|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| מס' קבוצה | אתר | | | תאריך הגשה |
| 60 | [www.attractiontickets.com](http://www.attractiontickets.com) | | | 15/04/2022 |
| חברי הצוות - מספרי ת.ז | | | | |
| 316320779 | | 207941105 | 315673186 | |

## פרויקט בסיסי נתונים

## מטלה 1 (30%) – רקע כללי על הארגון והאתר

Click to go to website or internet line art icon for apps and websites **על האתר**

***AttractionTickets.com*** הוא אתר לרכישת כרטיסים וחבילות מוזלות לאטרקציות ברחבי העולם. הלקוח מזמין מראש כרטיס/ חבילה לאטרקציות ומקבל במשלוח מהיר או בר-קוד שאיתו יכול להיכנס ישירות למקום מבלי להמיר את הכרטיס בכניסה. מטרתו האתר היא לתת שירות המוסיף אושר לחופשה ולהקל על הלקוחות ברכישת כרטיסים לאטרקציות תוך כדי שמירה על רווחיות כלכלית. האתר הוקם בשנת 2002 על מנת להוזיל את מחירי האטרקציות המובילות בעולם ולהקל על הצרכנים ברכישת הכרטיסים. בנוסף למכירת הכרטיסים לאטרקציות יש לחברה מוניטין מיוחד בשירות המופתי שהם מספקים ללקוחות פוטנציאליים וממשיים כאחד. Attraction Tickets מציעה גישה לאתר מתעדכן וקל לשימוש, חוברת חינם, מדריך מקוון לאורלנדו, בלוג ואפשרות ליצור קשר ישירות בטלפון עם מומחי אורלנדו של החברה. לחברה משרדים ואתרי אינטרנט ייעודיים בברלין, דבלין ולונדון, שם שוכן מטה החברה. האתר משרת למעלה מ-250,000 לקוחות בשנה, רובם נוסעים לאורלנדו פלורידה. בשנת 2006 הפכה החברה לחברה הראשונה אי פעם שהצליחה למכור יותר ממיליון כרטיסים לאתר דיסני וורלד בפלורידה בשנה אחת בלבד, מאז החברה ממשיכה למכור בקצב הזה ושואפת להיות המוכרת הגדולה בעולם ולהגיע במחיר הנוח ביותר ללקוח.

 **מטרותיו העסקיות**

האתר מאמין בחוויית הלקוח שמגיעה עם רכישת הכרטיסים ולכן שואף להיות האתר המוביל בעולם במכירת כרטיסים וחבילות לאטרקציות. כדי לבסס את מטרתו העסקית האתר מתחייב לתת ללקוחותיו את המחיר הזול ביותר ולכן מציע לשלם להם את הפרש המחיר אם ימצאו את אותם כרטיסים במחיר זול יותר באתר אחר.

 **אתגרים**

בדומה למתרחש בחברות רבות גם חברת Attraction Tickets מתמודדת עם אתגרים שונים ומגוונים. אתגר בולט הוא ה**תחרותיות במשק**. החברה חרטה על דגלה לספק באופן תמידי את המחיר הזול ביותר ללקוח, על מנת לעמוד בהבטחה זו על החברה לעשות סקרי שוק מתמידים אצל מתחריה ולוודא שמחיריה אכן זולים יותר, אחרת החברה משווה את מחיריה למחירי המתחרים. בנוסף, האתר משרת לקוחות מכל העולם וכדי **לשרת אותם בצורה הטובה ביותר** האתר פועל במספר שפות שונות ונותן ייעוץ מומחים 12 שעות ביממה. האתר מספק כרטיסים רשמיים לאתרים, לכן מחויב **לעבוד באופן ישיר עם מערכות הכרטוס של האטרקציות**. כל האתגרים האלו ניצבים ביחד עם האתגר הגדול ביותר בעידן הטכנולוגיה שהוא פיתוח מהיר והופעת חברות חדשות בשוק לעיתים תכופות, כדי להישאר רלוונטי על האתר **להתמיד בפיתוח והתחדשות** .

 **אתרים בעלי אופי דומה**

[**Zebratix**](https://www.zebratix.com/)הינו אתר ישראלי לרכישת כרטיסים **לחוויות**, אתרים ואטרקציות המתמקד בתאריכים. החברה מתמקדת ביצירה, חדשנות ופיתוח מערכות לטובת לקוחותיה ונוחיותם. בשונה מ Attraction Tickets הלקוחות מקבלים כרטיסים באמצעות ברקוד/ כרטיס מודפס ואותם מראים לקופאי בכניסה לאתר וממירים לכרטיס רשמי לאטרקציה, השוני הוא שהלקוח לא חוסך את ההמתנה בתור לקופאי.

[**Tiqets**](https://www.tiqets.com/en/)הינו אתר בינלאומי ל**רכישת כרטיסים וחבילות לאטרקציות בעולם בצורה נגישה**. האתר הוקם במטרה למנוע מלקוחותיו להמתין בתורים ארוכים כדי לגשת לאטרקציות. הלקוחות מקבלים את כרטיסים דיגיטליים באופן מיידי ונכנסים איתם ישירות לאטרקציה. בשונה מ Attraction Tickets האתר זמין במגוון רב יותר של שפות ומעניק תמיכה ללקוחות 24/7. בנוסף Attraction Tickets מציע דילים מוגבלים בזמן לתאריכים ספציפיים כדי לעודד רכישות, בעוד Tiqets מציעים מבצעים לאטרקציות ולא לפי תאריך.

## מטלה 2 (40%) – התהליך העסקי

תהליכים עסקיים בארגון כוללים מגוון תחומים: שירות לקוחות, פיתוח ואחזקה, בטיחות אשראי ורכישת כרטיסים לפארקי שעשועים ואטרקציות תיירות, כולל חבילות. האתר AttractionTickets.com כולל גם הזמנות חופשה במלונות, אנו נתמקד בתהליך העסקי "רכישת כרטיסים וחבילות מוזלות לאטרקציות ברחבי העולם". משתמש נכנס לאתר בתור אורח ולא נדרש להירשם לאתר טרם רכישת הכרטיסים. לאחר ביצוע רכישה ניתן להיכנס לפרטי ההזמנה באתר באמצעות הזנת כתובת המייל ומספר ההזמנה. בתהליך העסקי בו אנו מתמקדים, המשתמש יבחר את האטרקציה אליה ירצה לקנות כרטיס. תחילה ייקבע אופן קבלת הכרטיס\* (כרטיס נייר או כרטיס דיגיטלי), אחר כך נדרש מהמשתמש לבחור תאריך לאטרקציה\* ואז לבחור האם להזמין כרטיס סטנדרטי או משודרג \*\*(בעלות גבוהה יותר- תהליך זה חשוב מכיוון שהוא מייצר הכנסה משנית נוספת לאתר), לציין לאיזה גיל מיועד כל כרטיס\* ובנוסף לזה ישנם שלושה מסלולים שהלקוח יכול לבחור\*\*.

1. מסלול STANDARD: מכיל כיסוי קורונה ואפשרות לשינוי הכרטיס.
2. מסלול FLEX: מכיל את הכיסויים של ביטוח STANDARD ובנוסף כיסוי למקרה של נזק, גניבה או אבדה של הכרטיסים ואפשרות החלפת ואיסוף של כרטיסים מודפסים באורלנדו.
3. מסלול FLEX PLUS : מכיל את הכיסויים של ביטוח FLEX ובנוסף מציע החזר מלא על כרטיסים שלא נוצלו ואפשרות לביטול בחינם מכל סיבה שהיא.

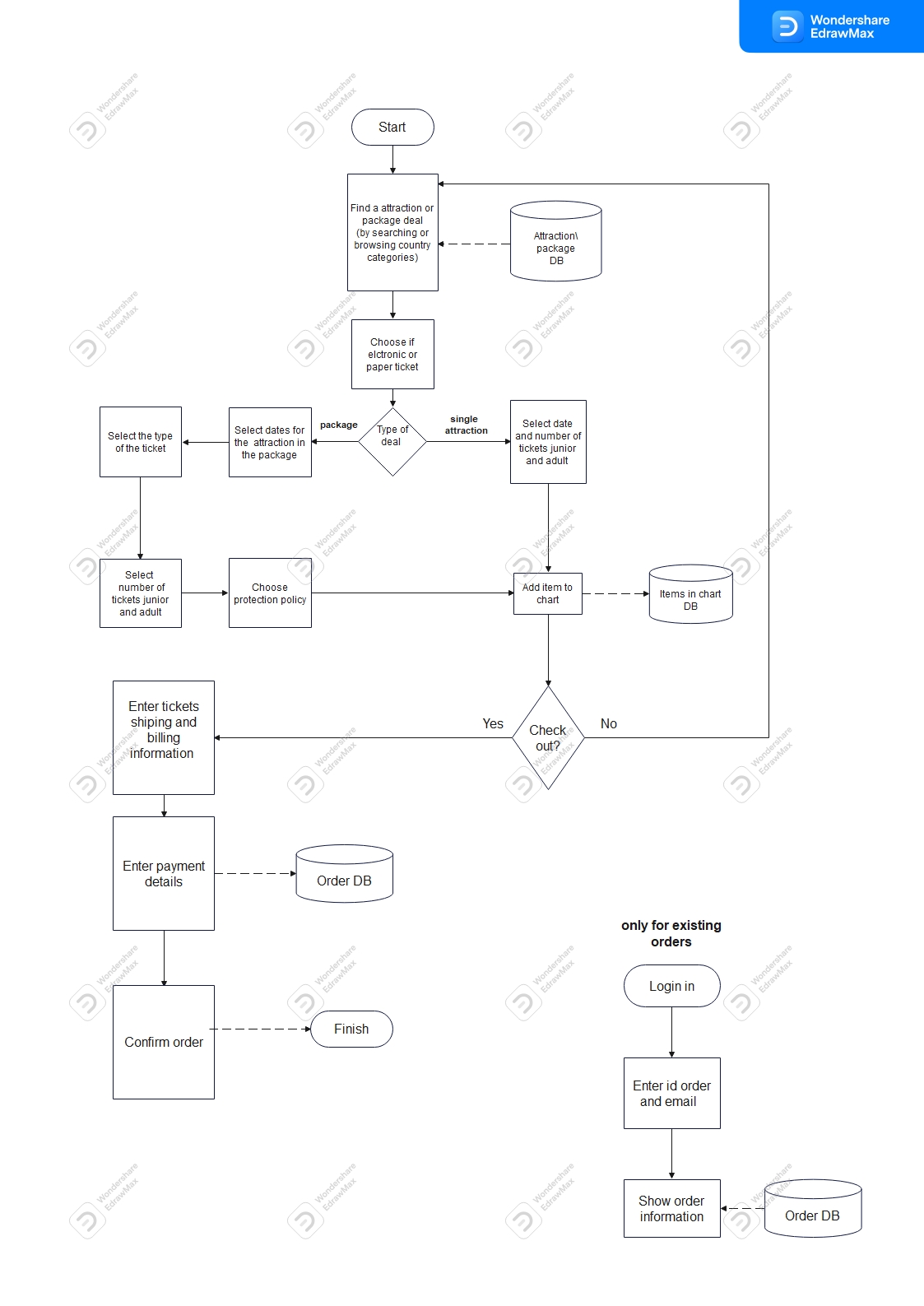
לאחר מכן יופיע פירוט העסקה, נוסיף את העסקה לסל הרכישה. בסל הרכישה יופיעו הכרטיסים ומתנות מהאתר. תהליך עסקי זה חשוב מכיוון שהוא נותן ללקוח הרגשה שמפנקים אותו וזוהי מטרת האתר – לספק ללקוח חוויה. הלקוח יכול להחליט אם להוסיף פריטים לסל או לבצע את הרכישה. כעת הלקוח יצטרך להזין את שמות משתמשי הכרטיסים ואת כתובת המייל של הרוכש\* (שבאמצעותו המערכת מזהה את הלקוח), לאחר מכן יש להזין את פרטי הלקוח לשליחת החשבונית \*(שם פרטי, שם משפחה, כתובת - רחוב, כתובת -עיר, כתובת -מדינה, כתובת -מיקוד, כתובת – מחוז(אם קיים), מס' פלאפון, מס' טלפון), בנוסף ניתן למצוא את הכתובת בעזרת הזנת מיקוד\*\*. במידה והלקוח בחר בקבלת כרטיסי נייר, ניתנת ללקוח שתי אופציות, או לקבל משלוח לכתובת שליחת החשבונית או למלא מחדש כתובת למשלוח הכרטיסים\*. לאחר מכן ניתנות 4 אופציות לסוג המשלוח\*\*:

1. משלוח סטנדרטי לאנגליה (בחינם ותוך 7 ימי עסקים)
2. משלוח מהיר לאנגליה (בתוספת 4 פאונד תוך 3 ימי עסקים)
3. משלוח ליום למחרת לאנגליה (בתוספת 6.9 תוך יום עסקים אחד)
4. משלוח לחו"ל (בתוספת 18 פאונד)

ניתן לבחור בשתי צורות תשלום\*, באמצעות קוד שובר או הזנת פרטי כרטיס אשראי (מסוג ויזה, מאסטר קארד). לבסוף מאשרים את תקנון האתר והפרטיות\*, מאשרים ומבינים כיצד בוחרים את תאריכי ההזמנה\*, קבלת פרסומים ומבצעים מהאתר \*\*.

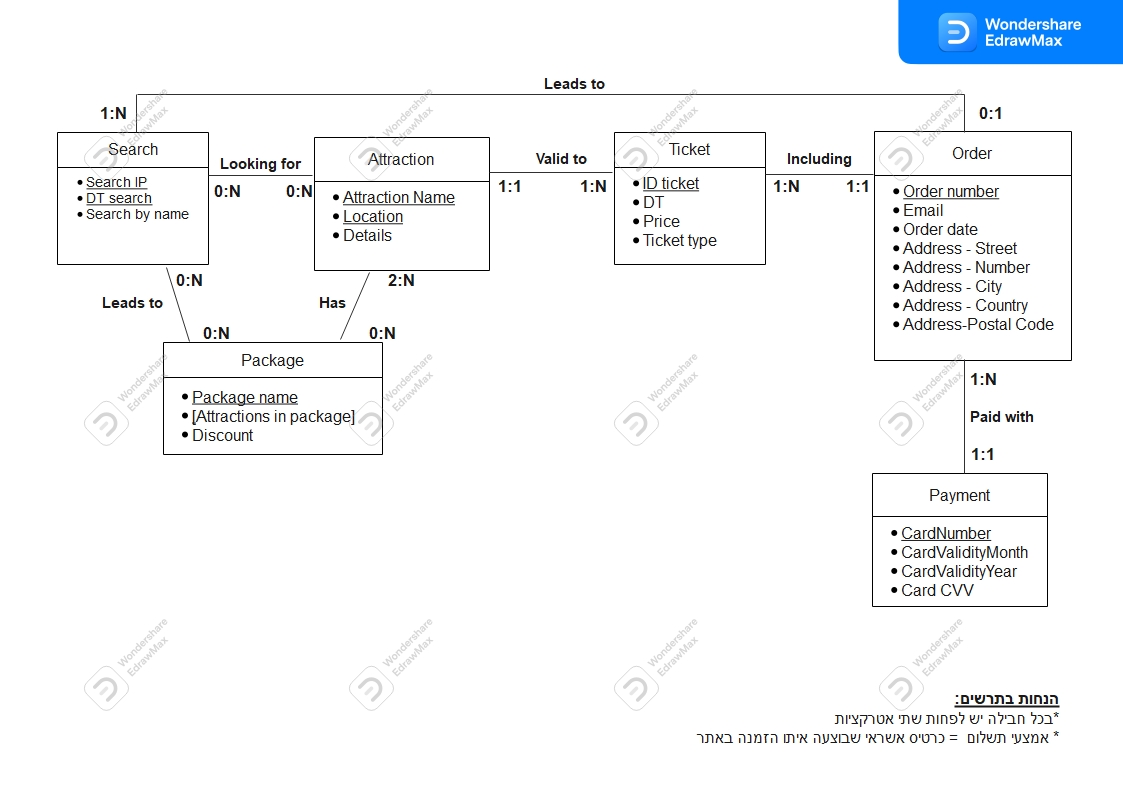
חברת Attraction Tickets עובדת בצורה מקוונת בלבד והיתרון המרכזי שנובע מכך הוא שהאתר נגיש לפלח שוק עולמי. מטרתה העיקרית של החברה היא למכור כרטיסים במחיר הזול בעולם, לכן העובדה שאינה מוציאה כסף על משרדים, עובדים ועמדות פיזיות ברחבי העולם חוסכת הוצאות רבות. זמינות האתר 24/7 מעודדת מכירות פאסיביות, ניתן להזמין גם כאשר העובדים בחופש. יתרון נוסף וחשוב מאוד לחברה הוא יכולת שמירת נתוני הגלישה של המשתמשים ובאמצעות מידע זה להפיק מסקנות עסקיות שיעזרו לחברה לקדם את המכירות. חיסרון בולט בעבודה באופן מקוון הוא תקלות טכניות באתר וכן, טעויות אנוש בעת ביצוע הזמנה. ללקוח גם יתרונות מדרך פעולה זו ,האתר משרת משתמשים בכל רחבי העולם, זמין בכמה שפות ובכך יכול לשרת אוכלוסיות רבות ומגוונות. האתר מאפשר לנוסע לטוס בראש שקט בכך שיש לו כרטיסים מוזמנים מראש. משתמש האתר יכול להתלבט ולשהות כמה זמן שהוא ירצה ללא הגבלה בתור פיזי בכניסה לאתרים. כתוצאה מחיסכון עלויות החברה גם הלקוח נהנה ממחירי כרטיסים זולים יותר. מצד השני אופן פעולה זה מקשה על אוכלוסיות ספציפיות, למשל לקהל יעד מבוגר שלא מסתדר עם הטכנולוגיה של רכישות באינטרנט. בנוסף לזה כל משתמש יכול לטעות בהזמנה שלו בבחירת תאריך או מיקום. כמו כן הלקוחות יעדיפו שירות אישי.

\* -חובה \*\* -אופציונלי

****

**חלק 2**

## מטלה 1 (40%) – עיצוב קונספטואלי (תרשים ישויות-קשרים)



## הנחות מודל ה- ERD

|  |  |
| --- | --- |
| **מרכיב** | **הנחות** |
| Entity: Ticket   * ID ticket * DT * Price * Ticket type | הלקוח מקבל כרטיסים רשמיים לאטרקציה שאיתם יכול להיכנס ישירות למקום מבלי להמיר את הכרטיס בכניסה. |
| Ticket type שדה | סוג הכרטיס לדוגמא standard, flex ,flex-plus |
| Entity: Order   * Email * OrderNumber * Order date * Address - Street * Address - Number * Address - City * Address - Country * Address - Country * Postal Code | 1. כדי להיכנס לפרטי הזמנה ספציפית יש להקליד את כתובת המייל ואת מספר ההזמנה. האתר לא מזהה את הלקוח כלקוח קבוע, ולא שומר את פרטיו ואת היסטוריית הזמנותיו. 2. Address הוא שדה מורכב, המכיל את כל פרטי כתובת הלקוח. |
| הקשר בין "Ticket"  ל""Order | ההזמנה מכילה את כלל הכרטיסים לאטרקציות וכל כרטיס משויך להזמנה. |
| אמצעי תשלום | ההזמנה מתבצעת רק ע"י אמצעי תשלום אחד. |
| מזהים ב Attraction - | כדי למנוע מצב של חוסר הבחנה בין שתי אטרקציות בעלות אותו שם (למשל פארקים של דיסני) נזהה את Attraction Name, Location האטרקציה ע"י שני שדות: |
| הקשר בין " Search "  ל""Order | הזמנה יכולה להכיל מספר אטרקציות וחבילות שונות אשר נבחרו בעזרת חיפוש אחד או מספר חיפושים ואילו חיפוש אחד יכול להוביל ל-0-1 הזמנות ולכן הקרדינאליות בין " Search " ל" Order " הוגדרה כ- 0:1 ו-N:1 |

## מטלה 2 (30%) – עיצוב לוגי (מודל נתונים טבלאי)

**SEARCHES** ( Search IP,DT search , Search by name, OrderNumber **(ORDERS)**)

**RESULTS ATTRACTIONS ({**Search IP,DT search} **(SEARCHES), {** Attraction Name , Location}(**ATTRACTIONS))**

**RESULTS PACKAGES ({**Search IP,DT search} **(SEARCHES),** Package name**(PACKAGES))**

**PACKAGES** (Package name, Discount)

**DEALS (**Package name**(PACKAGES), {**Attraction Name , Location} (**ATTRACTIONS)** **)**

**ATTRACTIONS** (Attraction Name , Location ,Details )

**TICKETS** ( ID Ticket ,DT ,Price ,Ticket type, **{**Attraction Name , Location} (**ATTRACTIONS),** OrderNumber **(ORDERS)**)

**ORDERS** (OrderNumber, email, Address - Order date, Address – Street, Address - Number, Address - City, Address - Country, Address - Postal Code, {CardNumber (**PAYMENTS**) )

**PAYMENTS** ( CardNumber , Validity – month, Validity – year , cardCVV)

# פירוט שדות בסיס הנתונים והערכת גודלם

|  |  |  |  |
| --- | --- | --- | --- |
| **Table** | **Field** | **Data Type** | **Size (Bytes)** |
| **SEARCHES** | Search IP | Varchar(50) | 50 |
| DT search | smalldatetime | 4 |
| Search by name | Varchar(30) | 30 |
| OrderNumber  **(ORDERS)** | Varchar(20) | 20 |
| **Total Record Size (Bytes)** | | **104** |
| **RESULTS ATTRACTIONS** | Search IP**(SEARCHES)** | Varchar(50) | 50 |
| DT search**(SEARCHES)** | smalldatetime | 4 |
| Attraction Name  (**ATTRACTIONS)** | Varchar(50) | 50 |
| Location (**ATTRACTIONS)** | Varchar(50) | 50 |
| **Total Record Size (Bytes)** | | **154** |
| **RESULTS PACKAGES** | Search IP **(SEARCHES)** | Varchar(50) | 50 |
| DT search **(SEARCHES)** | smalldatetime | 4 |
| Package name  **(PACKAGES)** | Varchar(100) | 100 |
| **Total Record Size (Bytes)** | | **154** |
| **PACKAGES** | Package name | Varchar(100) | 100 |
| Discount (%) | tinyInt | 1 |
| **Total Record Size (Bytes)** | | **101** |
| **DEALS** | Package name  **(PACKAGES)** | Varchar(100) | 100 |
| Attraction Name  (**ATTRACTIONS)** | Varchar(50) | 50 |
| Location (**ATTRACTIONS)** | Varchar(50) | 50 |
| **Total Record Size (Bytes)** | | **200** |
| **ATTRACTIONS** | Attraction Name | Varchar(50) | 50 |
| Location | Varchar(50) | 50 |
| Details | Varchar(100) | 100 |
| **Total Record Size (Bytes)** | | **200** |
| **TICKETS** | ID Ticket | Varchar(20) | 4 |
| DT | Date | 3 |
| Price | Money | 8 |
| Ticket type | Varchar(15) | 15 |
| Attraction Name  (**ATTRACTIONS)** | Varchar(50) | 20 |
| Location (**ATTRACTIONS)** | Varchar(50) | 20 |
| OrderNumber  **(ORDERS)** | Varchar(20) | 4 |
| **Total Record Size (Bytes)** | | **74** |
| **ORDERS** | OrderNumber | Varchar(20) | 20 |
| Email | Varchar(30) | 30 |
| Order date | Date | 3 |
| Address - Street | Varchar(10) | 10 |
| Address - Number | int | 4 |
| Address - City | Varchar(10) | 10 |
| Address - Country | Varchar(10) | 10 |
| Address - Postal Code | Varchar(20) | 20 |
| CardNumber  (**PAYMENTS**) | Varchar(12) | 12 |
| **Total Record Size (Bytes)** | | **119** |
| **PAYMENTS** | CardNumber | Varchar(12) | 4 |
| cardValidityMonth | tinyInt | 1 |
| cardValidityYear | int | 4 |
| cardCVV | int | 4 |
| **Total Record Size (Bytes)** | | **13** |

## פירוט הנחות והסברים נוספים לגבי שדות בסיסי הנתונים

|  |  |
| --- | --- |
| **מרכיב** | **הנחות** |
| שדה Discount ב **PACKAGES** | ההנחה היא באחוזים ולכן נשתמש בData type- מסוג tinyData. |
| שדה cardCVV ב **PAYMENTS** | בחלק מכרטיסי האשראי הcvv- כולל 4 ספרות ולכן השתמשנו בint ולא ב tinyInt |

## מטלה 3 (30%) - עיצוב פיסי (בסיס נתונים פעיל על שרת RDBMS)

**הזנת נתונים לבדיקה:**

**אילוצי בדיקה (CHECK)**

|  |  |
| --- | --- |
| constraint ck\_discount check ( discount>= 0 and discount<=100)  האילוץ בודק שההנחה היא בין 0 אחוז ל100 אחוז הנחה | |
| constraint ck\_cardValidityMonth check (cardValidityMonth>= 1 and cardValidityMonth<=12)  האילוץ בודק שהחודש הוא בין 1 ל12 | |
| constraint ck\_cardValidityYear check ( cardValidityYear >= year(getdate()))  האילוץ בודק שהכרטיס בתוקף | |
| constraint ck\_email check (email like '%@%.%' )  האילוץ בודק שמבנה כתובת האימייל תקין | |
| constraint ck\_price check ( price > 0 )  האילוץ בודק שהמחירים גדולים מ-0  **טבלאות חיפוש (Lookup Tables):** |

CREATE TABLE LOCATIONS(

Location\_ varchar(50) Not null,

constraint pk\_country primary Key (Location\_),

)

Insert Into LOCATIONS Values

('London UK'),

('Paris France'),

('Merritt Island, FL , US'),

('Orlando , FL, US')

alter table ATTRACTIONS

ADD CONSTRAINT fk\_locations foreign key (Location\_) references LOCATIONS(Location\_)

**הזנת נתונים לבדיקה – הקוד המלא**

DROP table IF EXISTS dbo.

RESULTS\_ATTRACTIONS

DROP table IF EXISTS dbo.

RESULTS\_PACKAGES

DROP TABLE IF EXISTS dbo.

SEARCHES

DROP table IF EXISTS dbo.

DEALS

DROP TABLE IF EXISTS dbo.

PACKAGES

DROP TABLE IF EXISTS dbo.

TICKETS

DROP TABLE IF EXISTS dbo.

ATTRACTIONS

DROP TABLE ORDERS

DROP TABLE PAYMENTS

DROP TABLE IF EXISTS dbo.LOCATIONS

CREATE TABLE PACKAGES (

packageName varchar(100) primary key not null ,

discount tinyint,

constraint ck\_discount check ( discount>= 0 and discount<=100)

)

CREATE TABLE PAYMENTS (

cardNumber varchar(12) not null ,

cardValidityMonth tinyint not null ,

cardValidityYear int not null,

cardCVV int

primary key (cardNumber),

constraint ck\_cardValidityMonth check (cardValidityMonth>= 1 and cardValidityMonth<=12) ,

constraint ck\_cardValidityYear check ( cardValidityYear >= year(getdate())) ,

)

CREATE TABLE ORDERS (

orderNumber varchar(20) not null ,

email varchar(30),

orderDT DATE,

[address-Street] varchar(10) ,

[address-Number] int ,

[address-City] varchar(10) ,

[address-Country] varchar(10),

[address-PostalCode] varchar(20) ,

cardNumber varchar(12) ,

primary key (orderNumber),

FOREIGN KEY (cardNumber) REFERENCES PAYMENTS(cardNumber),

constraint ck\_email check (email like '%@%.%' ) ,

)

CREATE TABLE ATTRACTIONS (

AttractionName varchar(50) not null ,

Location\_ varchar(50) not null ,

Details varchar(100) ,

primary key ( AttractionName, Location\_),

)

create table TICKETS (

IDTicket varchar(20) not null ,

DT DATE ,

price money ,

ticketType varchar(15) ,

AttractionName varchar(50) not null ,

Location\_ varchar(50) not null ,

orderNumber varchar(20) not null ,

primary key (IDTicket),

foreign key (orderNumber) references ORDERS(orderNumber) ,

foreign key (AttractionName,Location\_) references ATTRACTIONS(AttractionName, Location\_) ,

constraint ck\_price check ( price > 0 ) ,

)

create table SEARCHES (

SearchIP varchar(50)not null ,

DTSearch smallDateTime not null ,

SearchByName varchar(30) ,

orderNumber varchar(20) not null ,

primary key (SearchIP,DTSearch) ,

foreign key (orderNumber) references ORDERS(orderNumber)

)

CREATE TABLE DEALS (

PackageName varchar(100) not null ,

AttractionName varchar(50) not null ,

Location\_ varchar(50) not null ,

primary key (PackageName, AttractionName, Location\_) ,

foreign key (PackageName) references PACKAGES(PackageName) ,

foreign key (AttractionName,Location\_) references ATTRACTIONS(AttractionName,Location\_)

)

create table RESULTS\_ATTRACTIONS(

SearchIP varchar(50)not null ,

DTSearch smallDateTime not null ,

AttractionName varchar(50) not null ,

Location\_ varchar(50) not null ,

PRIMARY KEY (SearchIP,DTSearch, AttractionName,Location\_) ,

foreign key (AttractionName,Location\_) references ATTRACTIONS(AttractionName,Location\_) ,

foreign key (SearchIP,DTSearch) references SEARCHES(SearchIP,DTSearch) ,

)

create table RESULTS\_PACKAGES(

SearchIP varchar(50)not null ,

DTSearch smallDateTime not null ,

PackageName varchar(100) not null ,

PRIMARY KEY (SearchIP,DTSearch,PackageName) ,

foreign key (PackageName) references PACKAGES(PackageName) ,

foreign key (SearchIP,DTSearch) references SEARCHES(SearchIP,DTSearch) ,

)

INSERT INTO PAYMENTS (cardNumber, cardValidityMonth, cardValidityYear, cardCVV) VALUES

('458012314589','10','2025','1032'),

('458096512547','11','2022','111'),

('458051234125','12','2030','234'),

('789456123458','1','2026','430'),

('741258963123','2','2028','667'),

('850330901963','3','2025','343'),

('940116192761','5','2026','543'),

('102990148355' ,'4','2027','445'),

('111968677435','7','2028','567'),

('120947206515','8','2029','867')

INSERT INTO PACKAGES (PackageName, Discount) VALUES

('Disney 14-Day Magic Ticket','5'),

('Disneyland California 2 Day Ticket (One Park Per Day)','7'),

('Universal Orlando 3 Park Explorer Ticket','7'),

('Disney and Universal Combo Ticket','10'),

('Discovery Cove® Package','12'),

('Madame Tussauds Orlando & SEA LIFE Combo Ticket','20'),

('The lastminute.com London Eye Admission plus River Thames Cruise','19'),

('PortAventura & Ferrari Land 2 Day Ticket','15')

INSERT INTO ATTRACTIONS (AttractionName, Location\_, Details) VALUES

('London Eye','London UK','" nothing says iconic like the lastminute.com London Eye."'),

('Disneyland, Paris 1-Day/1 Park Dated Ticket','Paris France','Disneyland® Park: From the moment your foot (or glass slipper) hits the cobbles of Main Street.'),

('Kennedy Space Center Tickets','Merritt Island, FL , US','Kennedy Space Centre tickets provide a hugely entertaining and educational experience'),

('SEA LIFE Orlando','Orlando , FL, US','Be one of the first to explore the fantastic SEA LIFE Orlando Aquarium and come face to face') ,

('Madame Tussauds Orlando','Orlando , FL, US','Madame Tussauds Orlando is rolling out the red carpet to become Florida’s latest celebrity hotspot!'),

('Disney World','Orlando , FL, US','Make your Disney Theme Park visits easier and more fun with the magical Disney Genie Service'),

('Universal Studios Florida','Orlando , FL, US','Universal Studios Florida (also known as Universal Studios or USF'),

('Universals Islands of Adventure','Orlando , FL, US','Universals Islands of Adventure (also known as Islands of Adventure or IOA'),

('Universals Volcano Bay', 'Orlando , FL, US', 'Water is Life. Life is Joy.')

INSERT INTO DEALS (PackageName , AttractionName, Location\_) VALUES

('Madame Tussauds Orlando & SEA LIFE Combo Ticket','Madame Tussauds Orlando','Orlando , FL, US'),

('Madame Tussauds Orlando & SEA LIFE Combo Ticket','SEA LIFE Orlando','Orlando , FL, US'),

('Disney and Universal Combo Ticket', 'Disney World','Orlando , FL, US'),

('Disney and Universal Combo Ticket','Universal Studios Florida','Orlando , FL, US'),

('Universal Orlando 3 Park Explorer Ticket','Universal Studios Florida','Orlando , FL, US'),

('Universal Orlando 3 Park Explorer Ticket','Universals Islands of Adventure','Orlando , FL, US'),

('Universal Orlando 3 Park Explorer Ticket','Universals Volcano Bay','Orlando , FL, US')

INSERT INTO ORDERS (OrderNumber, email ,orderDT, [address-Street], [address-Number], [address-City], [address-Country], [address-PostalCode], CardNumber) VALUES

('3030303455','first@gmail.com', '2021-12-20','Frishman','3','Tel-Aviv','Tel-Aviv','435336','458012314589'),

('3030303456','second@gmail.com','2021-04-19','Pizza','10','Rome','Italy','8665556','458096512547'),

('3030303457','third@gmail.com', '2022-11-02','Olive','1','Tokyo','Japan','dc4456','458051234125'),

('3030303458','fourth@gmail.com','2023-08-09','Couwe','207','Hong-Kong','Hong-Kong','hk3579','789456123458'),

('3030303459','fifth@gmail.com' ,'2023-06-10','Wow','12','New -York','US','wow8080','741258963123'),

('303030660','sixth@gmail.com' ,'2020-12-22','Dragon','111','Tel-Aviv','Israel','16895776' ,'850330901963'),

('303030661','seventh@gmail.com','2021-09-27','Doney','132','Rome' ,'Italy','25125996' ,'940116192761'),

('303030662','eight@gmail.com','2022-11-10','Silver','154','Athens','Greece','dc4457','102990148355'),

('303030663','ninth@gmail.com','2023-02-12','Tapi','175','Hong-Kong','Hong-Kong','hk3580','111968677435'),

('303030664','tenth@gmail.com','2023-04-10','Flower','197', 'New -York' ,'US','wow8081','120947206515')

INSERT INTO TICKETS (IDTicket, DT, Price, ticketType, AttractionName, Location\_, OrderNumber) VALUES

('123777', '2021-12-20','36','Adult','London Eye', 'London UK', '3030303455'),

('303030' ,'2021-05-21','48','Adult','Disneyland, Paris 1-Day/1 Park Dated Ticket','Paris France','3030303456'),

('555555','2022-12-12','36', 'Child','Kennedy Space Center Tickets','Merritt Island, FL , US','3030303457'),

('488', '2023-08-14','25','Adult','SEA LIFE Orlando','Orlando , FL, US','3030303458'),

('34647', '2023-08-16' ,'25', 'Adult','Madame Tussauds Orlando','Orlando , FL, US', '3030303459'),

('24471', '2021-12-22' , '28', 'Adult', 'Universal Studios Florida','Orlando , FL, US', '303030660'),

('29367', '2022-05-23', '32', 'Adult', 'Universals Islands of Adventure','Orlando , FL, US', '303030661'),

('342633', '2022-11-12', '20', 'Child', 'Universals Volcano Bay','Orlando , FL, US', '303030662'),

('391', '2023-09-14', '35', 'Adult', 'Universals Volcano Bay','Orlando , FL, US', '303030663'),

('44054', '2023-08-17', '35', 'Adult', 'Universals Volcano Bay','Orlando , FL, US', '303030664')

INSERT INTO SEARCHES ( SearchIP, DTsearch, SearchByName, OrderNumber) VALUES

('35666666666HHM', '2020-01-10 22:22:22', 'London Eye', '3030303455'),

('10101010ASXCV', '2020-01-12 22:22:23', 'Disneyland', '3030303456'),

('300000000NCL', '2020-01-01 22:22:24', 'Kennedy Space Center', '3030303457'),

('346VBBBRRE555', '2020-06-01 22:22:25', 'Madame Tussauds Orlando', '3030303458'),

('12343111AZXC56' ,'2020-01-11 22:22:26', 'Madame Tussauds Orlando', '3030303459'),

('35454566666AHM', '2020-05-11 21:22:23' ,'Disney Theme Park', '303030660'),

('212121211XX', '2020-01-01 22:09:23', 'Disney 14-Day', '303030661'),

('67676767D', '2020-01-11 06:22:23', 'Disney', '303030662'),

('RERERR45', '2020-01-11 22:00:23' , '3 park', '303030663')

INSERT INTO RESULTS\_ATTRACTIONS (SearchIP, DTsearch, AttractionName, Location\_) VALUES

('35666666666HHM', '2020-01-10 22:22:22', 'London Eye','London UK'),

('10101010ASXCV', '2020-01-12 22:22:23', 'Disneyland, Paris 1-Day/1 Park Dated Ticket','Paris France'),

('300000000NCL', '2020-01-01 22:22:24', 'Kennedy Space Center Tickets','Merritt Island, FL , US'),

('346VBBBRRE555', '2020-06-01 22:22:25' , 'Madame Tussauds Orlando','Orlando , FL, US'),

('12343111AZXC56' ,'2020-01-11 22:22:26', 'Madame Tussauds Orlando','Orlando , FL, US'),

('35454566666AHM', '2020-05-11 21:22:23', 'Disney World','Orlando , FL, US')

INSERT INTO RESULTS\_PACKAGES(SearchIP, DTsearch, PackageName) VALUES

('212121211XX', '2020-01-01 22:09:23' ,'Disney 14-Day Magic Ticket'),

('67676767D', '2020-01-11 06:22:23', 'Disneyland California 2 Day Ticket (One Park Per Day)'),

('RERERR45', '2020-01-11 22:00:23','Universal Orlando 3 Park Explorer Ticket'),

('346VBBBRRE555', '2020-06-01 22:22:25','Madame Tussauds Orlando & SEA LIFE Combo Ticket')

CREATE TABLE LOCATIONS(

Location\_ varchar(50) Not null,

constraint pk\_country primary Key (Location\_),

)

Insert Into LOCATIONS Values

('London UK'),

('Paris France'),

('Merritt Island, FL , US'),

('Orlando , FL, US')

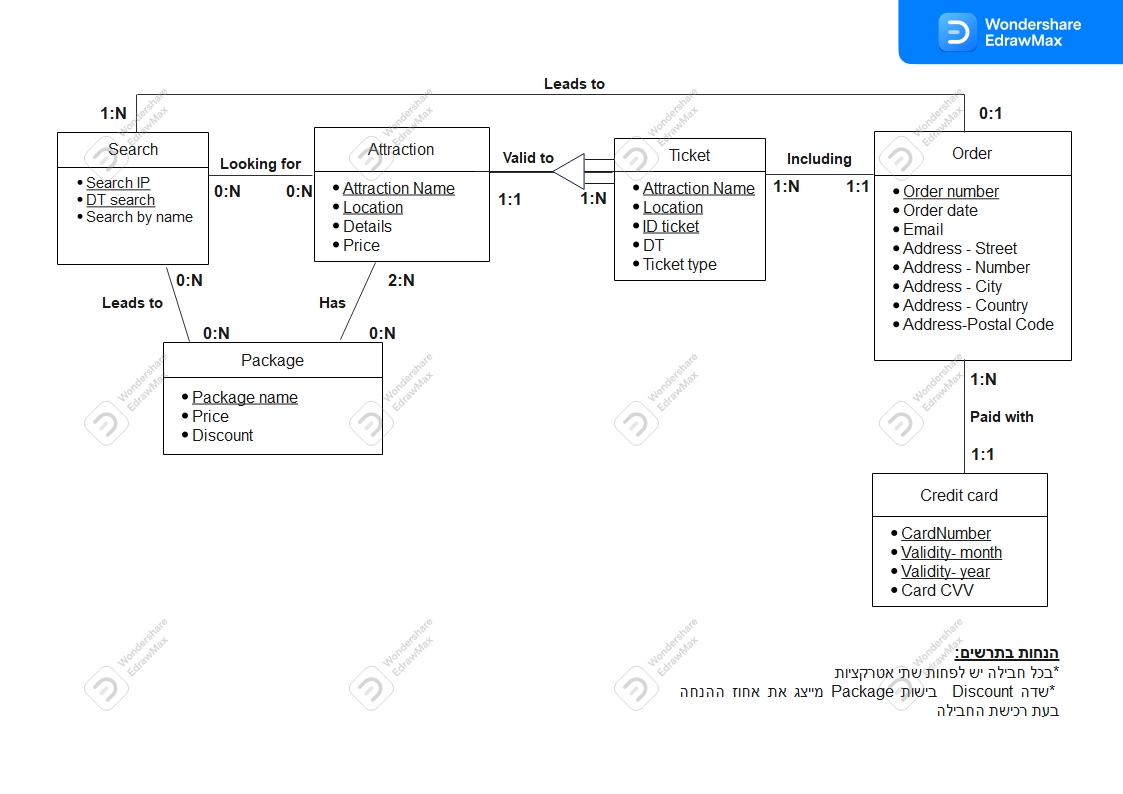
alter table ATTRACTIONS

ADD CONSTRAINT fk\_locations foreign key (Location\_) references LOCATIONS(Location\_)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| מס' קבוצה | אתר | | | תאריך הגשה |
| 60 | [www.attractiontickets.com](http://www.attractiontickets.com) | | | 25/06/2022 |
| חברי הצוות - מספרי ת.ז | | | | |
| 316320779 | | 207941105 | 315673186 | |

## פרויקט בסיסי נתונים – חלק ג'

**ERDמטלה מקדימה –מתוקן לחלק ג'**



עיקרי התיקונים

* העברנו את שדה price להיות תחת ישויות החבילה והאטרקציה.
* הפכנו את הכרטיסים לישות חלשה של האטרקציות.

## מטלה 2 (30%) – עיצוב לוגי (מודל נתונים טבלאי)

**SEARCHES** ( Search IP,DT search , Search by name, OrderNumber **(ORDERS)**)

**RESULTS ATTRACTIONS ({**Search IP,DT search} **(SEARCHES), {**Attraction Name , Location}(**ATTRACTIONS))**

**RESULTS PACKAGES ({**Search IP,DT search} **(SEARCHES),** Package name**(PACKAGES))**

**PACKAGES** (Package name, Price, Discount)

**DEALS (**Package name**(PACKAGES), {**Attraction Name , Location} (**ATTRACTIONS)** **)**

**ATTRACTIONS** (Attraction Name , Location ,Details, Price )

**TICKETS** (**{**Attraction Name , Location} (**ATTRACTIONS)**, ID Ticket , DT ,Ticket type**,** OrderNumber **(ORDERS)**)

**ORDERS** (OrderNumber, Order date, email, Address – Street, Address - Number, Address - City, Address - Country, Address - Postal Code,

{CardNumber, Validity – month, Validity – year}(**PAYMENTS**) )

**CREDITCARDS** ( CardNumber , Validity – month, Validity – year , cardCVV)

# פירוט שדות בסיס הנתונים והערכת גודלם

|  |  |  |  |
| --- | --- | --- | --- |
| **Table** | **Field** | **Data Type** | **Size (Bytes)** |
| **SEARCHES** | Search IP | Varchar(15) | 15 |
| DT search | smalldatetime | 4 |
| Search by name | Varchar(50) | 50 |
| OrderNumber  **(ORDERS)** | Varchar(20) | 20 |
| **Total Record Size (Bytes)** | | **89** |
| **RESULTS ATTRACTIONS** | Search IP**(SEARCHES)** | Varchar(50) | 50 |
| DT search**(SEARCHES)** | smalldatetime | 4 |
| Attraction Name  (**ATTRACTIONS)** | Varchar(50) | 50 |
| Location (**ATTRACTIONS)** | Varchar(50) | 50 |
| **Total Record Size (Bytes)** | | **154** |
| **RESULTS PACKAGES** | Search IP **(SEARCHES)** | Varchar(50) | 50 |
| DT search **(SEARCHES)** | smalldatetime | 4 |
| Package name  **(PACKAGES)** | Varchar(100) | 100 |
| **Total Record Size (Bytes)** | | **154** |
| **PACKAGES** | Package name | Varchar(100) | 100 |
| Price | Money | 8 |
| Discount (%) | Decimal(32) | 5 |
| **Total Record Size (Bytes)** | | **113** |
| **DEALS** | Package name  **(PACKAGES)** | Varchar(100) | 100 |
| Attraction Name  (**ATTRACTIONS)** | Varchar(50) | 50 |
| Location (**ATTRACTIONS)** | Varchar(50) | 50 |
| **Total Record Size (Bytes)** | | **200** |
| **ATTRACTIONS** | Attraction Name | Varchar(50) | 50 |
| Location | Varchar(50) | 50 |
| Details | Varchar(100) | 100 |
| Price | Money | 8 |
| **Total Record Size (Bytes)** | | **208** |
| **TICKETS** | ID Ticket | Varchar(20) | 4 |
| DT | Date | 3 |
| Ticket type | Varchar(15) | 15 |
| Attraction Name  (**ATTRACTIONS)** | Varchar(50) | 20 |
| Location (**ATTRACTIONS)** | Varchar(50) | 20 |
| OrderNumber  **(ORDERS)** | Varchar(20) | 4 |
| **Total Record Size (Bytes)** | | **66** |
| **ORDERS** | OrderNumber | Varchar(20) | 20 |
| Email | Varchar(30) | 30 |
| Order date | Date | 3 |
| Address - Street | Varchar(10) | 10 |
| Address - Number | int | 4 |
| Address - City | Varchar(10) | 10 |
| Address - Country | Varchar(10) | 10 |
| Address - Postal Code | Varchar(20) | 20 |
| CardNumber  (**PAYMENTS**) | Varchar(12) | 12 |
| **Total Record Size (Bytes)** | | **119** |
| **CREDITCARDS** | CardNumber | Varchar(12) | 4 |
| cardValidityMonth | tinyInt | 1 |
| cardValidityYear | int | 4 |
| cardCVV | Varchar(4) | 4 |
| **Total Record Size (Bytes)** | | **13** |

## פירוט הנחות והסברים נוספים לגבי שדות בסיסי הנתונים

|  |  |
| --- | --- |
| **מרכיב** | **הנחות** |
| שדה Discount ב **PACKAGES** | ההנחה היא באחוזים ולכן נשתמש בData type- מסוג tinyData. |
| שדה cardCVV ב **PAYMENTS** | בחלק מכרטיסי האשראי הcvv- כולל 4 ספרות ולכן השתמשנו בint ולא ב tinyInt |

## מטלה 1 (25%) – שאילתות

# שתי שאילתות SELECT ללא קינון (5%)

**שאילתה ללא קינון ראשונה:**

הרעיון העסקי:

לאחר שנים של שיתוף פעולה עם חברת אמריקן אקספרס (CCV מכיל 4 ספרות), "Attraction tickets" החליטה לשתף פעולה עם שאר חברות האשראי (CCV מכיל 3 ספרות) והחליטה לשלוח ללקוחות שחיפשו דילים בבלאק פריידיי (26.11.2021) וביצעו הזמנה, שוברי הנחה להזמנה הבאה במייל.

הפלט: יציג את כתובת המייל של הלקוח, תאריך החיפוש (בלאק פריידיי) ואת מספר ההזמנות שביצע.

SELECT O.email, [Date] = CAST (S.DTSearch as varchar), [Total Orders] = COUNT (\*)

FROM CREDITCARDS AS CC JOIN ORDERS AS O ON CC.cardNumber = O.cardNumber

JOIN SEARCHES AS S ON s.orderNumber= O.orderNumber

WHERE LEN(CC.cardCVV)= 3 AND S.DTSearch LIKE '%NOV 26 2021%'

GROUP BY O.email, S.DTSearch

HAVING COUNT (\*) > 0

ORDER BY O.email ASC

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי

**שאילתה ללא קינון שניה:**

הרעיון העסקי: חברת יוניברסל סטודיוס הפסידה בתביעה ותיאלץ לפצות ב10$ את לקוחותיה שביצעו הזמנות לשנת 2022, הכסף יוחזר לחשבונם באמצעות כרטיס האשראי איתו ביצעו את ההזמנה.

פלט: מס כרטיס אשראי, תאריך הזמנה ושם הפארק שהוזמן.

SELECT O.cardNumber, T.DT , T.AttractionName

FROM ORDERS AS O JOIN TICKETS AS T ON O.orderNumber = T.orderNumber

JOIN CREDITCARDS AS CC ON CC.cardNumber = O.cardNumber

WHERE T.DT LIKE '%2022%' AND T.AttractionName LIKE '%Universal%'

ORDER BY T.DT ASC

תמונה שמכילה טקסט, שולחן

התיאור נוצר באופן אוטומטי

# שתי שאילתות SELECT מקוננות (10%)

**שאילתה עם קינון ראשונה:**

הרעיון העסקי: בעקבות מגפת הקורונה הקשה ביפן בשנת 2022, המדינה החליטה על סגר אווירי. "Attraction tickets" רוצה להבין מה מספר ההזמנות ומהו אחוז הלקוחות שלא יוכלו לממש את הזמנותיהם השנה בעקבות הסגר האווירי.

פלט: ערך סקלארי - אחוז הלקוחות שביצעו הזמנות לשנת 2022 וגרים ביפן

select

[percentage of total orders 2022] =cast( [number of orders] as float)\*100/(select sum([number of orders])

from (select [address-Country], [number of orders] = count (o.orderNumber)

from orders as o join TICKETS as T on o.orderNumber = t.orderNumber

WHERE T.DT LIKE '%2022%'

group by [address-Country]) as [orders by country 2022])

from ( select [address-Country], [number of orders] = count (o.orderNumber)

from orders as o join TICKETS as T on o.orderNumber = t.orderNumber

WHERE T.DT LIKE '%2022%' and [address-Country] = 'japan'

group by [address-Country])

as tot\_amount



**שאילתה עם קינון שניה:**

הרעיון העסקי: חברת "Attraction tickets" שוקלת לתמחר את האטרקציות לפי ביקוש הזמנות ע"פ עונות השנה, לפיכך תפיק דוח הזמנות והיקף מכירות לפי עונות.

פלט: מספר הזמנות לפי עונה והיקף הכנסות.

select order\_seassons.seasson,[total orders per seasson]=count(\*),[total sales] = sum(a.price)

from (

select \*,seasson = (case

when month(o.orderDT) between 3 and 5 then 'spring'

when month(o.orderDT) between 6 and 8 then 'summer'

when month(o.orderDT) between 9 and 11 then 'fall'

else 'winter' end)

from orders as o

) as order\_seassons join TICKETS as t on order\_seassons.orderNumber = t.orderNumber join ATTRACTIONS as a on a.AttractionName = t.AttractionName

group by seasson

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

# שתי שאילתות מקוננות תוך שימוש במרכיבים נוספים (10%)

* **שאילתה 1:** יצירת שאילתה אחת של Update או Delete מקוננת.

רעיון עסקי: פעם בעשור מתקיימת ישיבת סיכום שבה מעריכים את האטרקציות ומחליטים איזה אטרקציות משתלם להציג בעמוד הבית של האתר, לשם כך נוסיף שתי עמודות – מס הזמנות לפי אטרקציה וסך הכנסות לפי אטרקציה ודירוג כוכבים. האטרקציות המובילות יוצגו בעמוד הבית של האתר.

פלט: פרטי האטרקציה, המכירות ודירוג הכוכבים.

alter table ATTRACTIONS drop column [Total sales]

alter table ATTRACTIONS drop column [Total orders]

alter table ATTRACTIONS drop column [stars rank of the year]

alter table ATTRACTIONS add [Total orders] int

alter table ATTRACTIONS add [Total sales] int

alter table ATTRACTIONS add [star rank] varchar(10)

update ATTRACTIONS

set [Total orders] =(

select COUNT(\*)

from TICKETS as t

where ATTRACTIONS.AttractionName = t.AttractionName

),

[Total sales] = (price\* [Total orders]),

[star rank] = CASE

When [Total orders] > 20 THEN '5 stars'

When [Total orders] <5 THEN '3 stars'

Else '4 stars'

end

-----------------------

--check

select \*

from ATTRACTIONS

order by [Total sales] desc

(מתוך 58 רשומות)

* **שאילתה 2:** יצירת איחוד/חיתוך/הפרש בין 2 שאילתות, כך שלפחות אחת מהן מקוננת.

רעיון עסקי: "Attraction tickets" רוצה לבדוק איזה אטרקציות לקוחות ממדינות מסוימות לא רוכשים ולאחר מכן לעשות קמפיין עידוד הזמנות מותאם לכל מדינה. לשם כך נחסר את כל האטרקציות שהוזמנו ע"י לקוחות מכלל המדינות מכל האופציות האפשריות להזמנה.

פלט: מדינות והאטרקציות שלא הוזמנו מלקוחות שגרים בהן.

select \*

from (

SELECT DISTINCT A.AttractionName,[Customer's Country] = O.[address-Country]

FROM ATTRACTIONS AS A CROSS JOIN ORDERS AS O

EXCEPT

SELECT DISTINCT T.AttractionName, [Customer's Country] = O.[address-Country]

FROM ORDERS AS O JOIN TICKETS AS T ON O.orderNumber= T.orderNumber

) as [AttractionNotVisited\_byCountry]

WHERE AttractionNotVisited\_byCountry.[Customer's Country] = 'china'

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

מטלה 2 (25%) – יישומי כלים מתקדמים

# View (5%)

הרעיון העסקי: ניצור view המציג את טבלת ההזמנות שלנו ללא עמודות פרטי כרטיס האשראי של הלקוח. המטרה שלנו להציג את הטבלה של ההזמנות לכלל עובדי המכירות בחברה שלא יכולים להיחשף לפרטי האשראי של הלקוחות מטעמי אבטחת מידע.

create view orders\_forMarketing as (

select orderNumber ,o.orderDT, o.email , o.[address-Street],o.[address-Number],o.[address-City],o.[address-Country], o.[address-PostalCode]

from orders as o)

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי(מדגם)

# פונקציות (Functions) (10%)

**פונקציה ראשונה:**

הרעיון העסקי: לבדוק ביקוש לאטרקציה או חבילה מסוימת ע"פ מספר הפעמים שחופשה באתר.

פלט: מקבל שם ובודק כמה פעמים חיפשו אותו באתר.

create function f\_num\_of\_searches\_byname (@NAME\_OF\_ATTRACTION varchar(20) )

returns int

as begin

declare @tot int

select @tot= count(\*)

from SEARCHES as s

where s.SearchByName=@NAME\_OF\_ATTRACTION

return @tot

end

---------------------------check

select [Number of searches for attraction] = dbo.f\_num\_of\_searches\_byname ('Disney World')



**פונקציה שנייה:**

הרעיון העסקי: לחברה יש מוקד שירות שנותן שירות לאנשים שמתקשרים אליו. הפונקציה הזו עוזרת לנותן השירות לחפש את המידע הדרוש דרך מספר ההזמנה שהוא ביצע

פלט: הכרטיסים שהלקוח הזמין באותה ההזמנה שמופיעים בהזמנה הספציפית שהלקוח ביצע.

create function search\_order\_details (@orderNumber varchar(20))

returns table

as return

select \*

from TICKETS as t

where t.orderNumber=@orderNumber

--check

select \*

from search\_order\_details('101785681563819')

# 

# Trigger פשוט (5%)

רעיון עסקי: אחד הרעיונות לשיפור השירות שיגרום לעליית המכירות הוא לייצר טבלה חדשה המכילה את כל ההזמנות המבוטלות. המטרה היא להעביר את הפרטים של הלקוח שמבטל את ההזמנה שלו למחלקת שיפור המכירות כך שנציג השירות יוכל להתקשר ללקוח לבדוק סיבת הביטול ולבדוק האם ירצה עזרה להזמנה חדשה.

CREATE TABLE canceled\_orders(

email varchar(30) not null,

[address-Country] varchar(10),

[address-City] varchar(10) ,

[address-Street] varchar(20) ,

[address-Number] int ,

[address-PostalCode] varchar(20) ,

primary key (email),

constraint ck\_emails check (email like '%@%.%' ),

)

----------------------- trigger

DROP Trigger IF EXISTS dbo.

canceling\_orders

create trigger canceling\_orders

on ORDERS

for delete

as

insert into canceled\_orders

select deleted.email,deleted.[address-Country],deleted.[address-City],deleted.[address-Street],deleted.[address-Number],deleted.[address-PostalCode]

from deleted

----------------------------------------

update searches

set orderNumber = null

where searches.orderNumber = '101785681563819'

DELETE FROM TICKETS WHERE orderNumber='101785681563819';

DELETE FROM ORDERS WHERE orderNumber='101785681563819';

----------------------------------------------------------

select\*

from canceled\_orders



# פרוצדורה שמורה ((Stored Procedure פשוטה (5%)

רעיון עסקי: אחת לתקופה מתבצע עדכון מחירים לאטרקציות ע"פ מדד האטרקציות העולמי, נעדכן את מחירי האטרקציות לפי המדד הנ"ל.

פלט: טבלת אטרקציות עם מחירים מעודכנים.

DROP PROCEDURE UPDATE\_Price

CREATE PROCEDURE UPDATE\_Price (@Location\_ varchar(50))

AS

UPDATE ATTRACTIONS

SET price = CASE WHEN

price<50

THEN 1.1\*price

ELSE 1.05\*price

END

FROM ATTRACTIONS AS A JOIN TICKETS AS T ON A.AttractionName=T.AttractionName

WHERE A.Location\_= @Location\_

-------------------------------

execute UPDATE\_Price 'Orlando , FL, US'

SELECT \* FROM ATTRACTIONS

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי(מתוך 58 רשומות)

## מטלה 3 (20%) – כלים להצגת נתונים

הדוח מיועד למחלקת השיווק בחברה, על מנת שיוכלו ללמוד את שוק הלקוחות הקיימים, לפנות לקהלי יעד חדשים ולמדינות שלא חשופות לאתר כיום. בנוסף לשווק לכל קהל יעד את האטרקציות בצורה המתאימה ביותר, להשקיע בפרסום בעונות שפל ולצאת במבצעים לעידוד הזמנות. באמצעות אסטרטגיה שיווקית מותאמת לקוח תוכל החברה להגדיל את רווחיה וכן להיות מהמובילות בתחום.

הדוח מכיל את התפלגות המדינות מהן באים לקוחות החברה, העונות החזקות והחלשות בתחום האטרקציות, התפלגות קהל המבקרים בכל אטרקציה ואת החיפושים שהסתיימו ללא הזמנות.

1. פילוח לקוחות ע"פ מדינות. באמצעות VIEW זה יוכלו לדעת מאיזה מגיעים לקוחות האתר – כך יוכלו להתאים את השיווק לפי מדינות ובנוסף יוכלו לפנות לקהלי יעד חדשים במדינות שלא נחשפו עדיין לאתר.

DROP view IF EXISTS Customers\_by\_countries

create view Customers\_by\_countries as

select [address-Country], [number of orders] = count (o.orderNumber)

from orders as o join TICKETS as T on o.orderNumber = t.orderNumber

group by [address-Country]

1. פילוח הכרטיסים לפי עונות השנה. באמצעות ה-VIEW ניתן לראות מה הן העונות החזקות במהלך השנה, מה ההבדלים בביקורים באטרקציות בין עונות השנה, לתמחר את האטרקציות לפי עונות ולעשות פרסומות מותאמות לכל עונה.

select ticket\_seassons.seasson,[total ticket per seasson]=count(\*)

from (

select \*,seasson = (case

when month(T.DT) between 3 and 5 then 'spring'

when month(T.DT) between 6 and 8 then 'summer'

when month(T.DT) between 9 and 11 then 'fall'

else 'winter' end)

FROM TICKETS AS T

) as ticket\_seassons join TICKETS as t on ticket\_seassons.orderNumber = t.orderNumber join ATTRACTIONS as a on a.AttractionName = t.AttractionName

group by seasson

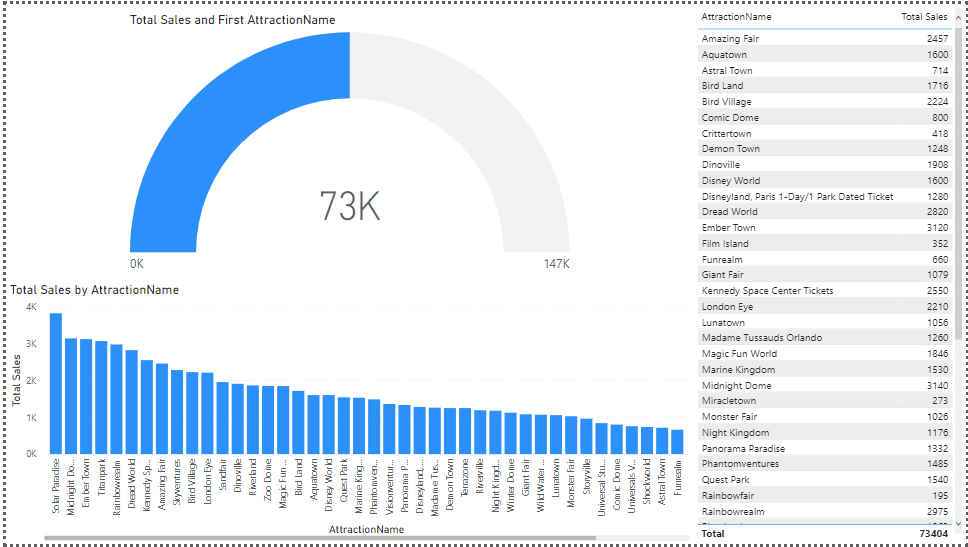
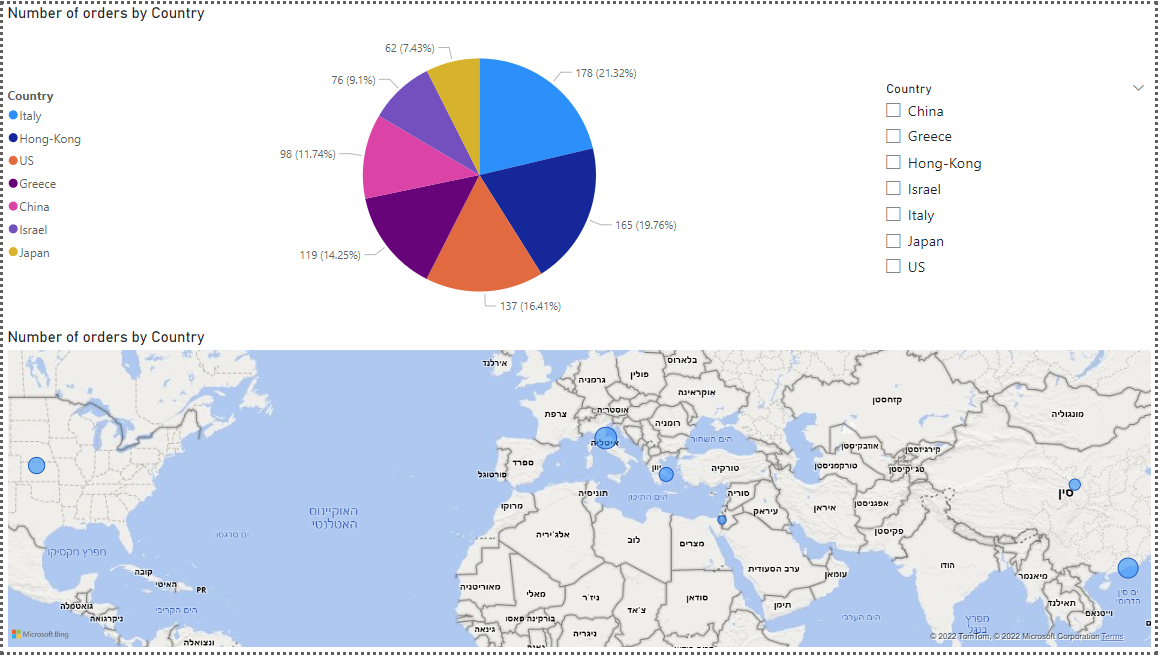
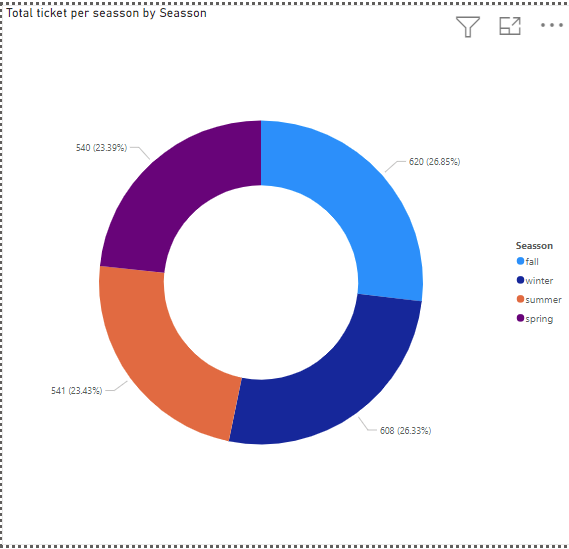
1. ה-VIEW מציג את התפלגות המבוגרים והילדים שמבקרים בכל אטרקציה. ניתן להסיק אילו אטרקציות פונות לקהל צעיר ואילו לקהל מבוגר יותר. באמצעות פילוח זה יוכלו המשווקים לשנות את השיווק כדי למשוך קהלי יעד שונים לכל אטרקציה.

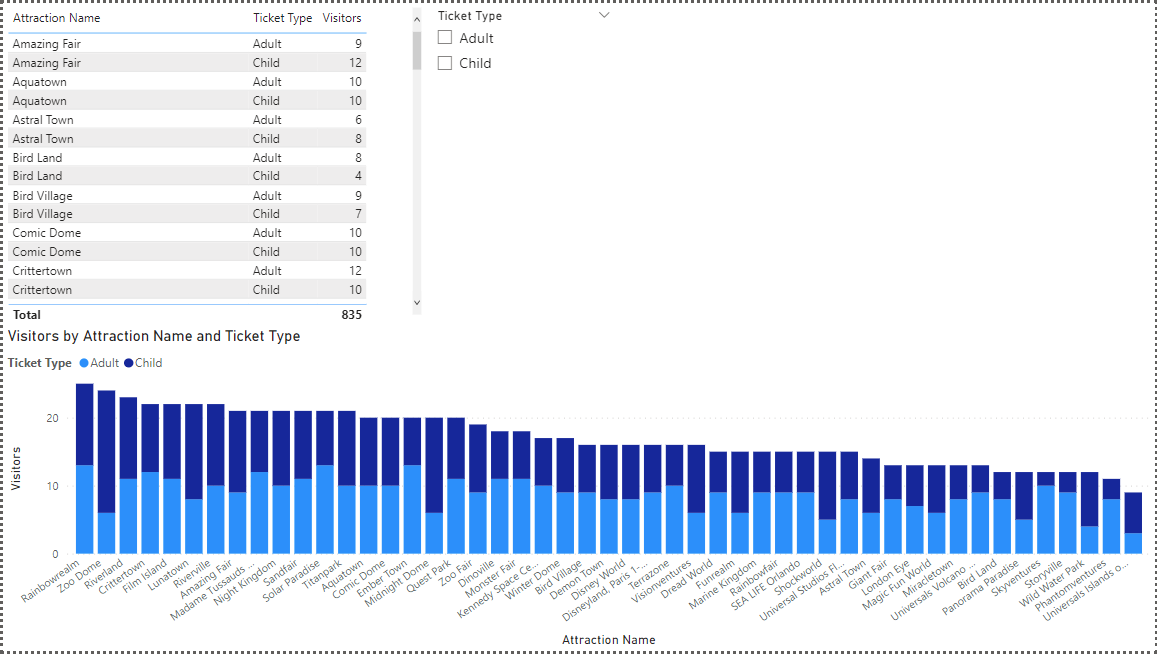
create view attractions\_by\_ticket\_type as

select t.AttractionName, t.ticketType ,visitors= count (\*)

from TICKETS as T

group by t.AttractionName,t.ticketType





**לוח מחוונים (10%)**

לוח המחוונים מציג תמונה כוללת של ההכנסות משנת 2020 עד שנת 2030 – התפלגות לפי חודשים ולפי אטרקציות. ניתן לבחור שנה מסוימת ולראות את התפלגות ההכנסות באותה השנה. כמו כן, ישנה אפשרות לסנן את האטרקציות לפי גודל ההכנסה.

כך החברה תוכל לשקול פעולות כלכליות רווחיות כמו מבצעים בחודשי שפל, העלאת מחירים בחודשי השיא, בחינת כדאיות עבודה מול אטרקציות שלא מניבות מספיק רווח וכדומה.

בכדי לחשב את סכום ההכנסות, השתמשנו בVIEW אשר מציג את הנתונים הדרושים, איחדנו בין טבלת ההזמנות, לכרטיסים ואטרקציות על מנת לשלוף את ההכנסות לפי שנים וחודשים.

CREATE VIEW Income\_attraction AS

select o.orderDT,a.AttractionName,

price = sum(a.price) OVER(PARTITION BY ( o.orderNumber))

from ATTRACTIONS as a join TICKETS as t on a.AttractionName =t.AttractionName

join ORDERS as o on o.orderNumber = t.orderNumber

נציג את החיפושים שהסתיימו ללא הזמנה כדי לראות את כמות ההזמנות שפוספסו ולהבין כיצד ניתן לצמצם מצבים כאלו.

CREATE VIEW Searches\_without\_ordering as

SELECT S.SearchByName, [total Searches without ordering]=count(S.SearchByName), S.DTSearch

FROM SEARCHES AS S

WHERE S.orderNumber IS NULL

GROUP BY S.SearchByName, S.DTSearch



## מטלה 4 (צבירה - עד 35%) – יישומים מתקדמים

## שאילתות עסקיות המשלבות Window Functions (10%)

**פונקציות "חלון" ראשונה:**

רעיון עסקי: מנהל המכירות טוען שמחלקת השיווק לא פועלת מספיק טוב בפרסום האטרקציות במספר מדינות, כדי לבדוק את טענתו לעומק ננתח את ההזמנות לפי המחיר ממוצע ההזמנות במדינה, נדרג את ההזמנות בכל מדינה לפי מחיר ההזמנה ונציג את ההזמנה היקרה ביותר שהתבצעה באותה המדינה. כך נוכל לשפר את השיווק, להתאים מבצעי קידום מכירות עבור כל מדינה ובכך להגדיל את נפח המכירות.

פלט: הפונקציה מציגה עבור כל הזמנה את המדינה של הלקוח, את מספר הזמנתו, את המייל שלו, את עלות ההזמנה, את עלות ההזמנה הממוצעת במדינה שלו, דירוג המדינה ואת ההזמנה הכי יקרה שהתבצעה במדינה.

select distinct country = o.[address-Country] ,

[order number] = o.orderNumber ,

o.email,

sale = s.sale ,

[avg sale per country] = avg(sale) OVER(PARTITION BY (o.[address-Country])),

rank () over (partition by o.[address-Country] order by sale desc )

[rank in each country],

FIRST\_VALUE(sale)

over (

partition by o.[address-Country] order by sale desc

rows between unbounded preceding and unbounded following) [hieghst price of this country]

from sale\_per\_order as s join ORDERS as o on o.orderNumber = s.orderNumber

order by o.[address-Country] , [rank in each country]

--drop view sale\_per\_order

create view sale\_per\_order as

select orderNumber = o.orderNumber ,sale = sum(a.price)

from ATTRACTIONS as a join TICKETS as t on a.AttractionName =t.AttractionName join ORDERS as o on o.orderNumber = t.orderNumber

group by o.orderNumber

**תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי(מדגם)**

**פונקציות "חלון" שנייה:**

רעיון עסקי: מנהל המכירות רוצה לעקוב אחרי השינויים השנתיים במכירות, לבדוק האם יש ירידה במכירות, לחקור את הסיבה ולמנוע המשך ירידה במכירות. כדי לבדוק את הסיבה לירידה נציין את המדינה שהזמינו ממנה הכי פחות במהלך השנה – כך נוכל לבדוק את הפערים בין המכירות ולהבין אם השינוי נקודתי או שמא מדובר בירידה בקנה מידה עולמי.

פלט: הפונקציה מציגה את התפלגות המכירות לפי שנים ומדינות - מציגה את שם המדינה, את השנה הנבדקת, את סך ההזמנות מהמדינה באותה השנה ואת סך אשתקד. בנוסף, מציגה את השינוי באחוז ההזמנות בין השנה הנוכחית לשנה הקודמת ואת המדינה שהזמינו ממנה הכי פחות באותה השנה.

select country,year, [total year sales],adjacent\_years.[previous year sales],[performance change in percentage] =round (

cast([total year sales]-[previous year sales] as float)/[previous year sales] , 2) , LAST\_VALUE(country) OVER(

partition by year

ORDER BY round (

cast([total year sales]-[previous year sales] as float)/[previous year sales] , 2)

RANGE BETWEEN

UNBOUNDED PRECEDING AND

UNBOUNDED FOLLOWING

) [country with the weakest performance of the year]

from (

select country,year, [total year sales],LAG([total year sales],1) OVER (

PARTITION BY country

ORDER BY year

) [previous year sales]

from sales\_per\_years

) as adjacent\_years

where year > '2020'

order by country ,year

--drop view sales\_per\_years

create view sales\_per\_years as

select year = year(o.orderDT),country=[address-Country], [total year sales] = sum(sale)

from orders as o join sale\_per\_order as s on s.orderNumber=o.orderNumber

group by year(o.orderDT) , [address-Country]

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

## שילוב מערכתי של מספר כלים (10%)

רעיון עסקי: "Attraction tickets" דוגלת בשיפור מתמיד ואחד מערכיה הבולטים הוא חווית הלקוח. החברה רוצה לשלוח סקר שביעות רצון ללקוחות לאחר ביקורם באטרקציה. לשם כך שומרת מאגר של "היסטורית לקוחות" – כל כרטיס שמומש נכנס לרשימה זו בחיבור עם המייל של המזמין.

פלט: רשומות המכילות את פרטי הכרטיסים שמומשו והמייל של המזמינים.

--drop table tickets\_history

create table tickets\_history (

AttractionName varchar(50) not null ,

Location\_ varchar(50) not null ,

IDTicket varchar(20) not null ,

DT DATE ,

ticketType varchar(15) ,

orderNumber varchar(20) not null ,

email varchar(30),

primary key (IDTicket),

foreign key (orderNumber) references ORDERS(orderNumber) ,

foreign key (AttractionName,Location\_) references ATTRACTIONS(AttractionName, Location\_) )

--------------------------

--drop function get\_email\_for\_customer\_reviews

create function get\_email\_for\_customer\_reviews (@id\_ticket varchar(20))

returns varchar(30)

as begin

declare @email varchar(30)

select @email=o.email

from ORDERS as O join tickets\_history as T on O.orderNumber=T.orderNumber

where t.IDTicket=@id\_ticket

return @email

end

------------------------------------stored procedure cancel old tickets

drop PROCEDURE update\_old\_tickets\_database

CREATE PROCEDURE update\_old\_tickets\_database

AS

delete

from TICKETS

where DT < '12/12/2023' --GETDATE()

---------------------------- triger move canceled tickets and update email

--drop trigger old\_tickets\_for\_customer\_review

create trigger old\_tickets\_for\_customer\_review

on tickets

for delete

as

insert into tickets\_history

select \* ,null

from deleted

update tickets\_history

set email = dbo.get\_email\_for\_customer\_reviews(t.IDTicket)

from tickets\_history as t

where t.email is null

------------------

----------------------checks

execute update\_old\_tickets\_database

select \*

from tickets\_history

תמונה שמכילה טקסט

התיאור נוצר באופן אוטומטי(מתוך 159 רשומות)

## דו"ח המושתת על שימוש בפסקת WITH מורכבת (10%)

החברה מפתחת מערכת ניתוח לחיפושים (SEARCHES) באמצעות מדדי הערכה מדדי הערכה לפריטים הנמכרים באתר (ATTRACTIONS) I – (PACKAGES) לדיוק ההצגה ושיפור המכירות.

* **SEARCHES** ( Search IP,DT search , Search by name, OrderNumber **(ORDERS)**)
* חיפושים: IP של החיפוש, תאריך ושעת החיפוש, הקלדת החיפוש, מספר הזמנה (אם בוצעה בעקבות החיפוש)
* **ATTRACTIONS** (Attraction Name , Location ,Details, Price )
* אטרקציה: שם האטרקציה, מיקום, פרטי אטרקציה, מחיר
* **PACKAGES** (Package name, Price, Discount)
* חבילות: שם החבילה, מחיר, הנחה
* **ORDERS** (OrderNumber, Order date, email, Address – Street, Address - Number, Address - City, Address - Country, Address - Postal Code,
* {CardNumber, Validity – month, Validity – year}(**PAYMENTS**) )
* הזמנות: מספר הזמנה, תאריך החיפוש, מייל, כתובת- רחוב, כתובת- מספר, כתובת- עיר, כתובת- מדינה, כתובת- מיקוד, מספר כרטיס אשראי, חודש פג תוקף אשראי, שנת פג תוקף אשראי
* **CREDITCARDS** ( CardNumber , Validity – month, Validity – year , cardCVV)
* כרטיס אשראי: (מספר כרטיס אשראי, חודש פג תוקף אשראי, שנת פג תוקף אשראי)

לניתוח רווחי החברה לפי החיפושים באתר הוכן דוח במבנה הבא:

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

תמונה שמכילה שולחן

התיאור נוצר באופן אוטומטי

הסבר השדות:

* rank by revenue, total revenue – סך ההכנסות ודירוג לפי הכנסה
  + יש להציג את ההכנסה של כל אטרקציה/ חבילה ואת ההכנסה הכוללת שלה על מנת להבין את רווחיותה לחברה.
* percentage of searches from total searches, numbers of searches – מספר החיפושים של כל אטרקציה/ חבילה ואחוזם מתוך כל החיפושים
  + יש להציג את מדדי החיפושים על מנת שהחברה תוכל להבין במה מתעניינים הלקוחות, ולבצע חשיפה יזומה לתחומים שפחות נחשפים אליהם.
* Name – שם האטרקציה/ חבילה שחופשה
  + יש להציג את שם האטרקציה/ חבילה על מנת שנוכל להתאים את החיפוש למוצג באתר.
* Type- סוג החיפוש
  + יש להציג את סוג החיפוש כדי להבין מה יותר פופולרי בקרב הלקוחות: אטרקציות/ חבילות.
* Price – מחיר האטרקציה/ חבילה
  + יש להציג את מחיר האטרקציה לחישוב הרווחים ועל מנת לבצע תמחור מחדש בעקבות הדוח.

**פתרון**

בלוק ה-WITH הבא מייצר בסופו תת-שאילתה בשם evalution המציגה לכל חיפוש אטרקציה או חבילה כמה פעמים חיפשו את האטרקציה, לכמה רווח החיפושים הללו הביאו, דירוג על פי הרווח וכמה אחוזים מהוו החיפוש לפריט מסוים מסך החיפושים ברשת.

with

Search\_classification as (

select type\_of\_product.name,

type\_of\_product.type,

s.SearchIP,

[leads to order]= (case when s.orderNumber is null then 'no' else 'yes' end ),

type\_of\_product.price

from (

select [name]= a.AttractionName,type= 'attration' ,a.price

from ATTRACTIONS as a

union

select [name]= p.packageName,type='package',p.price

from PACKAGES as p)as type\_of\_product join SEARCHES as s on s.SearchByName=type\_of\_product.name

),

Search\_popularity as (

select distinct sc.name ,

[numbers of searches]= count(\*) over ( partition by sc.name) ,

[percentage of searches from total searches] = round (cast( count(\*) over ( partition by sc.name) as float)/ (select count(\*)

from searches),3)

from Search\_classification as sc

),

revenue\_By\_name as (select distinct sc.name, [revenue]=sum(sc.price) over(partition by sc.name)

from Search\_classification as sc

where sc.[leads to order] = 'yes'

),

Evaluation as (

select distinct sc.name,

sc.type,

sc.price,

[numbers of searches],

sp.[percentage of searches from total searches],

[total revenue]=rev.revenue,

[rank by revenue] = case when

rev.revenue >1500 then 'A'

WHEN rev.revenue<1500 and rev.revenue>200 then 'C'

else 'B' end

from Search\_classification as sc join Search\_popularity as sp on sc.name=sp.name join revenue\_By\_name as rev on sp.name=rev.name

)

select \*

from Evaluation