייס

ניתן לזה שם חדש ונשמור את זה כקובץ csv וכעת נלך לאתר mockaroo –

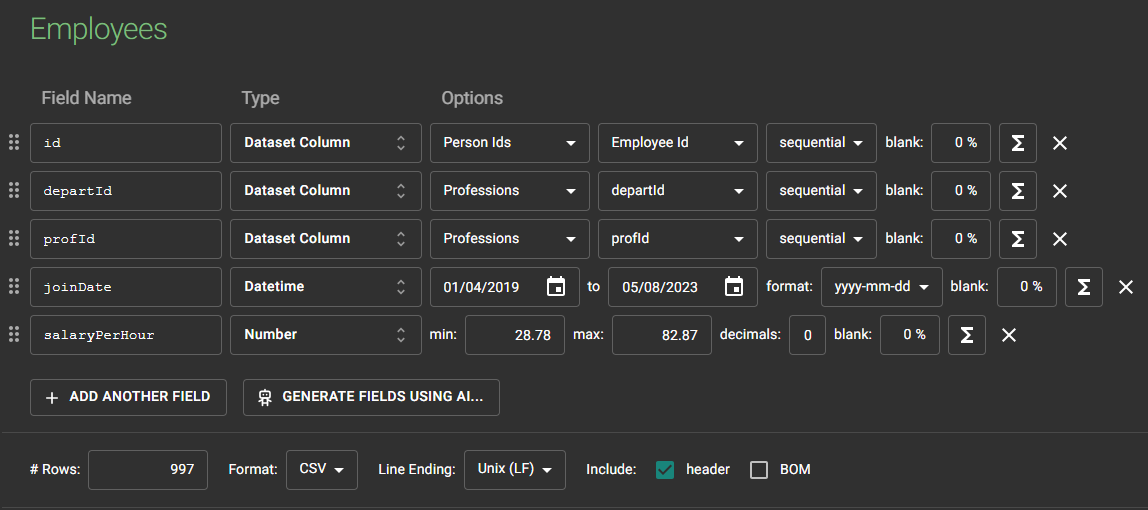
כפי שהיה לנו עם מחולל הנתונים גם פה נצטרך ליצור מאגר מידע על מנת שנשאב ממנה את הערכים עבור השדות הרצויות – במקרה דנן אנו רוצים שהשדה id בשדות העובדים תהיה מתואמת עם השדה id שבשדות האנשים.

נבחר datasets ונעלה מאגר מידע על ידי לחיצה על upload dataset

נעלה לשם את הקובץ וניתן למאגר הזה שם – Employee ids

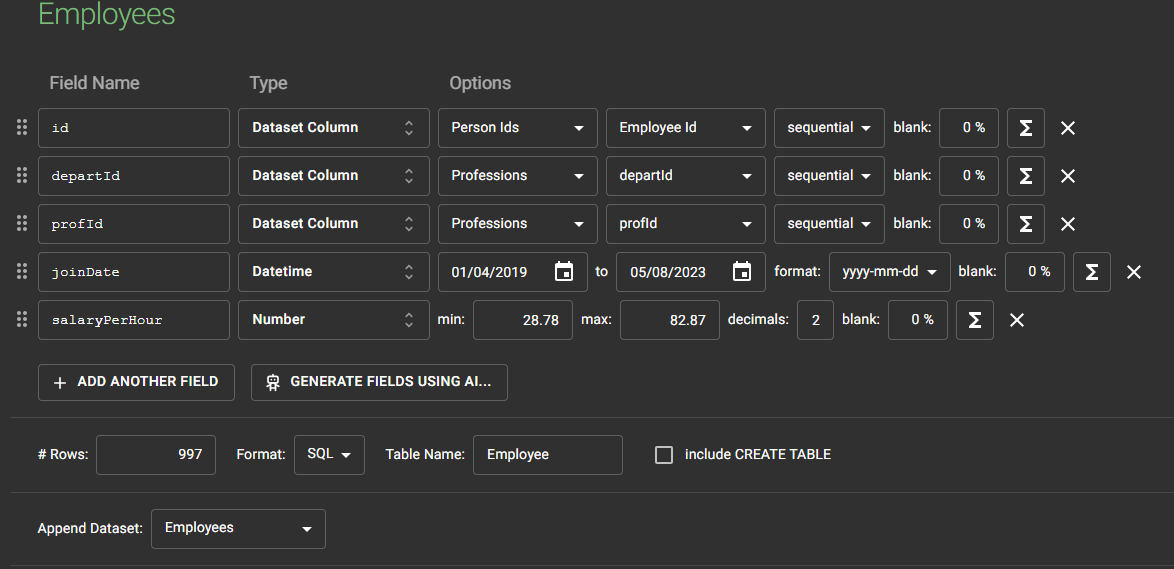
כעת נוודא שיש לנו מאגר מידע של התפקידים והמחלקות שלהם – נשתמש בקובץ Profession.csv שהשתמשנו בו מקודם.  
נעלה גם אותו לאתר mockaroo על מנת שתהיה לנו גישה אל המידע שבפנים.

וכעת אנחנו מוכנים ליצור את השורות עובדים שלנו –



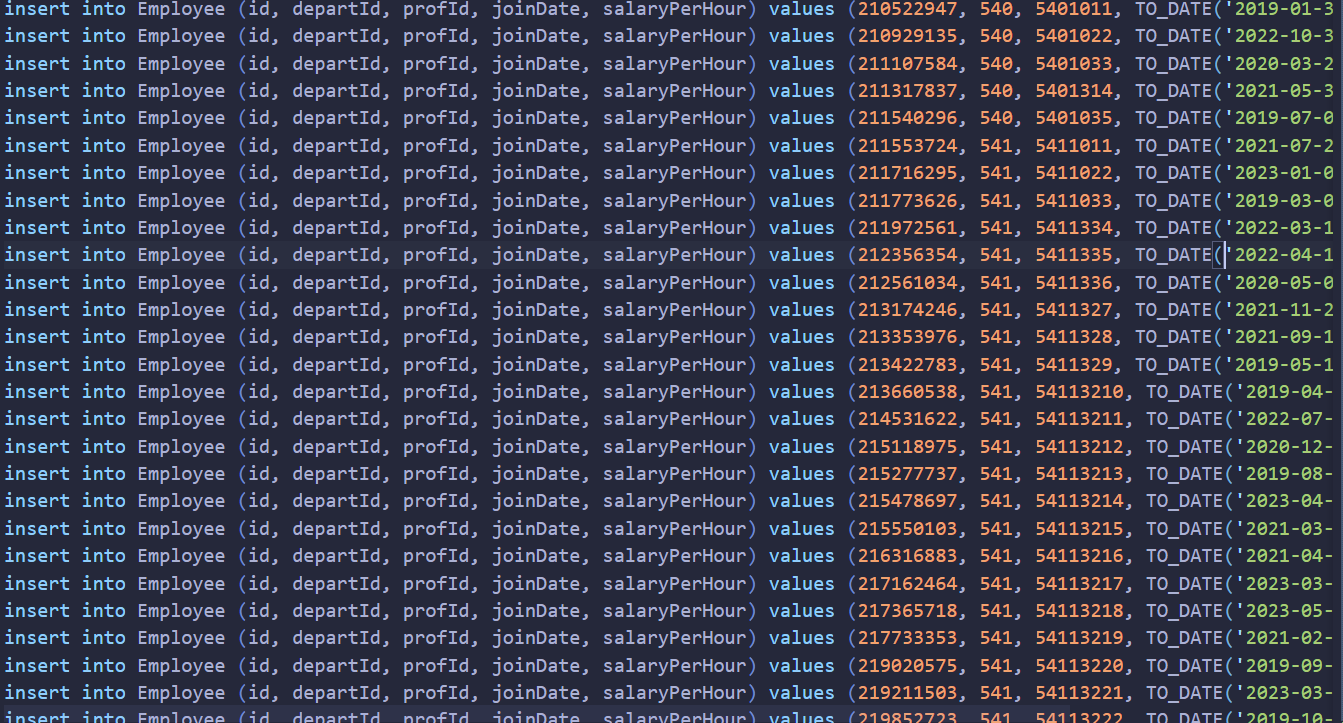
כפי שניתו לראות אנחנו משתמשים בערכים שנמצאים בקבצים שהעלנו בתור מארגי מידע (Person Ids Professions) ואנחנו משתמשים בעמודות מסוימות בכל אחד מהקבצים האלו (Employee Id וכו) ובהגדרות מסוימות למארגי מידע קיימים כמו טווח תאריכים עבור תאריך הכניסה לעבודה (joinDate) ומספר מינימלי ומסקימלי עבור המשכורת (salaryPerHour) כמובן שנבקש גם 2 ספרות לאחר המספר השלם.  
ובנוסף על ידי בחירה בsequential אנחנו מבקשים ממחולל הנתונים לייצר את הערכים כך שיעבור שורה אחר שורה ולא בצורה רנדומלית – היות ונרצה שעובד יהיה מתואר פעם אחת בלבד (ובנוסף ביקשנו רק 997 ערכים היות וקיימים לנו רק 997 עובדים במאגר המידע).

ולאחר מכן נבקש ממנו שיוריד לנו קובץ SQL שמכיל בתוכו את פקודות הinsert into על מנת שנכניס את המידע לבסיס הנתונים שלנו –



נבחר את המאגר מידע שיצרנו מקודם בחלקה של Append Dataset נרשום את השם של הטבלה שלנו ונלחץ על Generate Data

ניקח את הקובץ שירד למחשב שלנו ונזרוק אותו אל תוך Plsql –



על אף שזה מתואר כמו ש DATE מתואר – נצטרך לשנות את הערכים של התאריכים ככה שאוקרל ימיר את זה לערכים של DATE בדיוק ככה –



כעת כל מה שנותר לנו הוא להריץ את זה.

A screenshot of a computer

Description automatically generated

ובום! יש לנו את זה הצלחנו ליצור 997 עובדים כמנין תתקפזי שלא אומר כלום אבל לא אכפת לנו כי נתקענו על זה כל כך הרבה זמן.

יששששששששש.

כעת נעבור לשלבים הבאים שכוללים את המשמרות (לא משמרות המהפכה למרות שלפי המצב רוח של העובדים דווקא נראה לי שזה יקרה אוטוטו) ואת השילוב של העובדים עם המשמרות הקיימות.

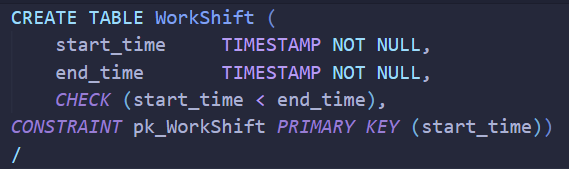
**"רציתי לבוא לעבודה" אמרתי לבוס "אבל הכלב שלי אכל את הרב קו" – טבלת המשמרות והרישום למשמרות (WorkShift, Set\_On\_Shift):**

החיים לא קלים לפעמים, פתאום הבוס חוטף ג'ננה ומבקש ממך לבוא לעבודה ב3 לפנות בוקר.  
למה מה קרה? מה אני עבד? אז מה אם נרשמתי למשמרת לילה צריך להתחשב באנשים שמנמנים במהלך העבודה.

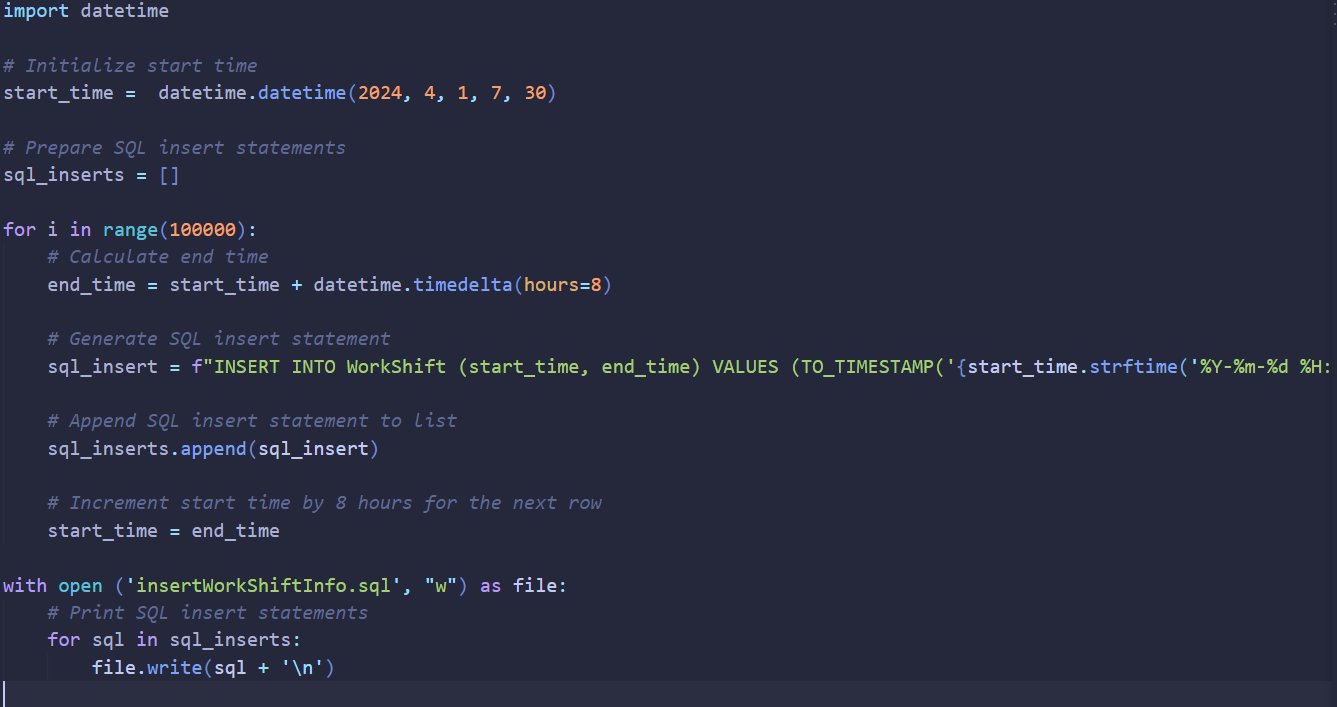
שגעונות כאלה אולי תשמעו במייקרוסופט אבל במלון "סה לה ווי" היוקרתי שלנו העובדים שלנו מרוצים מעבודתם – אין פלא כי הראשון שפתח את פיו על הדבר ישר הוצא להורג 😊

אבל כעת אנו נתרכז בלבנות לעובדים החרוצים שלנו משמרות שבהם יוכלו לכתת רגליים ולשבור את הגב.

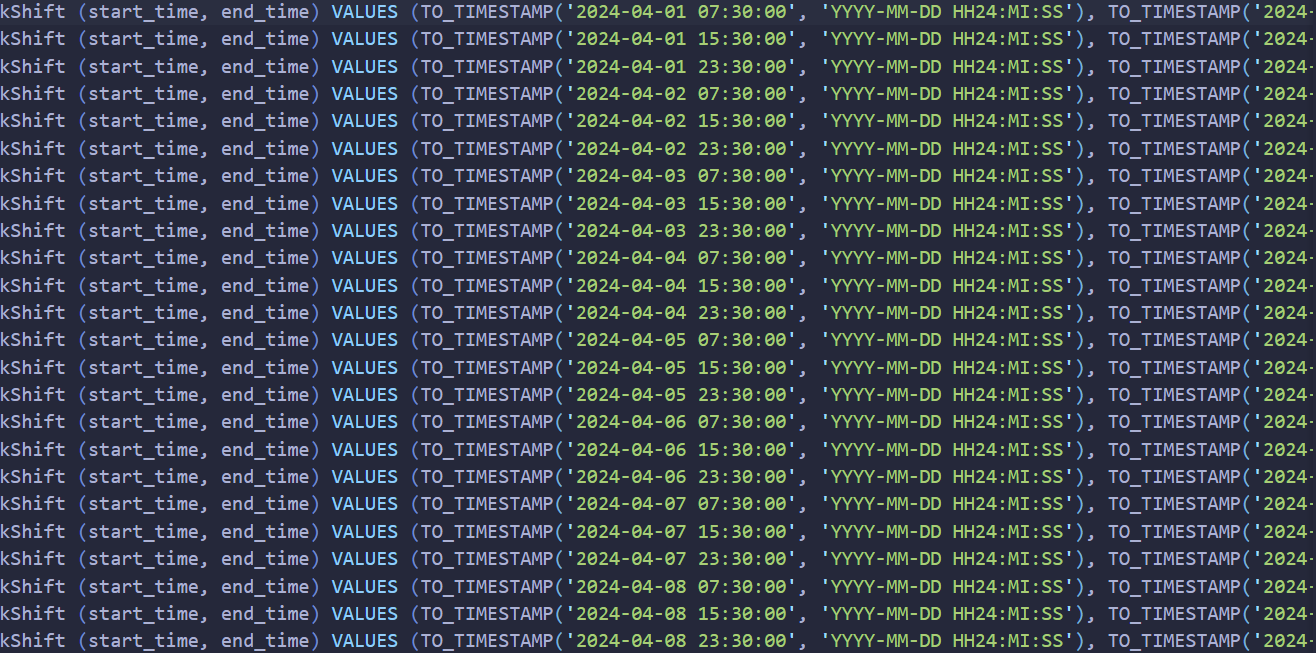
אז הבא נתחיל shall we? –

ניצור טבלה פשוטה המכילה את זמני תחילת המשמרת וסופה.  
שימו לב – לא ניתן לגמור משמרת לפני שנתחיל אותה 😌

על מנת ליצור את המידע שנכניס אל הטבלה בבסיס הנתונים שלנו אנחנו נעזרים שוב בחיית המחמד שלנו פייתי –



ביקשנו (יפה אחרת הוא נושך) שיצור לנו רשימה של משמרות – 5000 במספרם - עם זמן התחלה וזמן סוף בתקופת הזמן מהחודש הרביעי של 2024 ועד הרחק הרחק אל העתיד.  
כל זמן מורכב מתאריך מסוים ושעת תחילת המשמרת – 7:30 או 15:30.  
אורך כל משמרת הינו סך הכל 8 שעות בלבד.



והנה הערכים המג'ונרטים נריץ אותם בPLSQL כמו שעשינו עד כה ו ... -

A screenshot of a computer

Description automatically generated

הנה לנו 5000 משמרות עבור עובדינו יקרים.

כי אתם יודעים מה אומרים – "יותר עבודה פחות זמן לחשוב על העתיד או מקום עבודה יותר טובה".

נ.ב. אולי זה לא יעבוד לכם לכן כדאי לפצל את זה ל5 קבצים של 1000 נתונים כל אחת.

כעת נשדך את העובדים היקרים שלנו עם המשמרות היקרות שלנו.

על מנת לעשות את ניצור קובץ עם כל זמני תחילת המשמרות באמצעות הקוד הזה –

A computer screen shot of a program code

Description automatically generated

וכל מספרי הזהות של העובדים – מה שכבר קיים לנו - ונשמור אותם בתיקייה על מנת להשתמש במחולל הנתונים של PLSQL –

A screenshot of a computer

Description automatically generated

וכמו קודם נריץ את המחולל הזה –

A black background with white text

Description automatically generatedובום יש לנו את זה אנשים!

**מנהלים הגדרה: אנשים כמוך שמרוויחים יותר ממך מסיבה פוליטית כזאת או אחרת – טבלת המנהלים והמגייסים (Recruter, Manager):**

כעת ניצור קבוצה חדשה – היא לא בדיוק קבוצה יחודית אבל למחשבה לעתיד היא יכולה להועיל אם נרצה להפריד בין מנהלים לשאר האנשים.

נחפש את כל המנהלים שקיימים לנו בכל מחלקה ואת מספר המחלקה שלהם –

A screenshot of a computer

Description automatically generatedכעת נייצא את הערכים האלו אל קובץ csv חיצוני שנשתמש בו על מנת להכניס את הערכים אל טבלת המנהלים

נעביר את הקובץ הזה אל Text importer שבתוכנה ונכניס אל טבלת המנהלים –A screenshot of a computer

Description automatically generated

A computer screen with white text and green text

Description automatically generated

ניתן לראות כי שום מנהל לא התפספס במהלך הכנסת הנתונים אל טבלת המנהלים.

נעשה את אותו הדבר עבור המגייסים – נמצא את רשימת כל המגייסים ונשמור את מספר הזהות שלהם –

A screenshot of a computer

Description automatically generated

ניקח את התוצאות ונייצא לקובץ csv ונשתמש בText importer על מנת להכניס את הנתונים אליו

A screenshot of a computer program

Description automatically generatedוכעת לאחר כל זאת אנחנו רואים כי –

הוספנו את המגייסים היקרים שלנו אל טבלת המגייסים.  
כעת כל מה שנותר לנו לעשות הוא ליצור את טבלת המועמדים היקרים שלנו וליצור את הקשר אל העבודה בה הם מעונינים להשתעבד עומד להיות כיף.

**אם הייתי חוזר אחורה בזמן והייתי אומר למישהו ב1372 שאנשים רצים להשתעבד בשביל אחרים הם היו סוקלים אותי על עצם היותי חייזר – טבלת העובדים העתידיים וטבלת הקשר רוצה לעבוד ב... (FutureEmployee, Willing\_To\_Work\_At):**

כמיטב המסורת נתחיל דבר ראשון מתהליך יצירת הטבלאות –

A computer screen shot of text

Description automatically generatedטבלת העובדים העתידיים מכילה בתוכה את מספר הזהות של המועמד/.

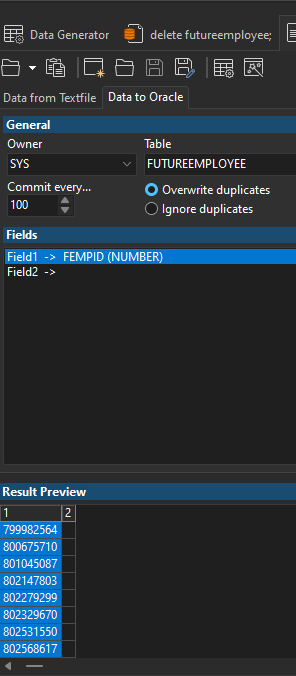
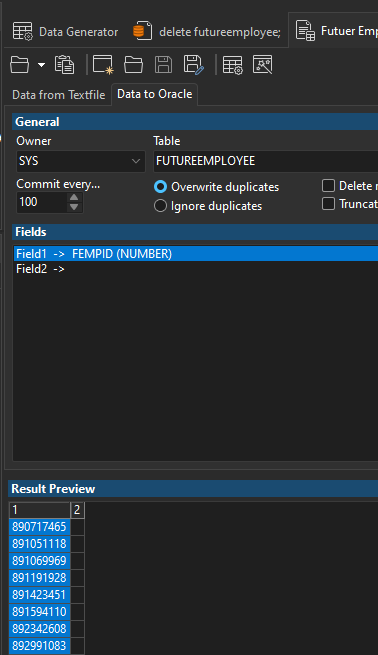
A computer screen shot of text

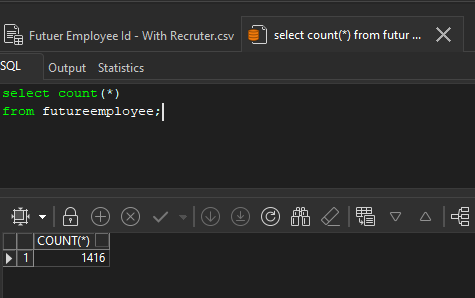
Description automatically generated

טבלת הקשר של העובדים המועמדים לטבלת העבודות.  
כפי שניתן לראות קשר זה הוא בין היחידי שניתן להם טבלה משלהם היות והוא קשר של רבים לרבים לכן לא ניתן פשוט לשמור את המפתח זר של הטבלה השניה באחת מהטבלאות כי יכול להיות יותר מעבודה אחרת לכל מועמד ויותר ממועמד אחד לכל עבודה.

כפי שניתן לראות היא מכילה בתוכה את המספר זהות של המועמד (id) את מספרי הזיהוי של העבודה שכולל את מספר התפקיד ומספר המחלקה שבה יעבוד, את תאריך הגשת המועמדות   
ובנוסף היות ואנחנו מדינה דמוקרטית לכן נתנו את האופציה שעובדים עתידיים יגישו מועמדות לעבודות לבדם ללא עזרת המגייסים – לכן הכרחנו את התכונה של המגייס להיות בעלת ערך NULL.

כעת לערכים – כפי שזכור לכם אנחנו הפרדנו בין העובדים למועמדים ובין המועמדים על ידי מגייסים וכאלה שלא.  
כעת זה יבוא לנו ממש בשימוש טוב – נכניס את כל המועמדים אל FutureEmployee אך בהכנסת הערכים לקשר נפריד ביניהם כך שיהיו לנו במערכת גם עובדים שלא הגישו מועמדות על ידי המגייסים.

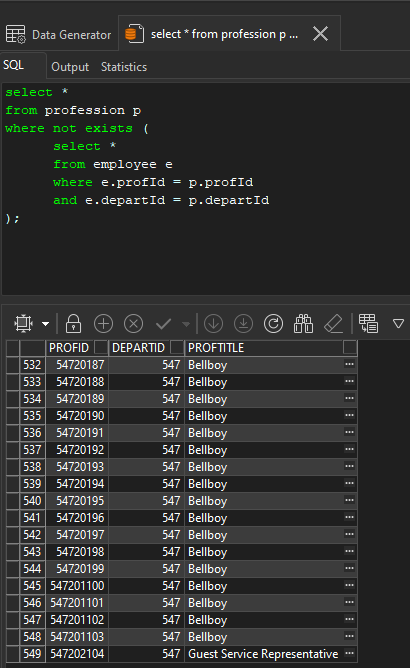
נתחיל עם הכנסת המועמדים –   
נשתמש בקבצי Future Employee שיש לנו בתיקיית הEmployee ונשתמש ב Text Importer



כעת נעבור לשלב יצירת הטבלה עבור הקשר Willing\_To\_Work\_At –

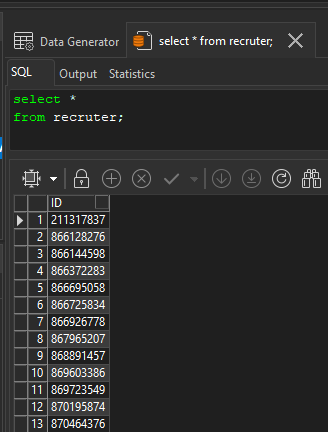
דבר ראשון נרצה לוודא איזה עבודות פנויות בחברה – הרי לא נרצה לקחת את העבודה למישהו אחר נכון? שואל רציני אי אפשר לדעת כבר בעולם הזה מה אתי ומה לא

נריץ את הפקודה הבאה כדי לקבל טבלת תוצאה עם העבודות הפנויות –



נו אתם יודעים כבר מה בא עכשיו נכון? ניקח את כל הערכים מטבלת התוצאה, נייצא לcsv ואז נשמור את המאגרים באתר mockaroo ושם נייצר את הערכים עבור הטבלה הזו.

אז קדימה בואו נעשה את זה כמו שאנחנו יודעים נלחץ על הכפתור בצד שמאל למעלה ונבחר את כל הטבלה  
ניצור קובץ csv ונשמור.

ניצור קובץ נוסף שמכיל בתוכו רשימת מספרי זהות של כל המגייסים –

נריץ את הפקודה הבאה שתראה לנו את רשימת כל המגייסים שנמצאים בקבוצת הישויות "מגייס".

ניקח את הערכים מטבלת התוצאה הזו ונייצא לקובץ csv ונשמור.

רשימת המועמדים - הן ברי מגייסים והן הבודדים ללא המגייסים - כבר נמצאים לנו בתיקייה יחד עם שאר קבצי הcsv לכן כל מה שנותר לנו הוא לייצר את המידע מהמאגרים הקיימים לנו.

נעלה לאתר mockaroo את כל המאגרים הנתונים לנו ונתחיל ליצור את המידע עבור המעומדים ברי המגייסים –

A screenshot of a computer

Description automatically generated

נבקש ליצור שורות מידע בהתאם למשרות הפתוחות בשימוש עם המגייסים החביבים שלנו   
בואו נראה איך זה יראה –

**A screen shot of a computer screen

Description automatically generated**

נבצע שינוי קטן – כמו תמיד נוסיף את הטקסט שאומר לאורקל להמיר לערך של DATE –

A screen shot of numbers

Description automatically generated

משהו כזה וכעת נריץ ונכניס את המידע אל תוך טבלת הקשר.

וכעת נעשה את כמעט אותו הדבר עם המועמדים שאינם ברי מגייסים –

A screenshot of a computer

Description automatically generated

השינוי היחיד הוא שלא הוספנו שדה של מספר זהות המגייס, כמו כן גם שינינו את המאגר מידע וכעת אנחנו משתמשים במאגר מידע של המועמדים ללא מגייסים.

נוריד את הסקריפט ונשנה אותו שוב עם תוספת קטנה של TO\_DATE על מנת להמיר לערך DATE –

A screenshot of a computer screen

Description automatically generated

נריץ את הקובץ וכעת יש לנו את העובדים המועמדים, כאלה שהגישו מועמדות וכאלה שלא, כאלה עם מגייסים וכאלה שלא – החיים הם לא שחור לבן, בכל שחור יש איזה בורדו קטן שמסתתר בו.

A screenshot of a computer

Description automatically generated

והנה הבורדו הקטן שלנו – חמוד ומספק סוף לסאגת יצירת הטבלאות ואכלוסיהן 🤩🤩.