



פרויקט בסיסי נתונים

מטרת הפרויקט היא התנסות מעשית בכלים וטכניקות להקמת בסיס נתונים בסביבה עסקית, במהלך הפרויקט הקמתי בסיס נתונים, הדרוש לתמיכה בתהליכים עסקיים של החברה. הפרויקט יכלול אב טיפוס של בסיס הנתונים והדגמת השימוש בו.

הפרויקט כולו התבצע בשרת MySQL ובכלים מתקדמים משפת ה־SQL. לצורך ניתוח המידע והדוחות העסקיים בוצע שימוש בכלי ה־power Bi

ניתוח האתר : hollywoodmemorabilia.com

תוכן עניינים:

1. רקע
 - 1.1. רקע כללי על האתר והארגון (3)
 - 1.2. תהליך עסקי (4)
 - 1.3. תרשים (5)
 - 1.4. מודל ERD - עיצוב קונספטואלי (6)
 - 1.4.1. הנחות מודל ה־ERD (6-7)
 - 1.5. מודל טבלאי מנורמל (8)
 - 1.6. פירוט שדות בסיס הנתונים והערכת גודלם (8-10)
 2. עיצוב פיזי - בסיס נתונים פעיל על שרת RDBMS
 - 2.1. אילוצי בדיקה - Check (11)
 - 2.2. טבלאות חיפוש - Lookup table (11)
 - 2.3. בניית טבלאות - Script (12-15)
 - 2.4. הזנת רשומות - דוגמאות (15-16)
 - 2.5. הסרת רשומות - דוגמאות (17)
 3. שאלות
 - 3.1. שתי שאלות SELECT ללא קיבץ
 - 3.1.1. עבור המחלקות בהן נעשו מעל 20 הזמנות, הצג את סכומן וכמותן בשלוש השנים האחרונות (18)
 - 3.1.2. אילו לקוחות רשומים מבצעים חיפוש ע"פ קטגוריית "אנשים" (18)
 - 3.2. שתי שאלות SELECT מקוננות
 - 3.2.1. באילו מחלקות חלה עליה בכמות החיפוש בשנה החולפת (19)
 - 3.2.2. הצגת הלקוחות שביצעו כמות הזמנות גבוהה מהמוצע (19)
 - 3.3. שתי שאלות SELECT שעושות שימוש בכלים מורכבים
 - 3.3.1. אילו מוצרים נמכרו ללא חתימת אותנטיות (20)
 - 3.3.2. שאלת המעדכנת שדה במחלקה בכל חיפוש נוסף (21)
 4. יישום כלים מתקדמים
 - 4.1. View מציג כמות מוצרים שנמכרה מכל מחלקה (22)
 - 4.2. פונקציות:



- 4.2.1. Functions פונקציה סקלרית הבודקת כמה מהלקוחות שביצעו הזמנה, ביצעו חיפוש מקדים(22)
- 4.2.2. Functions פונקציה שמקבלת שנה ומחזירה את טבלת הלקוחות שתוקף כרטיס האשראי שלהם יפוג במהלך השנה (23)
- 4.3. Stored Procedure פרוצדורה שמורה, המקבלת חודש ומסירה את כל המוצרים שלא חופשו מרשימת הנמכרים ביותר (24)
- 4.4. Trigger טריגר המפעיל טבלה נוספת ומופעל בכל הזמנה חדשה (25)
5. כלים להצגת נתונים – שימוש ב-power BI להצגת דוחות עסקיים
- 5.1. דוח עסקי ופונקציית VIEW (26)
- 5.2. לוח מחוונים ופונקציית VIEW (27-28)
6. יישומים כלים מתקדמים נוספים
- 6.1. שתי שאילתה עסקית המשלבת Windows Functions (29-31)
- 6.2. שילוב מערכתי של מספר כלים (32-33)
- 6.3. דוח המושתת על שימוש בפסקת WITH מורכב (34-35)



1. רקע

1.1. רקע כללי על הארגון והאתר

האתר "hollywoodmemorabilia" נוסד בשנת 1998, מטה האתר ממוקם במיאמי פלורידה. החברה נוסדה במטרה להיות יעד מקוון מספר אחת לחובבי הוליווד על ידי יצירת קהילת משתמשים וקטלוגים מקיפים של מוצריו. האתר מציע מוצרים אותנטיים החתומים על ידי סלבריטאים בהוליווד בתחומים הבאים: מוזיקה, סרטים, סדרות טלוויזיה, קומיקס, היסטוריה ותיאטרון ומאפשר למעריצים להתחבר, לשתף עצות וטיפים, ולעיין בקטלוגי מזכרות מקיפים. כמו כן, לחברה קיימות פלטפורמות נוספות בטוויטר, באינסטגרם ובפייסבוק.

מטרתה העסקית והעיקרית של החברה היא מקסום רווחים והיא עושה זאת על ידי מכירת מוצרים. על מנת להגדיל את מכירות החברה, האתר מאפשר חווית קנייה איכותית ופשוטה ככל שניתן על ידי: חווית שירות דיסקרטית ופרטית אשר מותאמת ללקוח במטרה ליצור חווית קנייה מהירה ונטולת דאגות, מוקד מכירות טלפוני הזמין 7 ימים בשבוע משמונה בבוקר ועד חצות, דף ייחודי לעזרה בתחומים שונים ומענה על שאלות נפוצות. בנוסף, בדף הבית של האתר ישנה חלוקה ברורה של התחומים השונים בהם הוא מתעסק ובכך מאפשר נגישות למוצריו.

האתר מציע מוצרים לקהל חובבי הוליווד, המוצרים בעלי ערך סנטימנטלי עבורם ואספקתם מתבצעת על ידי ספקים ספציפיים אשר מספקים לחברה תעודות אחריות ואותנטיות של המוצרים, לכן מחיר הקנייה והמכירה של מוצרים אלו הינו גבוה, משמע שקהל הלקוחות שיכול להרשות לעצמו לרכוש מוצרים אלו צריך להיות בעל אמצעים כלכליים מתאימים לכך ומכאן ניתן להסיק שהוא מצומצם.

היות ומדובר בחברה שמטרתה העסקית המרכזית היא רווח מקסימלי על ידי הגדלת המכירות וקהל היעד ניצב אל מול החברה אתגר, היכולת של החברה להגדיל את קהל היעד שלה ובכך להגדיל את מכירותיה מוגבל.

אתגר נוסף אתו החברה מתמודדת, לאור העובדה שהאתר מציע מוצרים מקוריים שנחתמו על ידי סלבריטאים מתחומים שונים, על האתר להוכיח את אמינותו ולספק ללקוחותיו מידע על אותנטיות המוצרים כדי שהם יתנו לו את אמונם ואכן ירכשו ממנו.

לצורך השוואה לקחנו את האתר הבינלאומי "Hollywood Collectibles" (<https://hollywoodcollectibles.com>) בדומה לאתר שלנו הוא מתמחה במוצרים מתחום התרבות הבינלאומית. ניתן למצוא חולצות חתומות, קלפי משחק, תמונות ממסגרות ועוד. בנוסף, ישנה הבטחת תעודת מקוריות ואף סיפוק שירותי מומחים להתייעצות והערכות מחיר. ההבדל העיקרי בו ניתן להבחין הינו אופן מכירת המוצרים, Hollywood Collectibles פועל באופן של הגשת הצעת מחיר ותאריך פקיעה עתידי לכל מוצר ולא מציג מוצרים מתומחרים מראש.

בנוסף האתר הישראלי שבחנו הוא "האספן" (<https://www.haasfan.co.il>) הפורטל הישראלי המוביל בתחום האספנות בארץ, אתר זה מציע פלטפורמה למסחר בין אספנים במגוון פריטים ובתחומים מגוונים החל ממטבעות עתיקים, בולים, סיכות ועוד. הדמיון המובהק בין בשניים הינו רכישת מוצרים בעלי ערך סנטימנטלי, פריטים ייחודיים ומקוריים. ההבדל העיקרי ביניהם הוא היקף



המוצרים הנמכרים, אתר "האספן" מתמקד בפרטים אותנטיים ממדינת ישראל, ולכן כמות התוכן הנמכר בו מצומצמת ביחס לאתר שבחרנו.

1.2. התהליך העסקי

התהליך העסקי הינו רכישת פרטי אספנות על אמנים וניהול תהליך המכירה. קיימים באתר תהליכים עסקיים נוספים כמו ניהול מערך ההזמנות עבור לקוחות, מתן שירות לקוחות וכד'.
הליך זה הינו הדומיננטי והחשוב ביותר לחברה, הוא מציג בתוכו את מקור הרווח העיקרי של החברה. פרט להליך זה, לחברה יש תהליכים עסקיים נוספים החיצוניים לאתר כמו ניהול מערך המשלוחים ממחסני החברה. מלבד זאת החברה אינה מספקת שירותים מחוץ לאתר הן בפלטפורמה הדיגיטלית והן במכירה קמעונית.

יתרונות הלקוח מרכישת מוצרי אספנות באינטרנט:

- א. יתרון עיקרי עבור הלקוח הינו בריכוז מגוון מוצרים במקום אחד, ניתן להשוות בין מוצרים דומים ולחפש אחר המוצר הרלוונטי ביותר.
- ב. בחיפוש אחר מוצרי אספנות תמיד קיימים מי שינסו "לנצל" את הסיטואציה ולבקש מחיר גבוה עבור מוצרים בעלי אלמנט סנטימנטלי לרוכש. ברכישה של מוצר באינטרנט קל לערוך השוואה בין אתרים שונים המציעים מוצרים דומים ולבדוק כי המחיר המוצע הולם את שווי המוצר.
- ג. זמינות, רכישה באינטרנט נעשית מכל מקום ובכל זמן, קל לבצע אותה ואין צורך להגיע פיזית למקום המכירה. רכישת פרטי אספנות מטבעה תתבצע במקומות שונים בעולם. רכישה באינטרנט מספקת פלטפורמה אחת המאגדת בתוכה מגוון רחב של מוצרים, ומאפשרת פתרון יעיל לבעיה זו. בכך ניתן לספק משלוחים לכל רוכש בכל מקום ובכל זמן.

חסרונות הלקוחות מרכישת מוצרי אספנות באינטרנט:

- א. חווית הקניה במוצרי עתיקות ואספנות נפגעת משמעותית כאשר אי אפשר לגעת במוצר, להרגיש אותו לעיתים אף להריח.
- ב. למוצרים בקטגוריה זו ישנה נטייה להיות שבירים ולכן הם זקוקים להגנה מרבית. בעת משלוח הרוכש תלוי רבות בטיב השילוח של עובדי החברה וקיימת הסכנה כי המוצר ייפגם בדרכו.

יתרונות החברה מרכישת מוצרי אספנות באינטרנט:

- א. איסוף מידע נוח על הלקוחות, בכך יכולה החברה לעקוב אחר לקוחותיה הקיימים והפוטנציאלים.
- ב. הגדלת קהל לקוחותיה, תתאפשר פנייה בשפות שונות ובמדינות שונות לקהל רחב בכל מקום בעולם בצורה פשוטה יותר בעזרת האינטרנט.
- ג. החוסר בהפעלת חנויות קמעוניות מאפשר חיסכון כלכלי רב בעובדים ובשכירות.

חסרונות החברה ממכירת מוצרי אספנות באינטרנט:

- א. טבעם של רוכשי ומוכרי עתיקות הינם בשווקים ועל כן מאבדת החברה לקוחות מזדמנים פוטנציאלים שמחפשים את מבוקשם כחלק מתהליך קניה אמוציונלי ו-"חיפוש אחר אוצרות".
- ב. קשרי לקוחות, קשר אישי הוא מרכיב מרכזי בשימור לקוחות ובהיעדרו לא ניתן להבטיח את חזרת הלקוחות לרכישה חוזרת בחברה.

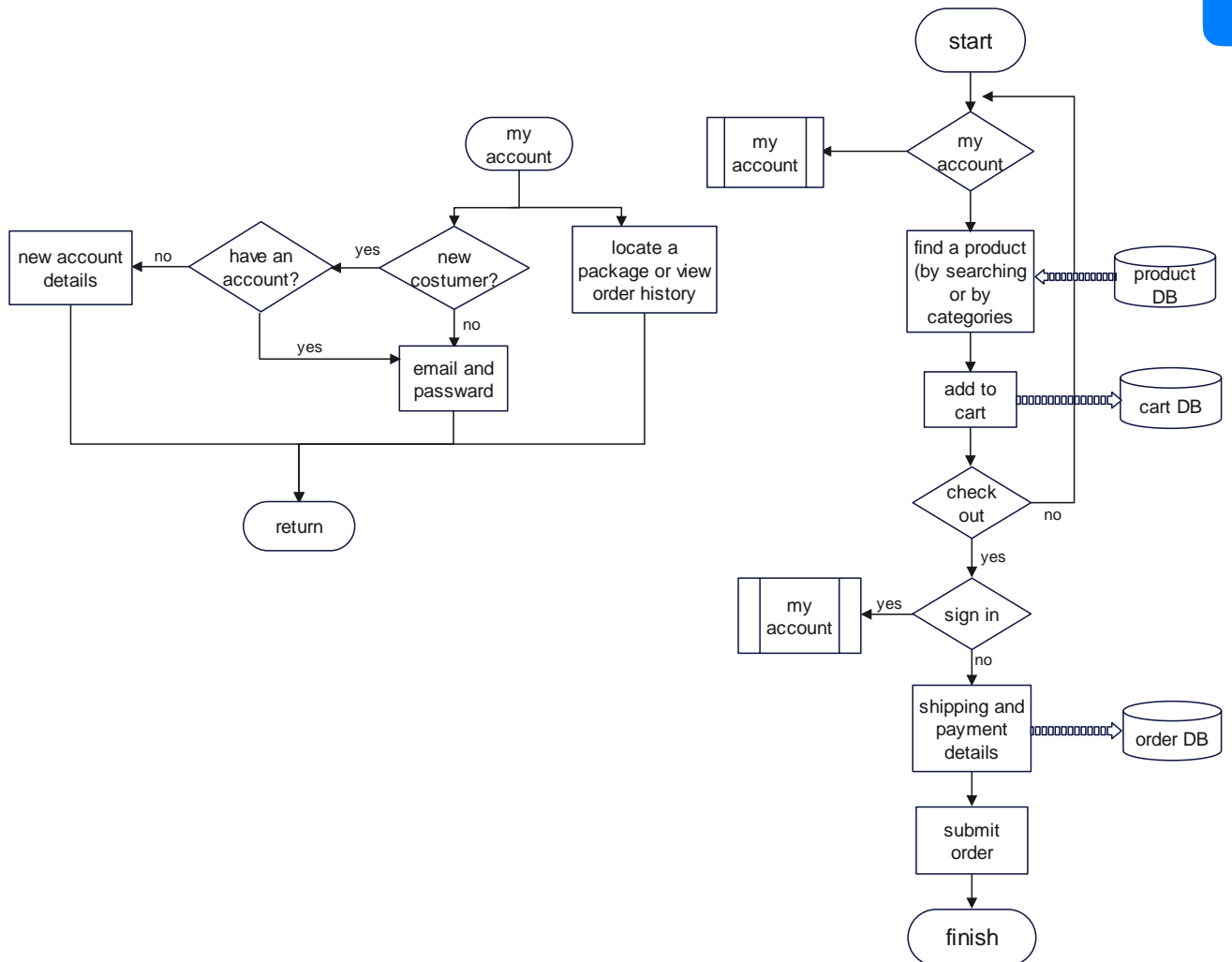
המידע הנאסף כחובה במהלך התהליך העסקי: שם מלא, כתובת אימייל, מספר טלפון, מדינה, עיר, כתובת מלאה מיקוד מלא ופרטי כרטיס אשראי/ אמצעי תשלום אחר.

מידע הנאסף כרשות במהלך התהליך העסקי: האם הרכישה מיועדת כמתנה, האם לשלוח לכתובת המצוינת על מבצע ההזמנה או כתובת אחרת.

החברה מאפשרת הוספת כרטיס ברכה במידה ומדובר במתנה.

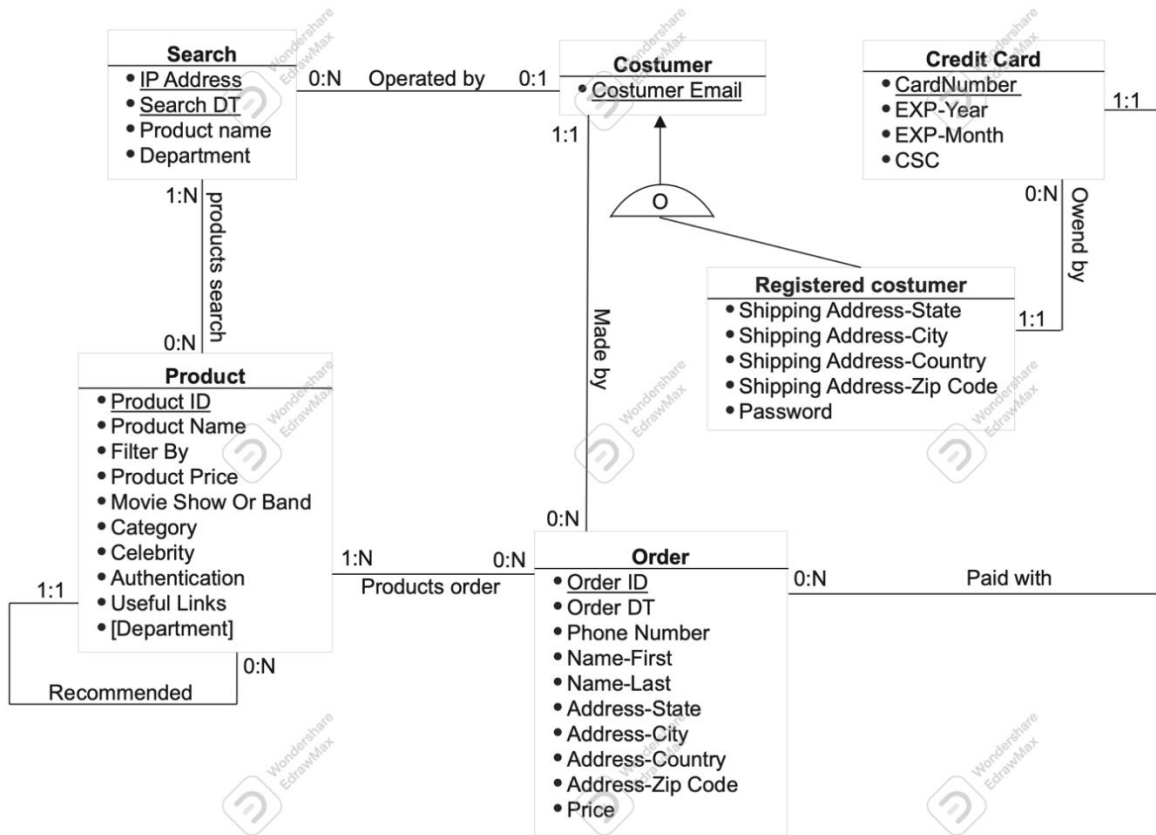


1.3. התרשים





1.4. מודל ERD - עיצוב קונספטואלי



1.4.1. הנחות מודל ה- ERD

| מרכיב | הנחות |
|--|---|
| השדות "department" "product name" "Search" בישויות | באתר קיימות שש מחלקות על פיהן מחולקים המוצרים והן: People, Movies, TV shows, Theater, Comics, History כל מוצר משויך ללפחות מחלקה אחת. אנחנו מניחים כי ניתן לבצע חיפוש על ידי הזנת שם המוצר או על ידי הזנה של ערך המשתייך לאחת המחלקות. |
| קרדינליות 0:N בין הישות "search" לישות "product" | בעת ביצוע החיפוש יכול להתקבל מצב שלא ימצא ערך לחיפוש שהוון ולכן הגבול התחתון הינו אפס, מצד הגבול העליון יכולים להתקבל עד N ערכים אפשריים לחיפוש שבוצע. |
| השדה "department" "Product" בישות | באתר קיימות שש מחלקות על פיהן מחולקים המוצרים והן: People, Movies, TV shows, Theater, Comics, History כל מוצר משויך ללפחות מחלקה אחת. בכדי לסווג את המוצרים למחלקות הוספנו לישות מוצר את השדה מרובה הערכים מחלקה. |



| | |
|---|---|
| קרדינליות 0:1 בין ישות "Search" לישות "costumer" | אנחנו מניחים כי חיפוש באתר יכול להתבצע על ידי לקוח של האתר (רשום או שביצע רכישה) או על ידי אדם שהוא לא לקוח של האתר כלל ולכן הקרדינליות היא 0:1. |
| קרדינליות 0:N בין ישות "Search" לישות "costumer" | אנחנו מניחים כי חיפוש באתר הוא כתיבת ערך בשורת החיפוש ולא הקשה על אחת המחלקות בדף הבית של האתר. לקוח של האתר (רשום או שביצע רכישה בעבר) יכול לבצע N חיפושים או לא לבצע חיפוש כלל ולכן הקרדינליות היא 0:N. |
| הישות "credit card" | אנחנו מניחים במסגרת הקורס כי לקוח משלם ישירות דרך האתר ולא דרך אתרי תשלום מאובטחים (PayPal, amazon pay). |
| הזמנה באתר על ידי costumer כאשר קיימים buyer costumer ו- registered costumer (מודל הורשה) | הזמנה באתר יכולה להתבצע על ידי לקוח שרשום לאתר ועל ידי לקוח שלא רשום לאתר. הגדרנו שני סוגי לקוחות, לקוח המבצע רכישה ללא רישום לאתר (buyer costumer) ולקוח שמבצע רכישה לאחר רישום לאתר (registered costumer). המזהה של לקוח הוא האימייל שלו, שנדרש להזנה על ידי שני סוגי הלקוחות בעת ביצוע ההזמנה והייחודיות של לקוח רשום היא שיש לו סיסמה ואת האפשרות להזין כתובת למשלוח. |
| שדה כתובת | השדה כתובת קיים גם בישות של לקוח רשום וגם בישות הזמנה מתוך ההנחה כי לקוח רשום יכול להחליט לשלוח את ההזמנה לכתובת שונה מהכתובת המוגדרת לו. בנוסף, לקוח רשום לא חייב להזין כתובת למשלוח, זהו שדה אופציונלי. |
| שדה "price" בישות "order" | לכל הזמנה יש מחיר הזמנה הכולל גם את מחיר המשלוח (במידה וקיים). |
| הקשר בין "registered costumer" ו- "credit card" | ללקוח אשר רשום לאתר ניתנת האפשרות לא להזין או להזין שיטת תשלום (כרטיס אשראי) או כמה שיטות תשלום ולכן הקרדינליות היא 0:N. |
| הקשר בין הישות "order" לבין הישות "credit card" | לקוח אשר לא רשום לאתר מבצע תשלום באמצעות כרטיס אשראי בזמן ביצוע ההזמנה. בדומה לכתובת המשלוח, לקוח אשר רשום לאתר יכול לשלם על הזמנה באמצעות כרטיס אשראי שונה מכרטיס האשראי שמוגדר לו כשיטת תשלום. לכן ישנו הקשר paid with בין הישות "order" לבין הישות "credit card". בנוסף, אנחנו מניחים כי כל הזמנה יכולה להתבצע על ידי כרטיס אשראי אחד בלבד. |
| הקשר הרקורסיבי של "product" | דרך כל מוצר ניתן להגיע לארבעה מוצרים מומלצים אחרים. מהרצה של כמה חיפושים באתר אנו מניחים כי בכל מוצר ממליץ על ארבעה מוצרים נוספים שממנו ניתן לגשת אליהם. |



1.5. מודל טבלאי מנורמל :

- **Orders** (order ID, order DT, email address, phone number, price, first name, last name, customer email (customers), customer email Buyer (Buyer customer), customer email Registered (Registered customer), Card number (credit card))
- **Customers** (Customer email)
- **Registered customers** (Customer email (Customers), password)
- **Shipping addresses** (customer email (customers), state, city, country, zip code)
- **Addresses** (order ID (orders), state, city, country, zip code)
- **Products** (product ID, name, filter, price, movie show or brand, category, celebrity, authentication, useful links, product(products))
- **Searches** (IP address, search DT, Customer email(Customers))
- **Credit cards** (card number, CSC, Customer Email (Registered Customer))
- **Expiration date** (card number (credit cards), year, month)
- **Products search** ({IP address, search DT}(Search), Products ID (products (
- **Products order** (Product ID (Products), Order ID (orders(
- **Departments** (Product ID (Products), department)

הסברים נוספים:

- הפרות נרמול מסדר ראשון – פירוק שדות מרובי ערכים לטבלאות נפרדות במעבר מ ERD לטבלאות: PRODUCTS - DEPARTMENTS
- הפרות נרמול מסדר ראשון – פירוק שדות מורכבים לטבלאות נפרדות במעבר מ ERD לטבלאות: REGISTERED CUSTOMERS – SHIPPING ADDRESSES, ORDERS - ADDRESSES, ORDERS – NAMES, CREDIT CARDS – EXPERATION DATE
- הוספת טבלאות לקשרים: PRODUCT SEARCH, PRODUCT ORDER בגלל שהם מציינים קשר של רבים לרבים.

1.6. פירוט שדות בסיס הנתונים והערכת גודלם

| Table | Field | Data Type | Size (Bytes) |
|---------------------|---------------------------|-------------|--------------|
| Customers | <u>Customer_email</u> | Varchar(40) | 40 |
| | Total Record Size (Bytes) | | 40 |
| Buyer customer | Customer_email (Customer) | Varchar(40) | 40 |
| | Total Record Size (Bytes) | | 40 |
| Registered customer | Customer_email (Customer) | Varchar(40) | 40 |
| | Password | Varchar(20) | 20 |
| | Total Record Size (Bytes) | | 60 |
| Orders | <u>Order_ID</u> | Integer | 4 |
| | Order_DT | DateTime | 8 |



| | | | |
|--------------------|---|--------------|-----|
| | Phone_number | Varchar(18) | 18 |
| | Price | Money | 8 |
| | Buyer_email_address (buyer costumer) | Varchar (40) | 40 |
| | Registered_email_address (registered costumer) | Varchar (40) | 40 |
| | Card_number (credit cards) | Varchar(20) | 20 |
| | Total Record Size (Bytes) | | 138 |
| Searches | IP_address | Varchar(25) | 25 |
| | Search_DT | DateTime | 8 |
| | Product_name | Varchar(40) | 40 |
| | Department | Varchar(10) | 10 |
| | Total Record Size (Bytes) | | 83 |
| Products | Product ID | Integer | 4 |
| | Product_name | Varchar(40) | 40 |
| | Filter_by | Varchar(40) | 40 |
| | Product_price | Money | 8 |
| | Movie_show_or_brand | Varchar(50) | 50 |
| | Category | Varchar(40) | 40 |
| | Celebrity | Varchar(40) | 40 |
| | Authentication | Varchar(20) | 20 |
| | Useful_links | Varchar(15) | 15 |
| | Department | Varchar(10) | 10 |
| | Total Record Size (Bytes) | | 267 |
| Credit cards | Card_number | Varchar(20) | 20 |
| | CSC | Integer | 4 |
| | Costumer_email (registered costumer) | Varchar(40) | 40 |
| | Total Record Size (Bytes) | | 64 |
| Expiration Date | Card_number (credit cards) | Varchar(20) | 20 |
| | Year | Date | 4 |
| | Month | Date | 4 |
| | Total Record Size (Bytes) | | 28 |
| Names | Order ID (orders) | Integer | 4 |
| | First_name | Varchar(20) | 20 |
| | Last_name | Varchar(20) | 20 |
| | Total Record Size (Bytes) | | 44 |
| Addresses | Order ID (orders) | Integer | 4 |
| | State | Varchar(20) | 20 |
| | City | Varchar(20) | 20 |
| | Country | Varchar(20) | 20 |
| | Zip_code | Integer | 4 |
| | Total Record Size (Bytes) | | 68 |
| Shipping addresses | Costumer_email (costumers) | Varchar(40) | 40 |
| | State | Varchar(20) | 20 |



| | | | |
|-----------------|---------------------------|-------------|-----|
| | City | Varchar(20) | 20 |
| | Country | Varchar(20) | 20 |
| | Zip_code | Integer | 4 |
| | Total Record Size (Bytes) | | 104 |
| Products search | IP_adresse | Varchar(25) | 25 |
| | Product_ID | Integer | 4 |
| | Search_DT | DateTime | 8 |
| | Total Record Size (Bytes) | | 37 |
| Products order | Product_ID | Integer | 4 |
| | Order_ID | Integer | 4 |
| | Total Record Size (Bytes) | | 8 |
| Departments | Product_ID | Integer | 4 |
| | department | Varchar(10) | 10 |
| | Total Record Size (Bytes) | | 14 |

הסברים נוספים:

- שדות התאריכים בטבלה SEARCHES נבחרו כתאריך ולא כתאריך/זמן, כיוון שהאתר אינו מאפשר חיפוש על פי שעות ספציפיות.
- שדה מספר הטלפון בטבלה ORDERS מכיל עד 18 ספרות מכיוון שבחרנו לקחת בחשבון מספרי טלפון ממקומות שונים בעולם, וקידומות של מדינות כמו +972
- בטבלת PRODUCTS על אף שברד ER סימנו את השדה product כמרוכה ערכים, לשם פישוט לא עשינו את זה במודל הטבלאי, אך מדובר בקישור לארבעה מוצרים נוספים המוצעים בהתאם לכל מוצר



2. עיצוב פיסי (בסיס נתונים פעיל על שרת RDBMS)

2.1. אילוצי מרחב ערכים אילוצי בדיקה (CHECK):

- א. אילוץ אי שליליות על הזמנה:
`ALTER TABLE orders ADD CONSTRAINT ck_order_price CHECK (price>0)`
- ב. אילוץ אי שליליות על מחיר:
`ALTER TABLE products ADD CONSTRAINT ck_price CHECK ([product price]>0)`
- ג. אילוץ כתיבת אימייל תקין:
`ALTER TABLE Costumers ADD CONSTRAINT ck_Email CHECK ([Costumer email] LIKE '%@%.%')`

2.2. טבלאות חיפוש (Lookup Tables):

א. טבלת חיפוש ל-PRODUCTS: חיפוש עבור סוג הסיווג של המוצר.

```
)CREATE TABLE Filters
[filter By] Varchar(40) PRIMARY KEY
(
INSERT INTO Filters VALUES
('autographed'), ('unsigned')
ALTER TABLE products
ADD CONSTRAINT FK_filters foreign key ([filter by]) references Filters ([filter by])
```

ב. טבלת חיפוש ל-PRODUCTS: חיפוש עבור קישורים שעשויים להיות רלוונטים.

```
)CREATE TABLE Links
[useful links] Varchar(15) PRIMARY KEY
(
INSERT INTO Links VALUES
('Best-Seller'), ('Free-Shipping'), ('New-Arrivals'), ('SALE')
ALTER TABLE products
ADD CONSTRAINT FK_links foreign key ([useful links]) references links ([useful links])
```



ג. טבלת חיפוש לSEARCHES: חיפוש מוצרים לפי סוג המחלקה אליה הם משתייכים.

```
CREATE TABLE [All Departments]
department Varchar(10) PRIMARY KEY
(
INSERT INTO [All Departments] VALUES
('People'), ('Movies'), ('TV Shows'), ('Comics'), ('History'), ('Theater'), ('Music')
ALTER TABLE searches
ADD CONSTRAINT FK_departments foreign key (department) references [All Departments] (department)
```

2.3. Script לבניית הטבלאות

```
create table dbo.products
)
,[product ID] int primary key not null
,[product name] varchar(40) not null
,[filter By] varchar(40) null
,[product price] money not null
,[movie show or band] varchar(50) null
,[category] varchar(40) null
,[celebrity] varchar(40) null
,[authentication] varchar(20) null
,[useful links] varchar(15) null
,[department] varchar(10) not null
(product int not null

alter table products
add constraint fk_products foreign key (product) references products ([product ID])

)create table dbo.[credit cards]
,[card number] varchar(20) primary key not null
,[csc] int not null
([Costumer email] varchar(40) not null foreign key references [Registered Costumer] ([Costumer email])
```



```
create table dbo.[expiration date]
)
,[card number] varchar(20) primary key not null foreign key references [credit cards] ([card number])
,[year] datetime not null
[month] datetime not null
(

)create table dbo.searches
,[IP address] varchar(25) not null
,[search DT] datetime not null
,[product name] varchar(40) null
,[department] varchar(10) null
,[primary key ([IP address],[search DT])
[Costumer email] varchar(40) not null foreign key references [Buyer Costumer] ([Costumer email])
(

)create table departments
, [product id] int not null
,[department] varchar(10) not null
(primary key ([product id],[department])
alter table departments
add constraint fk_departments foreign key ([product id]) references products ([product ID])

)create table dbo.orders
,[order ID] int primary key not null
,[order DT] datetime not null
,[phone number] char(18) not null
,[price money] not null
,[Costumer email Buyer] varchar(40) not null foreign key references [Buyer Costumer] ([Costumer email])
,[Costumer email Registerted] varchar(40) not null foreign key references [Registered Costumer] ([Costumer email])
([Card number] varchar(20) not null foreign key references [credit cards] ([card number])
```



```
)create table dbo.names
```

```
,[order ID] int primary key not null foreign key references orders ([order ID])
```

```
,[first name] varchar(20) not null
```

```
([last name] varchar(20) not null
```

```
)create table dbo.addresses
```

```
,[order ID] int primary key not null foreign key references orders ([order ID])
```

```
,[state] varchar(20) not null
```

```
,city varchar(20) not null
```

```
,country varchar(20) not null
```

```
([zip code] int not null
```

```
)create table dbo.Costumers
```

```
([Costumer email] varchar(40) not null primary key
```

```
)create table dbo.[Buyer Costumer]
```

```
([Costumer email] varchar(40) not null primary key foreign key references costumers ([Costumer email])
```

```
)create table dbo.[Registered Costumer]
```

```
,[Costumer email] varchar(40) not null primary key foreign key references costumers ([Costumer email])
```

```
(password varchar(20) not null
```

```
)create table dbo.[Shipping addresses]
```

```
,[Costumer email] varchar(40) not null primary key foreign key references costumers ([Costumer email])
```

```
,[state] varchar(20) not null
```

```
,city varchar(20) not null
```

```
,country varchar(20) not null
```

```
([zip code] int not null
```

```
)create table [products search]
```

```
,[IP address] varchar(25) not null
```

```
,[Products ID] int not null
```



```
([search DT] dateTime not null
```

```
alter table [products search]
```

```
add constraint PK_products_search primary key ([IP address],[Products ID],[search DT])
```

```
alter table [products search]
```

```
add constraint fk_products_search_references foreign key ([IP address],[search DT]) references searches ([IP address],[search DT])
```

```
alter table [products search]
```

```
add constraint fk_products_search_references_2 foreign key ([products ID]) references products ([product ID])
```

```
)create table [products order]
```

```
,[Products ID] int not null
```

```
([order ID] int not null
```

```
alter table[products order]
```

```
add constraint PK_products_order primary key ([products id],[order id])
```

```
alter table[products order]
```

```
add constraint pk_products_order_1 foreign key ([products id]) references products ([product ID])
```

```
alter table[products order]
```

```
add constraint pk_products_order_2 foreign key ([order ID]) references orders ([order ID])
```

2.4. SCRIPT להזנת רשומות

```
insert into Costumers values
```

```
('avinimni@gmail.com','dormicha@gmail.com','eliranatar@gmail.com','eranzehavi@gmail.com','maorbuzaglo@gmail.com',  
'osherdewida@gmail.com'),('shohamdin@gmail.com'),('daniellitman@gmail.com'),('galgadot@gmail.com')
```

```
insert into [Shipping addresses] value ('avinimni@gmail.com','israel','tel
```

```
aviv','israel',88888),('dormicha@gmail.com','america','los angeles','californya',151515),  
('maorbuzaglo@gmail.com','israel','rehovot','israel',12345),('eliranatar@gmail.com','america','las vegas','navada',908077),  
(('eranzehavi@gmail.com','israel','yahud','israel',67854),  
(osherdewida@gmail.com','israel','lod','israel',345623')
```

```
insert into [Registered Costumer] values
```

```
('dormicha@gmail.com','rakmaccabi'),('eliranatar@gmail.com','bneyyehuda'),('eranzehavi@gmail.com','pyopyo'),
```



```
('maorbuzaglo@gmail.com','admatai'), ('avinimni@gmail.com', 'shimongershon')
,('osherdewida@gmail.com','oreooo')
```

```
insert into [Buyer Costumer] values ('avinimni@gmail.com','dormicha@gmail.com'),
('shohamdin@gmail.com'),('daniellitman@gmail.com'),('galgadot@gmail.com')
```

```
insert into [credit cards] values
('123456789',161,'eliranatar@gmail.com'),('123555555',151,'dormicha@gmail.com'),('123789456',777,'eranzehavi@gmail.
com'),('987654320',341,'osherdewida@gmail.com'),('878954326',735,'maorbuzaglo@gmail.com'),
('avinimni@gmail.com',371,'556432871')
```

```
insert into [expiration date] values ('123456789','2030-11-11','2030-11-11'),('123555555','2025-11-12','2023-12-
(2023-12-30','2030-11-29','123789456'),('30
('2028-11-05','2028-11-05','878954326'),('2026-09-11','2026-09-11','987654320')
```

```
insert into orders values (124,'2022-05-20','052888888',120,'dormicha@gmail.com','dormicha@gmail.com','123555555')
,('121','2022-04-20','0509546738',2000,'avinimni@gmail.com','avinimni@gmail.com','556432871
,'avinimni@gmail.com','avinimni@gmail.com','556432871',654,'0509546738','2022-01-19',122)
,('avinimni@gmail.com','avinimni@gmail.com','556432871',890,'0509546738','2022-03-06',125)
(3,'taylorswiftCDcover',NULL,450,NULL,NULL,NULL,NULL,'music',5),(4,'adeleautographedgitar','autographed',6500,N
ULL,NULL,NULL,'autentic',NULL,'music',2),(5,'bobblyanautographedmicrophone','autographed',164,NULL,NULL,NULL,'aute
ntic',NULL,'music',2)
```

```
insert into names values (123,'dor','micha'), (121, 'avi', 'nimni')
```

```
insert into products values
(1,'maccabishirt',NULL,120,NULL,NULL,NULL,NULL,'comics',2),(2,'maccabifirstball','autographed',250,NULL,NULL,NUL
L,'autentic',NULL,'history',3)
```

```
insert into addresses values (123,'america','new york city','new tork','151515')
```

```
insert into searches values ('123456','2021-12-30
00:38:54','maccabiChamp','history','avinimni@gmail.com'),('123678','2022-11-21
21:01:33','maccabiShow','movies','dormicha@gmail.com'),('123456','2021-12-12
, ('123456','2021-12-23 ('00:42:37','maccabiShow','movies','avinimni@gmail.com
, ('123456','2021-12-27 ('00:42:37','maccabiShow','movies','avinimni@gmail.com
('00:23:37','maccabiShow','movies','avinimni@gmail.com
```

```
insert into departments values(1,'comics'),(2,'history')
('21:01:33 2022-11-21',2,'123678'),('insert into [products search] values('123456',1,'2021-12-30 00:38:54
```

```
(1,124) (insert into [products order] values (2,121),(3,122),(4,123),(5,126)
```




```
insert into [products search] values('123456',3,'2021-12-12 00:42:37'),('123456',4,'2021-12-23 00:42:37'),  
,('00:23:37 2021-12-27',5,'123456'),('00:42:37 2021-12-23',4,'123456'),('00:42:37 2021-12-12',3,'123456')  
(('00:23:37 2021-12-27',5,'123456'),'
```

2.5. SCRIPT להסרת טבלאות ואילוצים

```
drop table products  
drop table [credit cards]  
drop table [expiration date]  
drop table searches  
drop table departments  
drop table orders  
drop table names  
drop table addresses  
drop table Costumers  
drop table [Registered Costumer]  
drop table [Buyer Costumer]  
drop table [Shipping addresses]  
drop table [products search]  
drop table [products order]
```



3. שאלות

3.1. שתי שאלות SELECT ללא קינון

3.1.1. שאלת 1

שאלתה המציגה את הסכום והכמות הכוללת של ההזמנות שנעשו בשלוש השנים האחרונות מכל המחלקות בהן נעשו יותר מ-20 הזמנות.

מטרה עסקית: פלט השאלתה מסתכל על נתונים עדכניים בלבד ומסודר בסדר יורד של כמות הזמנות. באופן זה ניתן לראות שלא דווקא כמות ההזמנות היא זו המניבה את הרווח הגבוהה ביותר עבור האתר כי אם מחיר המוצרים לעיתים הוא המכריע.

```
Select department, [orders amount] = count (*), [sum orders] = sum (o.price)
From orders as o join [products order] as po on o. [order ID]= po. [order ID]
      join departments as d on d. [product id] = po. [Products ID]
Where YEAR(o.[order DT]) between YEAR (getdate())-3 and YEAR (getdate())
Group by d.department
Having count (*) > 20
Order by [orders amount] desc
```

פלט:

| | department | orders amount | sum orders |
|---|------------|---------------|------------|
| 1 | comics | 30 | 2622451.30 |
| 2 | People | 29 | 2550381.59 |
| 3 | TV shows | 23 | 3479674.24 |

3 שורות סה"כ

3.1.2. שאלת 2

שאלתה המציגה כמה פעמים בשנה לקוחות הרשומים לאתר מחפשים מוצרים במחלקת "אנשים", בסדר יורד של שנים.

מטרה עסקית: יכולת זיהוי של דפוס במידה וקיים – עלייה או ירידה בכמות חיפושים זו תעזור להבין אם יש שינוי בביקוש של אנשים למוצרים מ"אנשים"

```
Select [year] = Year(ps.[search DT]), searches = count (*)
From departments as d join [products search] as ps
On d.[product id] = ps.[Products ID] join searches as s
On ps.[IP address] = s.[IP address] and ps. [search DT] = s. [search DT]
join [Registered Costumer] as rc
On rc.[Costumer email] = s.[Costumer email]
Where d.department = 'people'
GROUP BY Year(ps.[search DT])
order by Year(ps.[search DT]) desc
```

פלט:

| | year | searches |
|---|------|----------|
| 1 | 2022 | 2 |
| 2 | 2021 | 3 |
| 3 | 2020 | 5 |
| 4 | 2019 | 10 |
| 5 | 2018 | 1 |
| 6 | 2017 | 4 |
| 7 | 2016 | 8 |
| 8 | 2015 | 3 |

8 שורות סה"כ



3.2. שתי שאלות SELECT מקוננות

3.2.1. שאלתא 1

שאלתה המציגה עבור אילו מחלקות כמות החיפושים בשנה שעברה גבוהה מכמות החיפושים בשנה שלפניה ובכמה

מטרה עסקית: הצגת המחלקות בהן נעשה שינוי חיובי בפרק זמן של שנה, מבחינת כמות הזמנות. מידע אשר יכול לכוון את הדרג הרלוונטי לשמר מדיניות מסוימת (של שיווק למשל).

```
select department=[last year].department,[the last year]=[last year].amount,[the year before]=[the year before].amount, gap = [last year].amount-[the year before].amount
from (select department=d.department, amount = COUNT(*)
from [products search] as ps join departments as d
on ps.[products ID] = d.[product ID]
where year (ps.[search DT])=year (GETDATE())-1
group by d.department) as [last year]
join (select department=d.department, amount = COUNT(*)
from [products search] as ps join departments as d
on ps.[Products ID] = d.[product ID]
where year (ps.[search DT]) = YEAR(getdate())-2
group by d.department) as [the year before]
on [last year].department = [the year before].department
where [last year].amount-[the year before].amount>0
order by [last year].amount-[the year before].amount
```

פלט:

| | department | the last year | the year before | gap |
|---|------------|---------------|-----------------|-----|
| 1 | Movies | 9 | 8 | 1 |
| 2 | People | 10 | 9 | 1 |
| 3 | comics | 8 | 6 | 2 |

3 שורות סה"כ

3.2.2. שאלתא 2

מי הם הלקוחות שביצעו כמות הזמנות גבוהה מהממוצע ומה כמות הזמנות שלהם.

הערך העסקי: מנהלי האתר מעוניינים לדעת מיהם הלקוחות בעלי מספר הזמנות הגבוה באתר ומהו מספר הזמנות הממוצע באתר.

בעזרת נתון זה יוכלו מנהלי האתר לעודד קנייה נוספת אצל לקוחות אילו, לשלוח להם פרסומים נוספים וגם לשלוח להם קופוני הנחה ובכך לעודד קנייה באתר.

```
select c.[Costumer email],[Number of orders]=COUNT(o.[order ID])
from Costumers as c join orders as o on c.[Costumer email]=o.[Costumer email Registered]
group by c.[Costumer email]
having COUNT(o.[order ID])>(
select [Average]= avg([Orders Number])
from(
select c.[Costumer email],[Orders Number]=cast(COUNT(o.[order ID])as decimal(10,2))
from Costumers as c join orders as o on c.[Costumer email]=o.[Costumer email Registered]
group by c.[Costumer email]
)as [Average])
order by [Number of orders] desc
```



פלט:

| | Customer email | Number of orders |
|----|------------------------------|------------------|
| 1 | avinimni@gmail.com | 5 |
| 2 | ahallan2o@rambler.ru | 3 |
| 3 | cvigrass3n@usgs.gov | 3 |
| 4 | dantonignetti2a@harvard.edu | 2 |
| 5 | daudibertv@t-online.de | 2 |
| 6 | dbenko2s@reference.com | 2 |
| 7 | dbroomhead13@addthis.com | 2 |
| 8 | dbuckerfield18@yale.edu | 2 |
| 9 | dchuney2v@nydailynews.com | 2 |
| 10 | dhanselman2m@stumbleupon.com | 2 |
| 11 | djorin27@naver.com | 2 |
| 12 | dmazzia3a@usgs.gov | 2 |
| 13 | dmuzzini2f@t.co | 2 |
| 14 | dornmicha@gmail.com | 2 |
| 15 | dpetr17@un.org | 2 |
| 16 | drosenfarb3i@devhub.com | 2 |
| 17 | dslatcherk@symantec.com | 2 |
| 18 | dwhilde1q@mitbeian.gov.cn | 2 |
| 19 | egratrix2@t-online.de | 2 |

19 שורות מתוך 144

3.3. שתי שאילתות SELECT מקוננות תוך שימוש בכלים מורכבים

3.3.1. שאילתא 1

אילו מוצרים מהמוצרים שהוזמנו הם ללא חתימת אותנטיות ומה מחירם.

מטרה עסקית: מנהלי האתר מעוניינים לקבל תמונת מצב על כמות המוצרים שנמכרים באתר ללא חתימת אותנטיות והאם מחירים גבוהה או נמוך.

בעזרת קבלת תמונת מצב זו יוכלו מנהלי האתר להבין האם משתלם להם למכור מוצרים ללא חתימת אותנטיות או שעליהם למזער את ההוצאות על מוצרים מסוג זה ולספקם פחות באתר. בנוסף, יוכלו לקבל מידע על רמת אמינות מוצרים אילו בעיני הלקוחות.

```
select po.[Products ID],p.[product price],p.[filter By]
from [products order] as po join products as p on po.[Products ID]=p.[product ID] join orders as o on
o.[order ID]=po.[order ID]
group by po.[Products ID], p.[product price],p.[filter By]
INTERSECT
(select p.[product ID] ,p.[product price],p.[filter By]
from products as p
group by p.[product ID],p.[product price],p.[filter By]
except
select p.[product ID] ,p.[product price],p.[filter By]
from products as p
where p.[filter By]='autographed'
group by p.[product ID],p.[product price],p.[filter By])
```



פלט:

| | Products ID | product price | filter By |
|----|-------------|---------------|-----------|
| 1 | 7 | 180.35 | unsigned |
| 2 | 8 | 603.22 | unsigned |
| 3 | 10 | 71.40 | unsigned |
| 4 | 11 | 872.74 | unsigned |
| 5 | 15 | 430.17 | unsigned |
| 6 | 17 | 272.89 | unsigned |
| 7 | 20 | 644.76 | unsigned |
| 8 | 24 | 772.46 | unsigned |
| 9 | 28 | 363.98 | unsigned |
| 10 | 29 | 508.27 | unsigned |
| 11 | 31 | 527.09 | unsigned |
| 12 | 36 | 231.72 | unsigned |
| 13 | 38 | 607.09 | unsigned |
| 14 | 41 | 710.35 | unsigned |
| 15 | 45 | 513.22 | unsigned |
| 16 | 47 | 318.62 | unsigned |
| 17 | 49 | 936.05 | unsigned |
| 18 | 52 | 871.63 | unsigned |
| 19 | 53 | 961.69 | unsigned |

19 שורות מתוך 37

3.3.2. שאלתא 2

שאלתה המעדכנת במחלקת all departments את השדה sale searches כל פעם שנעשה חיפוש נוסף בקטגוריית sale (פר מחלקה).

מטרה עסקית: באמצעות איתור המחלקות שבהן נעשה חיפוש נרחב לפי sale נוכל להעריך את התועלת בהנחות על פריטים נוספים מאותה הקטגוריה על מנת להגדיל מכירות.

```
UPDATE [All Departments]
set sale_searches = (
select searches_amount = COUNT (*)
from Costumers as c join searches as s on c.[Costumer email] = s.[Costumer email]
join [products search] as ps on s.[IP address] = ps.[IP address] and s.[search DT] = ps.[search DT]
join (select department = d.department, p_ID = p.[product ID]
from products as p join departments as d on p.[product ID] = d.[product id]
where p.[useful links] = 'SALE') as X
on ps.[Products ID] = X.p_ID
where X.department = [All Departments].department
group by X.department )
```

אחרי

לפני

7 שורות סה"כ

| | department | sale_searches | | department | sale_searches |
|---|------------|---------------|----|------------|---------------|
| ▶ | Comics | 7 | | Comics | 7 |
| | History | 4 | | History | 4 |
| | Movies | 11 | | Movies | 11 |
| | Music | NULL | | Music | NULL |
| | People | NULL | | People | NULL |
| | Theater | 4 | | Theater | 4 |
| | TV Shows | 5 | | TV Shows | 4 |
| * | NULL | NULL | ▶▶ | NULL | NULL |



4. יישומי כלים מתקדמים

4.1. View

VIEW פשוט המציג כמה מוצרים נמכרו מכל מחלקה.

מטרה עסקית: נרצה לדעת איזו מחלקה היא המבוקשת ביותר ואיזו הכי פחות מבוקשת על מנת להפעיל "דירבונים" כלשהם ללקוחות (פרסום נוסף לפי מחלקה או הנחות שמתבצעות באופן סלקטיבי).

```
CREATE VIEW V_ORDER_DEPARTMANT AS
select      d.department, amount_of_products = COUNT (*)
from        departments as d join [products order] as po
on          d.[product id] = po.[Products ID]
group by    d.department
```

| | department | amount_of_pr... |
|---|------------|-----------------|
| ▶ | comics | 60 |
| | history | 41 |
| | Movies | 49 |
| | People | 62 |
| | Theater | 41 |
| | TV shows | 48 |

4.2. functions (פונקציות)

4.2.1. פונקציה 1

פונקציה סקלרית שמקבלת מייל ובודקת כמה מהלקוחות שביצעו הזמנה, ביצעו חיפוש מקדים למוצר וגם את כמות הפעמים שביצעו את החיפוש.

מטרה עסקית: מטרת הבדיקה לבדוק האם הרכישה באתר היא יותר ספונטנית או מושכלת, במידה ונבין כי לקוחות מבצעים חיפוש מקדים בתדירות גבוהה לפני ביצוע הזמנה יכול להיות שנחליט לשלוח קוד קופון או הודעה על ירידה במחיר ונזרז את ביצוע ההזמנה שלנם

```
dbo.[countCostumer] (@input_costEmail varchar (40))
returns int
as begin
declare @output int
select @output=count (distinct [Costumer email])
from searches
where [Costumer email] like @input_costEmail
return @output
end

select [Costumer email Registered], countSum=dbo.[countCostumer]([Costumer email Registered])
from orders
order by countSum desc
```

פלט:



| | Costumer email Registered | countSum |
|----|----------------------------|----------|
| 1 | acatenot1m@nbcnews.com | 1 |
| 2 | ahallan2o@rambler.ru | 1 |
| 3 | cvigrass3n@usgs.gov | 1 |
| 4 | ahallan2o@rambler.ru | 1 |
| 5 | cvigrass3n@usgs.gov | 1 |
| 6 | agraeme42@shop-pro.jp | 1 |
| 7 | aaxworthy1i@ehow.com | 1 |
| 8 | daudibertv@t-online.de | 1 |
| 9 | drosenfarb3i@devhub.com | 1 |
| 10 | atruwert3j@symantec.com | 1 |
| 11 | bwhiston34@symantec.com | 1 |
| 12 | dwhilde1q@miitbeian.gov.cn | 1 |
| 13 | dpetr17@un.org | 1 |
| 14 | amedler2i@devhub.com | 1 |
| 15 | cbastable30@narod.ru | 1 |

15 שורות מתוך 300 (כולל אפס עבור הלקוחות שחיפוש שלהם הוביל לרכישה)

4.2.2. פונקציה 2

פונקציה שמקבלת שנה ומחזירה את טבלת הלקוחות שתוקף כרטיס האשראי הטעון במערכת יגמר במהלך השנה הנוכחית.

הערך העסקי: להתריע עבורם במהלך השנה (במייל למשל) כי יצטרכו לעדכן את כרטיס האשראי בקרוב ולהציע להם לעשות זאת מראש על מנת לקצר את זמן הרכישה באתר.

```
create function dbo.[dateCreditCard] (@input smallint)
returns table
as return
select cc.[Costumer email],ed.year
from [credit cards] as cc join [expiration date] as ed on cc.[card number]=ed.[card number]
where ed.year =@input

select *
from dbo.[dateCreditCard](2022)
```

פלט:

| | Costumer email | year |
|---|-----------------------|------|
| 1 | tcrapper24@tiny.cc | 2022 |
| 2 | maorbuzaglo@gmail.com | 2022 |
| 3 | osherdewida@gmail.com | 2022 |

3 שורות סה"כ



4.3. פרוצדורה שמורה (Stored Procedure)

פרוצדורה שמורה המקבלת חודש מסוים ומסירה את כל המוצרים שלא חופשו מעל חודש מרשימת הנמכרים ביותר - 'Best-Seller'.

```
--drop procedure update_best_saler
create procedure update_best_saler @lastMonth int
as
    update products set [useful links]=null
    from products as p join [products search] as ps on p.[product ID]=ps.[Products ID]
    where p.[useful links] like 'Best-Seller' and (@lastMonth - month(ps.[search DT]) > 0) and YEAR
(ps.[search DT])= year (getdate())

select *
from products as p join [products search] as ps on p.[product ID]=ps.[Products ID]
where YEAR(ps.[search DT]) = 2022 and p.[useful links] like 'Best-Seller'

exec update_best_saler 6
```

הערות:

1. בגלל שהנתונים הוכנסו במחולל נתונים ישנן רשומות של חיפושים עתידיים.
2. הכנסנו לתוך הפרוצדורה תנאי של שנה=2022 ע"מ שלא ימחקו כל הפרמטרים מקטגוריית "הנמכרים ביותר" לשאר השאלות, כמובן שללא העבודה היינו מסירים אותם.

פלט לפני הפעלת הפרוצדורה:

| | product ID | product name | filter By | product price | movie show or band | category | celebrity | authentication | useful links | department | product | quantity | IP address | Products ID | search DT |
|---|------------|-----------------------|-------------|---------------|-----------------------|--------------|-----------------|----------------|--------------|------------|---------|----------|----------------|-------------|-------------------------|
| 1 | 61 | Stranger in the House | autographed | 986.87 | Stranger in the House | tv photos | Serena Williams | yes | Best-Seller | History | 40 | NULL | 127.202.39.131 | 61 | 2022-11-02 00:00:00.000 |
| 2 | 57 | Six Ways to Sunday | unsigned | 453.77 | Six Ways to Sunday | movie poster | Jay Leno | no | Best-Seller | History | 248 | NULL | 160.58.65.249 | 57 | 2022-03-06 00:00:00.000 |
| 3 | 59 | Big Game | unsigned | 143.70 | Big Game | movie photos | Hilary Clinton | no | Best-Seller | TV shows | 61 | NULL | 252.218.225.66 | 59 | 2022-10-03 00:00:00.000 |
| 4 | 37 | The End of the Tour | autographed | 349.83 | The End of the Tour | music albums | Simon Cowell | no | Best-Seller | Comics | 217 | NULL | 252.24.120.15 | 37 | 2022-10-03 00:00:00.000 |
| 5 | 51 | In the Winter Dark | autographed | 597.31 | In the Winter Dark | movie poster | Fred Astaire | yes | Best-Seller | Comics | 130 | NULL | 30.115.20.90 | 51 | 2022-10-03 00:00:00.000 |
| 6 | 40 | Operation Pacific | autographed | 440.18 | Operation Pacific | tv photos | Kevin Spacey | no | Best-Seller | TV shows | 61 | NULL | 34.139.68.19 | 40 | 2022-03-06 00:00:00.000 |
| 7 | 43 | As We Were Dreaming | autographed | 102.68 | As We Were Dreaming | movie photos | Helen Keller | no | Best-Seller | Theater | 98 | NULL | 42.169.80.150 | 43 | 2022-03-06 00:00:00.000 |
| 8 | 43 | As We Were Dreaming | autographed | 102.68 | As We Were Dreaming | movie photos | Helen Keller | no | Best-Seller | Theater | 98 | NULL | 42.169.80.150 | 43 | 2022-04-05 00:00:00.000 |
| 9 | 51 | In the Winter Dark | autographed | 597.31 | In the Winter Dark | movie poster | Fred Astaire | yes | Best-Seller | Comics | 130 | NULL | 45.120.203.120 | 51 | 2022-12-06 00:00:00.000 |

9 שורות סה"כ

פלט אחרי הפעלת הפרוצדורה:

| | product ID | product name | filter By | product price | movie show or band | category | celebrity | authentication | useful links | department | product | quantity | IP address | Products ID | search DT |
|---|------------|-----------------------|-------------|---------------|-----------------------|--------------|-----------------|----------------|--------------|------------|---------|----------|----------------|-------------|-------------------------|
| 1 | 61 | Stranger in the House | autographed | 986.87 | Stranger in the House | tv photos | Serena Williams | yes | Best-Seller | History | 40 | NULL | 127.202.39.131 | 61 | 2022-11-02 00:00:00.000 |
| 2 | 59 | Big Game | unsigned | 143.70 | Big Game | movie photos | Hilary Clinton | no | Best-Seller | TV shows | 61 | NULL | 252.218.225.66 | 59 | 2022-10-03 00:00:00.000 |
| 3 | 37 | The End of the Tour | autographed | 349.83 | The End of the Tour | music albums | Simon Cowell | no | Best-Seller | Comics | 217 | NULL | 252.24.120.15 | 37 | 2022-10-03 00:00:00.000 |
| 4 | 51 | In the Winter Dark | autographed | 597.31 | In the Winter Dark | movie poster | Fred Astaire | yes | Best-Seller | Comics | 130 | NULL | 30.115.20.90 | 51 | 2022-10-03 00:00:00.000 |
| 5 | 51 | In the Winter Dark | autographed | 597.31 | In the Winter Dark | movie poster | Fred Astaire | yes | Best-Seller | Comics | 130 | NULL | 45.120.203.120 | 51 | 2022-12-06 00:00:00.000 |

5 שורות סה"כ



4.4. Trigger

טריגר המאזין לטבלת PRODUCTS ORDER ומופעל בכל פעם שנכנסת הזמנה נוספת. ברגע שזה קורה, הטריגר מעדכן את טבלת PRODUCTS בשדה QUANTITY (שמעיד על כמות הפריטים שנותרו מכל מוצר) בהתאם לתכולת ההזמנה.

מטרה עסקית: מידע תדיר ועדכני לגבי מלאי.

```
create trigger update_quantity
on [products order] for update
as
update products
set quantity = products.[quantity] - (select count (*)
                                         from [products order] as po
                                         where po.[Products ID] = products.[product id])
where [product id] in (select [product id] from inserted)
```

לפני הפעלת הטריגר:

| product ID | product name | filter By | product price | movie show or... | category | celebrity | authentication | useful links | department | product | quantity |
|------------|------------------|-------------|---------------|------------------|--------------|----------------|----------------|--------------|------------|---------|----------|
| 105 | Pont du Nord, Le | autographed | 177.7800 | Pont du Nord, Le | movie poster | Hilary Clinton | yes | sale | TV shows | 240 | 4 |

לאחר הפעלת הטריגר:

| product ID | product name | filter By | product price | movie show or... | category | celebrity | authentication | useful links | department | product | quantity |
|------------|------------------|-------------|---------------|------------------|--------------|----------------|----------------|--------------|------------|---------|----------|
| 105 | Pont du Nord, Le | autographed | 177.7800 | Pont du Nord, Le | movie poster | Hilary Clinton | yes | sale | TV shows | 240 | 3 |

השורה היחידה שעודכנה בהכנסת הזמנה



5. כלים להצגת נתונים

5.1. דו"ח עסקי

השאלה העסקית של הדו"ח: מי הסלבס שמצריהם נמכרים הכי הרבה באתר ובאיזה מדינות.

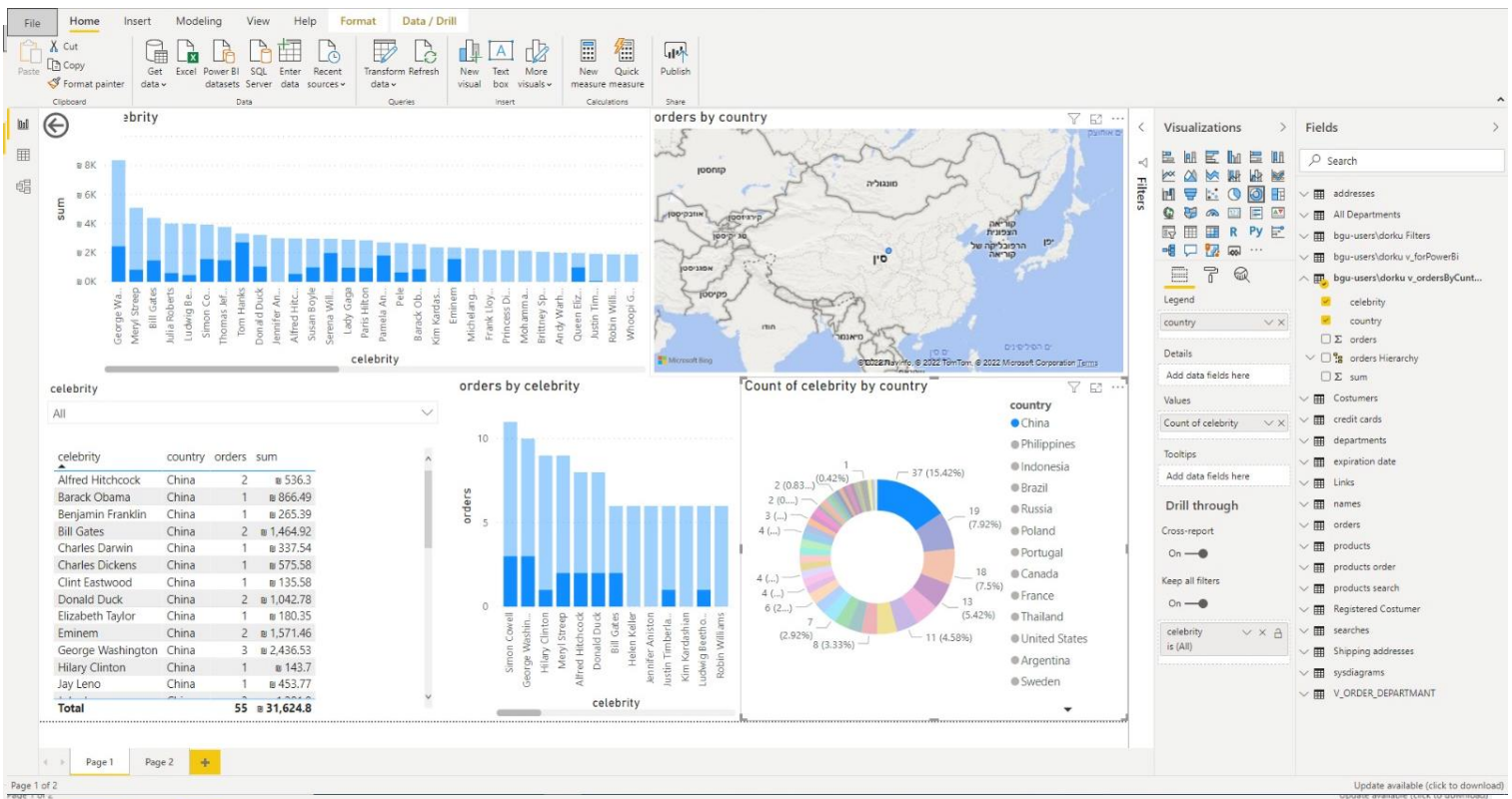
הדוח מיועד הן למחלקת שיווק ע"מ שיוכלו לדעת באיזה מדינות לפרסם מוצרים של אמנים פופולריים, לסמנכ"ל התפעול ע"מ שיוכל לדעת איזה אמן הנמכר ביותר באתר ולדאוג למכור מוצרים נוספים שלו.

דרגת הגרעיניות בינונית, מציגה את כמות המכירות של כל סלבריטאי בצורה סיכומית ומאפשר לצלול לרמה פרטנית של מיקום כל מכירת פריט.

קווים מנחים לבחירת העיצוב: מתן כלי סיכומי נוח להצגת הרווח מכל אמן וכמות ההזמנות שלו, בנוסף היה חשוב להציג טבלה סיכומית דינמית שתאפשר הצגה של כלל האינפורמציה ע"פ מדינה או אמן.

פונקציית ה view בשביל הדו"ח:

```
--drop view v_ordersByCuntry_and_celebrity--
create view v_ordersByCuntry_and_celebrity
as
select sa.country,p.celebrity,orders = count (*),sum=SUM([product price])
from [Shipping addresses] as sa join orders as o
on sa.[Costumer email]=o.[Costumer email Registered]
join [products order] as po on po.[order ID] = o.[order ID]
join products as p on p.[product ID]=po.[Products ID]
where p.celebrity is not null
group by sa.country,p.celebrity
```



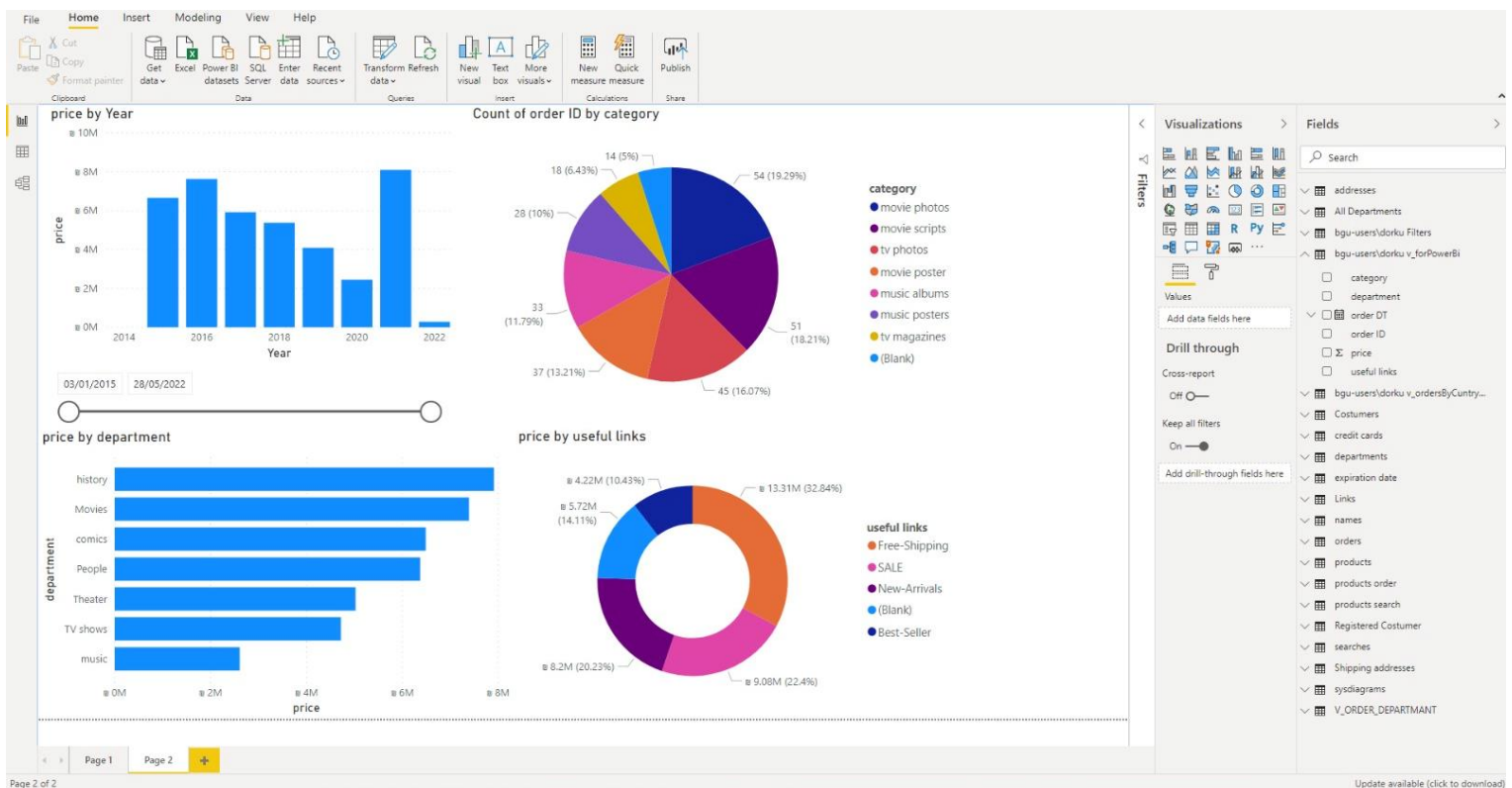


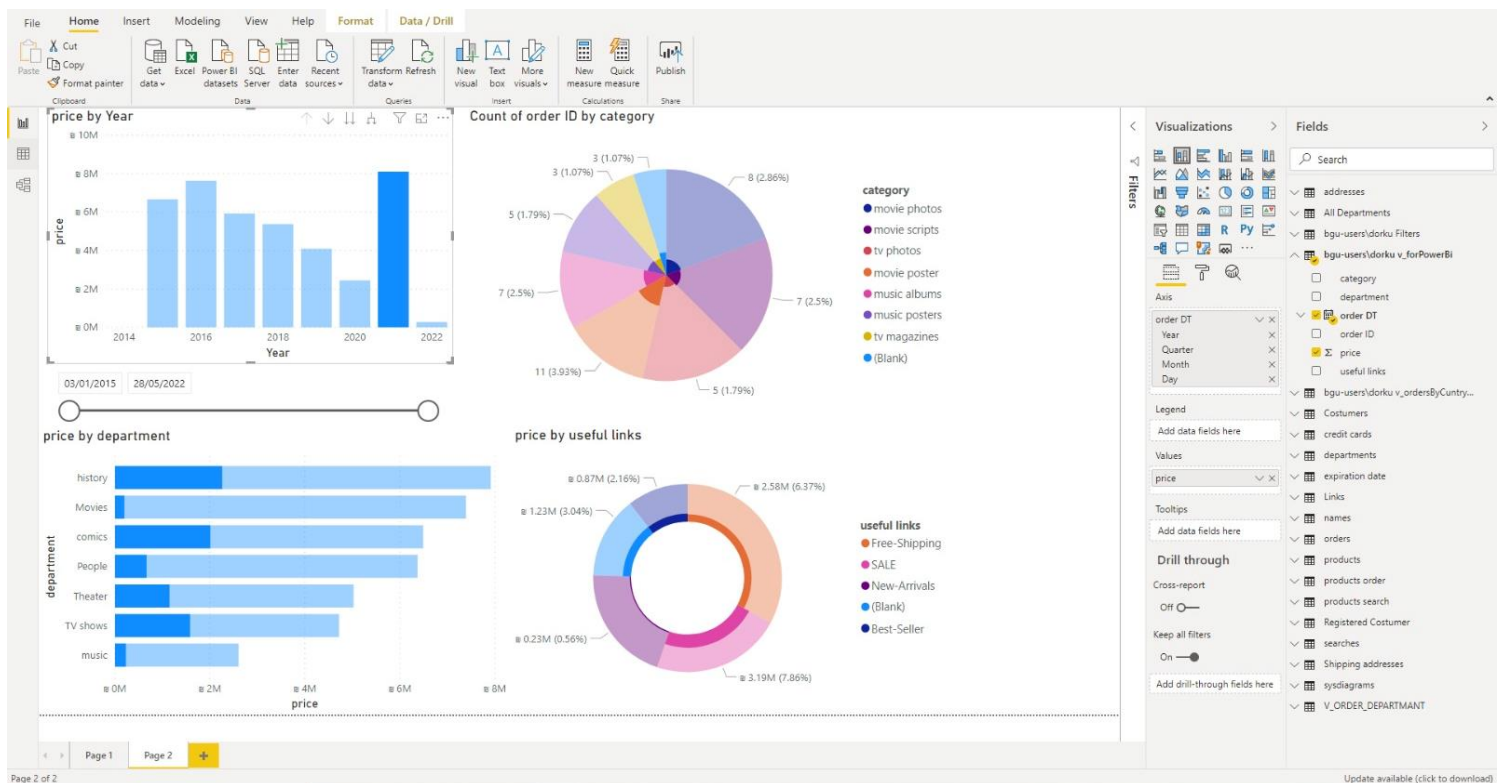
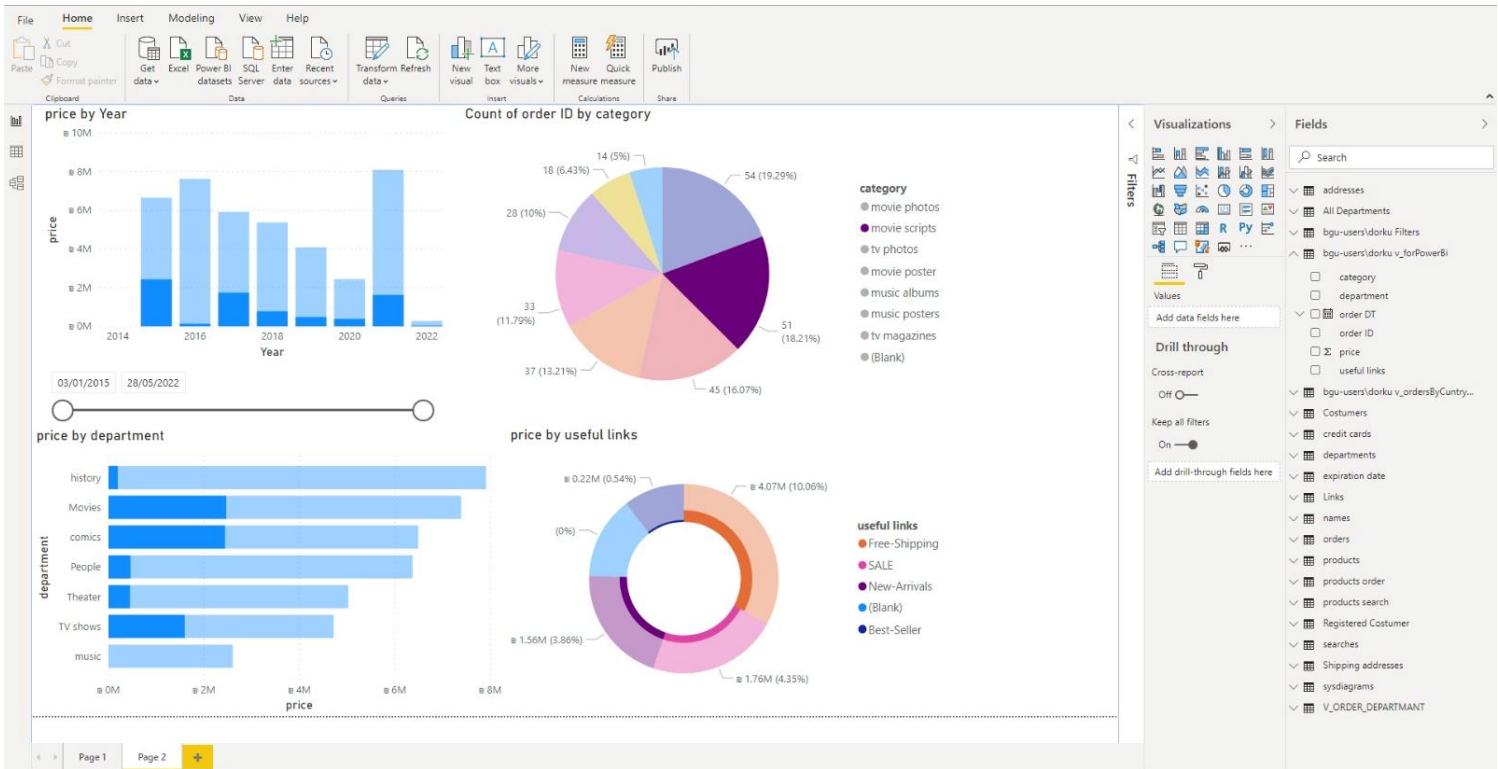
5.2. לוח מחוונים

הדו"ח הינו לוח מחוונים סיכומי למנהל החברה ומשקף את מצב החברה. הדוח מציג את הרווח על פני שנים, הקטגוריות והמחלקות הרווחיות ביותר.

הדו"ח ברמת אגרגטיביות נמוכה, מציג את מידע בצורה סיכומית וסכמתית ועתיד לעזור בקבלת החלטות אסטרטגיות כגון, קו המוצרים אותם נרצה לקדם, המחלקות שנמכרים מהן יותר מוצרים והמשקל שכל קטגוריה תופסת במכירות החברה.

```
-- drop view v_forPowerBi--
create view v_forPowerBi
as
SELECT o.[order ID], o.[order DT], o.price, p.category, p.department, p.[useful links]
FROM orders AS o JOIN [products order] AS po ON o.[order ID] = po.[order ID]
JOIN products AS p ON p.[product ID] = po.[Products ID]
WHERE (YEAR(o.[order DT]) > 2014)
```







6. יישום כלים מתקדמים נוספים

6.1. שאלות עסקיות המשלבות Window Functions

6.1.1. שאלת 1

מטרה עסקית: השאלתא משתמשת בשלוש פונקציות חלון: פונקציית SUM (כפונקציית חלון) ותפקידה לסכום את כל ההכנסות של החברה עד לתאריך המצוין ברשומה.

פונקציית LAG שמטרתה היא לחשב את ההפרש בין ההכנסות של החודש הנוכחי להכנסות של החודש הקודם, פונקציית FIRST_VALUE המציגה את ההכנסות בחודש שהיו בו ההכנסות הנמוכות ביותר באותה השנה.

מנהלי האתר יקבלו דו"ח שנתי לפי חודשים ובעזרתו יוכלו לקבל החלטות לשנים הבאות ולחזות את הכנסותיהם ולקבל החלטות אודות שינויים עתידיים לפי ההתנהלות עד כה.

טבלת VIEW המציגה את ההכנסה עבור כל הזמנה שבוצעה:

```
--drop view dbo.[V_profitByOrder]--
create view dbo.[V_profitByOrder] as
select o.[order ID],[profit per order]=SUM(p.[product price]),o.[order DT]
from [products order] as po join products as p on po.[Products ID]=p.[product ID] join orders as o on
o.[order ID]=po.[order ID]
group by o.[order ID],o.[order DT]
```

פונקציה המציגה את ההכנסה עבור כל שנה:

```
--drop function dbo.[ProfitPerYear]--
create function dbo.[ProfitPerYear] (@year smallint, @month tinyint)
returns real
as begin
    declare @profit real
    select @profit= SUM([profit per order])
    from dbo.[V_profitByOrder]
    where @month=MONTH([order DT]) AND @year= YEAR([order DT])
    return @profit
end
```

טבלת VIEW המציגה את כלל ההכנסות:

```
--drop view dbo.[V_TotalProfitReport]--
create view dbo.[V_TotalProfitReport] as
select year,month
    ,[profit per year and month]
    ,[Cumulative profit]= SUM([profit per year and month]) over (order by year,month)
    ,[difference from last month]=[profit per year and month]-(LAG([profit per year and
month],1) over (order by year,month))
    ,[minimum profit in this year]=FIRST_VALUE([profit per year and month]) over(partition by
year order by [profit per year and month])
from (select distinct year= year([order DT])
    ,month= month([order DT])
    ,[profit per year and month]= dbo.[ProfitPerYear] (year([order DT]),month([order DT]))
    from orders
    )as best
where [profit per year and month] is not null
```

שימוש:

```
--drop procedure dbo.[ProfitReport]--
create procedure dbo.[ProfitReport] (@year smallint, @month1 tinyint, @month2 tinyint) as
select *
```



```
from dbo.[V_TotalProfitReport]
where year=@year AND month between @month1 and @month2
order by month
```

הפעלה:

```
EXECUTE dbo.[ProfitReport] 2021,1,12
```

פלט:

11 שורות סה"כ

| | year | month | profit per year and month | Cumulative profit | difference from last month | minimum profit in this year |
|----|------|-------|---------------------------|-------------------|----------------------------|-----------------------------|
| 1 | 2021 | 2 | 3522.94 | 114750.640511513 | 3082.76 | 418.61 |
| 2 | 2021 | 3 | 418.61 | 115169.250496864 | -3104.33 | 418.61 |
| 3 | 2021 | 4 | 2607.94 | 117777.190438271 | 2189.33 | 418.61 |
| 4 | 2021 | 5 | 4650.71 | 122427.900399208 | 2042.77 | 418.61 |
| 5 | 2021 | 6 | 1974.41 | 124402.310433388 | -2676.3 | 418.61 |
| 6 | 2021 | 7 | 1047.85 | 125450.160408974 | -926.5601 | 418.61 |
| 7 | 2021 | 8 | 1012.34 | 126462.500435829 | -35.50995 | 418.61 |
| 8 | 2021 | 9 | 3862.81 | 130325.310494423 | 2850.47 | 418.61 |
| 9 | 2021 | 10 | 1155.65 | 131480.960518837 | -2707.16 | 418.61 |
| 10 | 2021 | 11 | 9144.34 | 140625.300362587 | 7988.69 | 418.61 |
| 11 | 2021 | 12 | 6739.09 | 147364.390206337 | -2405.25 | 418.61 |

6.1.2. שאילתא 2

מטרה עסקית: שאילתא המבצעת דירוג של המחלקות לפי מספר ההזמנות שבוצעו. הפונקציות בהן השתמשנו הן: RANK שקובעת את הדירוג עבור כל מחלקה ופונקציית LAST_VALUE המציגה את החודש בעל הדירוג הנמוך ביותר עבור מחלקה זו.

בדרך זו מנהלי האתר יוכלו לקבל מידע עבור הפופולריות של המחלקות בהתאם לחודשי השנה ובהתאם לכך לפרסם מוצרים מסוימים בזמנים רלוונטיים, להוסיף מוצרים ממחלקות מסוימות בזמנים שהן הפופולריות ביותר וכדומה.

בעת שימוש בתהליך השמור שמקבל מחלקה ודירוג כלשהו יוצג דו"ח שמציג את החודשים בהם המחלקה הייתה בדירוג שהזון, את החודש בעל מספר ההזמנות הנמוך ביותר וכן את מספר הזמנות זה וההפרש בין סה"כ ההזמנות באותו חודש לבין החודש הנמוך ביותר (עבור כל רשומה).

טבלת VIEW המציגה את ההזמנות עבור כל מחלקה:

```
--drop view dbo.[v_salses for department]--
create view dbo.[v_salses for department] as
select d.department, [month]=MONTH(o.[order ID]), [Total orders]=COUNT(*)
from [products order] as po join departments as d on po.[Products ID]=d.[product id] join orders as o on
po.[order ID]=o.[order ID]
group by d.department, MONTH(o.[order ID])
```

טבלת VIEW המציגה את הדירוג עבור כל מחלקה לפי מספר ההזמנות שבוצעו:

```
--drop view dbo.[v_departments ranks]--
create view dbo.[v_departments ranks] as
select department,[month], [Total orders], RANK() over( partition by [month] order by [Total orders] desc)
rank, LAST_VALUE([month])
over(partition by department
```




```
order by [Total orders] desc
rows between unbounded preceding and unbounded following) [Lowest month for department]
from dbo.[v_sales for department]
group by department,[month], [Total orders]
```

שימוש:

```
--drop procedure dbo.[Department report]--
create procedure dbo.[Department report]
@department varchar(20), @rank int
as
select MONTH, [Total orders], [Lowest month for department],
[Total orders in the lowest month]=
(select [Total orders]
from dbo.[v_departments ranks]
where department=@department and [month]=[Lowest month for department]),
[Diffrence with the lowest month]=([Total orders]-
(select [Total orders]
from dbo.[v_departments ranks]
where department=@department and [month]=[Lowest month for department]))
from dbo.[v_departments ranks]
where department=@department and [rank]=@rank
```

הפעלה:

```
EXECUTE dbo.[Department report] 'Comics', 1
```

פלט:

| | MONTH | Total orders | Lowest month for department | Total orders in the lowest month | Diffrence with the lowest month |
|---|-------|--------------|-----------------------------|----------------------------------|---------------------------------|
| 1 | 8 | 7 | 4 | 2 | 5 |
| 2 | 2 | 6 | 4 | 2 | 4 |
| 3 | 10 | 6 | 4 | 2 | 4 |

3 שורות סה"כ



6.2. שילוב מערכתי של מספר כלים

שתי פרוצדורות המשנות את שדה useful links בטבלת products בהתאם ל-25 המוצרים שהזמינו מהם הכי הרבה. מוצרים שלא נמצאים יותר ב-25 המוזמנים ביותר, יהפכו מ-best seller ל-free shipping. מנת לאפשר הגדלת מכירות מחודשת ולמשוך את הקהל אליהם (ואולי להחזירם להיות best seller). מוצרים שלא נמצאים ב-25 המוזמנים ביותר ובעקבות הזמנות אחרונות נכנסו לקטגוריה זו, יהפכו ל-best seller על מנת למשוך את תשומת ליבם של הלקוחות למוצרים פופולרים.

Trigger המאזין לטבלת products order ובכל הזמנה חדשה מעדכן את עמודת sold בטבלת products.

```
-----trigger-----
create trigger sold
on [products order]
for insert
as
update products
set sold = (select distinct count (*)
            from [products order] as po
            where po.[Products ID] = products.[product ID])
where [product ID] in (select [product ID] from inserted)
```

view המחשב ומציג את 25 המוצרים שנמכרו הכי הרבה עד כה.

```
-----VIEW-----
CREATE VIEW [most sold] AS
SELECT TOP 25 productID = P.[Product ID], sold = p.sold
FROM products as p
GROUP BY p.sold, [product ID]
ORDER BY p.sold desc
```

שתי הפרוצדורות האחראיות לשינוי של USEFULL LINKS

```
-----PROCEDURE1-----
--DROP PROCEDURE [Update best seller]

CREATE PROCEDURE [Update best seller]
AS
BEGIN
    UPDATE PRODUCTS
    SET [useful links] = 'Best-Seller'
    WHERE [product ID] in (SELECT distinct productID
                          FROM [most sold])

END

-----PROCEDURE2-----
--DROP PROCEDURE [Update not best seller]

CREATE PROCEDURE [Update not best seller]
AS BEGIN
    UPDATE PRODUCTS
    SET [useful links] = 'Free-Shipping'
    WHERE [product ID] in (select [product ID] from products where [useful links] = 'Best-Seller'
                             except
                             select productID from [most sold])

END
```

הפעלה:

```
EXECUTE [Update not best seller]
EXECUTE [Update best seller]
```




לפני הפעלת הטריגר והפרוצדורות:

| product ID | product name | filter By | product price | movie show or... | category | celebrity | authentication | useful links | department | product | quantity | sold |
|------------|--------------------|-------------|---------------|--------------------|---------------|-------------------|----------------|---------------|------------|---------|----------|------|
| 1 | maccbishirt | NULL | 120.0000 | Flesh & Blood | tv photos | NULL | no | Best-Seller | comics | 2 | 4 | 20 |
| 2 | maccabifirstball | autographed | 250.0000 | As We Were Dr... | NULL | Meryl Streep | yes | Free-Shipping | history | 3 | 18 | 32 |
| 3 | taylorswiftCDc... | NULL | 450.0000 | Shrooms | tv photos | Justin Timberlake | no | sale | music | 5 | 4 | 38 |
| 4 | adeleautograph... | autographed | 6500.0000 | City of Joy | NULL | Simon Cowell | yes | NULL | music | 2 | 7 | 23 |
| 5 | bobdylanautog... | autographed | 164.0000 | NULL | NULL | NULL | yes | Free-Shipping | music | 2 | 18 | 13 |
| 6 | Delgo | autographed | 938.0900 | Delgo | movie scripts | George Washin... | no | sale | TV shows | 59 | 16 | 31 |
| 7 | Crimson Perma... | unsigned | 180.3500 | Crimson Perma... | music albums | Elizabeth Taylor | no | Free-Shipping | TV shows | 283 | 16 | 29 |
| 8 | On a march? su... | unsigned | 603.2200 | On a march? su... | movie photos | Justin Timberlake | no | Best-Seller | People | 87 | 5 | 13 |
| 9 | Flesh & Blood | autographed | 61.5700 | Flesh & Blood | tv magazines | Marilyn Monroe | no | sale | Comics | 234 | 18 | 40 |
| 10 | Brass Target | unsigned | 71.4000 | Brass Target | movie poster | Roger Federer | no | Best-Seller | People | 11 | 7 | 16 |
| 11 | Drishyam | unsigned | 872.7400 | Drishyam | tv photos | Ludwig Beetho... | no | Free-Shipping | Theater | 25 | 8 | 18 |
| 12 | Lloyds of London | autographed | 38.7500 | Lloyds of London | movie photos | Mark Twain | no | New-Arrivals | History | 30 | 16 | 19 |
| 13 | Everly | autographed | 303.3000 | Everly | tv photos | Robin Williams | yes | Free-Shipping | Movies | 213 | 5 | 18 |
| 14 | Against All Odds | autographed | 328.3700 | Against All Odds | music albums | Robin Williams | no | NULL | History | 71 | 10 | 17 |
| 15 | Comedy of Terr... | unsigned | 430.1700 | Comedy of Terr... | music albums | Donald Duck | no | New-Arrivals | TV shows | 270 | 5 | 12 |
| 17 | General's Daug... | unsigned | 272.8900 | General's Daug... | movie scripts | Bob Hope | no | Free-Shipping | Comics | 263 | 4 | 9 |
| 18 | Casablanca | autographed | 769.5900 | Casablanca | tv photos | Michelangelo | no | Best-Seller | Theater | 161 | 11 | 11 |
| 19 | Head | autographed | 265.3900 | Head | tv photos | Benjamin Frank... | no | New-Arrivals | History | 74 | 5 | 17 |
| 20 | Bigger Than the... | unsigned | 644.7600 | Bigger Than the... | tv photos | Pele | yes | Best-Seller | Comics | 142 | 5 | 13 |

אחרי הפעלת הטריגר והפרוצדורות:

| product ID | product name | filter By | product price | movie show or... | category | celebrity | authentication | useful links | department | product | quantity | sold |
|------------|--------------------|-------------|---------------|--------------------|---------------|-------------------|----------------|---------------|------------|---------|----------|------|
| 1 | maccbishirt | NULL | 120.0000 | Flesh & Blood | tv photos | NULL | no | Best-Seller | comics | 2 | 4 | 49 |
| 2 | maccabifirstball | autographed | 250.0000 | As We Were Dr... | NULL | Meryl Streep | yes | Best-Seller | history | 3 | 18 | 73 |
| 3 | taylorswiftCDc... | NULL | 450.0000 | Shrooms | tv photos | Justin Timberlake | no | sale | music | 5 | 4 | 38 |
| 4 | adeleautograph... | autographed | 6500.0000 | City of Joy | NULL | Simon Cowell | yes | NULL | music | 2 | 7 | 23 |
| 5 | bobdylanautog... | autographed | 164.0000 | NULL | NULL | NULL | yes | Best-Seller | music | 2 | 18 | 56 |
| 6 | Delgo | autographed | 938.0900 | Delgo | movie scripts | George Washin... | no | sale | TV shows | 59 | 16 | 31 |
| 7 | Crimson Perma... | unsigned | 180.3500 | Crimson Perma... | music albums | Elizabeth Taylor | no | Best-Seller | TV shows | 283 | 16 | 64 |
| 8 | On a march? su... | unsigned | 603.2200 | On a march? su... | movie photos | Justin Timberlake | no | Free-Shipping | People | 87 | 5 | 40 |
| 9 | Flesh & Blood | autographed | 61.5700 | Flesh & Blood | tv magazines | Marilyn Monroe | no | sale | Comics | 234 | 18 | 40 |
| 10 | Brass Target | unsigned | 71.4000 | Brass Target | movie poster | Roger Federer | no | Free-Shipping | People | 11 | 7 | 16 |
| 11 | Drishyam | unsigned | 872.7400 | Drishyam | tv photos | Ludwig Beetho... | no | Best-Seller | Theater | 25 | 8 | 66 |
| 12 | Lloyds of London | autographed | 38.7500 | Lloyds of London | movie photos | Mark Twain | no | Best-Seller | History | 30 | 16 | 50 |
| 13 | Everly | autographed | 303.3000 | Everly | tv photos | Robin Williams | yes | Free-Shipping | Movies | 213 | 5 | 35 |
| 14 | Against All Odds | autographed | 328.3700 | Against All Odds | music albums | Robin Williams | no | NULL | History | 71 | 10 | 22 |
| 15 | Comedy of Terr... | unsigned | 430.1700 | Comedy of Terr... | music albums | Donald Duck | no | New-Arrivals | TV shows | 270 | 5 | 41 |
| 17 | General's Daug... | unsigned | 272.8900 | General's Daug... | movie scripts | Bob Hope | no | Free-Shipping | Comics | 263 | 4 | 10 |
| 18 | Casablanca | autographed | 769.5900 | Casablanca | tv photos | Michelangelo | no | Free-Shipping | Theater | 161 | 11 | 36 |
| 19 | Head | autographed | 265.3900 | Head | tv photos | Benjamin Frank... | no | Best-Seller | History | 74 | 5 | 62 |
| 20 | Bigger Than the... | unsigned | 644.7600 | Bigger Than the... | tv photos | Pele | yes | Free-Shipping | Comics | 142 | 5 | 19 |



6.3. דו"ח המושתת על שימוש בפסקת WITH מורכבת

דו"ח נתונים המסתמך על with, הדוח מפרט על המוצרים שהוזמנו באתר ומנסה לנתח מהם השלכות לעתיד לגבי דפוס ההזמנות באתר.

בין הנתונים הדוח מציג את שם המוצר, מס ההזמנה, מספר החיפוש שהלקוח ביצע לפני שהזמין אותו, מספר החודשים שהמתין עד שהזמין (מחושב ע"י חיסור מהחודש בו הזמין פחות החודש בו חיפש לראשונה), האם המוצר יקר (האם מחירו גבוהה מהממוצע באתר) והאם הוא פופולרי (הוזמן מעל פעמיים).

הפרמטרים המוצגים מיועדים להסקת מסקנות אודות המוצרים, אילו מוצרים יכולנו להוריד את מחירים כאשר חופשו ולא נקנו ע"מ למקסם מחירות, ואילו ימכרו בכל מקרה.

הפונקציה searches_not_Buy משפיע ישירות על 2 העמודות הראשונות בטבלה (חיפושים לפני הזמנה והזמן שעבר עד להזמנה הראשונה).

הפונקציה expensive_products חישבה את הממוצע באתר ובעזרתו בדקנו האם המוצר יקר.

הפונקציה המורכבת popular_product בדקה אילו מוצרים הוזמנו מעל פעמיים ממנה בדקנו האם המוצרים שהוזמנו נחשבו פופולרי.

הקוד של with ולאחריו הפונקציה:

```
with searches_not_Buy as (select ps.[Products ID] , ps.[IP address], s.[search DT], s.[product name], p.[product price]
                           from searches as s join [products search] as ps on s.[search
DT]=ps.[search DT]
                           join products as p on s.[product name]=p.[product name]),
expensive_products as (select avgg=avg([product price])
                           from products),
popular_product as (select p.[product ID]
                     from products as p
                     where( select count (*)
                           from [products order] as po
                           where po.[Products ID]=p.[product ID]>2)

select snb.[product name], o.[order ID], count_of_search_befor=count(*),
time_waiting_by_month=month(max([order DT]))-month(min([search DT])),
expensive= case when (snb.[product price]) > (select avgg from expensive_products)
then 'true' else 'false'
end,
popular = ( case when (select count(*) from popular_product as p where p.[product ID]=snb.[Products ID]) >0
then 'popular' else 'false' end)
from orders as o join [products order] as po on o.[order ID]=po.[order ID]
join searches_not_Buy as snb on snb.[Products ID]=po.[order ID]
where [search DT]<[order DT] and year( [order DT]) = year ([search DT])
group by snb.[product name], o.[order ID], snb.[product price] , snb.[Products ID]
```



הפלט:

| | product name | order ID | count_of_search_befor | time_waiting_by_month | expensive | popular |
|----|-------------------------|----------|-----------------------|-----------------------|-----------|---------|
| 1 | As We Were Dreaming | 43 | 1 | 4 | false | popular |
| 2 | Big Game | 66 | 1 | 0 | false | popular |
| 3 | Bigger Than the Sky | 62 | 1 | 4 | true | popular |
| 4 | Captains Courageous | 26 | 3 | 6 | true | popular |
| 5 | Caveman | 26 | 3 | 6 | true | popular |
| 6 | Christine | 53 | 1 | 9 | false | popular |
| 7 | Christine | 86 | 1 | 4 | false | popular |
| 8 | Comanche Station | 7 | 4 | 6 | true | popular |
| 9 | Comanche Station | 83 | 1 | 8 | true | popular |
| 10 | Corpo Celeste | 7 | 4 | 6 | false | popular |
| 11 | Corpo Celeste | 36 | 1 | 5 | false | popular |
| 12 | Corpo Celeste | 83 | 1 | 8 | false | popular |
| 13 | Crimson Permanent ... | 7 | 4 | 6 | false | popular |
| 14 | Crimson Permanent ... | 83 | 1 | 8 | false | popular |
| 15 | Foo Fighters: Back a... | 7 | 4 | 6 | true | popular |

15 שורות מתוך 55