

דוחות ניהול

- **ממוצע תפוזת טיסות שהתקיימו.**

עבור כל אחת מהטיסות שהתקיימו, נחשב את תפוזתה של הטיסה.
לבסוף נחשב את הממוצע (של כל "התפוזות").

איך עשינו את זה ב-SQL?

1. השכבה הפנימית (התחת-שאלילה)

כאן אנחנו בונים את הבסיס עבור כל טיסה בנפרד:

TotalCapacity): אנחנו מכפילים שורות בעמודות גם בחלוקת תיירים וגם בעסקים
ומחברים ביניהם. זה נותן לנו את מקסimum המושבים הזמינים במטוס.

סינון הזמנות מבוטלות: בחיבור לטבלת-orders, הוספנו תנאי
($= 'confirmed' = o.Status = AND$). זה מבטיח שהשאילתת "תראה" רק כרטיסים ששולם
ולא בוטלו.

ספרת כרטיסים (TicketsSold): אנחנו סופרים את מספר השורות בטבלה-h-Tickets
שמקורות להזמנות המאושרות.

חישוב אחוז התפוצה לטיסה: מחלקים את הkartisim שנמכרו בחלוקת הכללת ומכפילים
ב-100 כדי לקבל אחוז.

2. הקשר בין הטבלאות (JOINs)

JOIN בין טיסות למטוסים: חובה, כדי לדעת איזה מטוס ביצע איזו טיסה ומה הגודל שלו.
JOIN LEFT להזמנות וכרטיסים: קרייטי! השתמשנו ב-Join Left כדי שגם טיסות שלא
נמכר אף כרטיס (0 תפוצה) ייכנסו לחישוב הממוצע ולא ייעלמו מהדווח.

3. השכבה החיצונית (הסתיכום)

OccupancyRate(AVG): אחרי שיש לנו רשימה של אחוזי תפוצה לכל הטיסות במערכת,
אנחנו מחשבים את הממוצע הכללי שלהם. זו ה"שורה התחתונה" שהמנהלים ביקשו.

AverageOccupancyRa...

1.33250000

- **הכנסות בחחר גודל מטוס, יצירנית מטוס ומחלקה**

כלומר, נחשב את הכנסות הכלליות (בהתחשב בביטולים ובמידניות) עבור כל גודל מטוס, יצרנית מטוס ומחולקה.

חשוב לנו לציין שעבור חתך מסוים שסך הכנסות שלו הוא 0 לא הכנסנו רשומה - ככלומר למשל אם לא חמינו אף כרטיס במחלקה העסקים במטוסים גדולים של חברת "โบราינג" - לא נראה רשומה זו בדוח.

איך עשינו את זה ב-SQL? החלק הראשון: מטפל בהזמנות **confirmed** בטיסות פעילות. כאן הלוקוח שילם מחיר מלא, שכן אנחנו סוכמים 100% מעלות ה الكرטיס.

החלק השני: מטפל בהזמנות **cancelled** (ביטול ל Koh). לפי המדיניות עברו ל Kohות עבור אורחים ורשומים, הלוקוח מקבל זיכוי כמעט מלא אבל משלם 5% דמי ביטול. שכן אנחנו מחשבים רק 5% מהמחיר המקורי כהכנסה לחברת.

ביטול מנהל: לא הכנסנו תת-שאלתה עבור טיסות בסטטוס **cancelled** (ביטול מנהל) כי במערכת הזיכוי הוא מלא והתשלים מתעדכן ל-0, אך אין להן תרומה לשורת הכנסות.

קביעת גודל המטוס (AirplaneSize):

השתמשתי בתת-שאלתה קטינה בתוך **Select** שמחזירה 'Large' אם 0 או 'Small' אם הוא **NULL** או 0, בדיק ל פי הלוגיקה של **Seed** שלנו.

מניעת כפל מחירים:

השתמנו ב-**UNION** פנימי כדי לשולוף את המחיר הנוכחי (**EconomyPrice** או **BusinessPrice**) בהתאם למחלקה רשומה על ה الكرטיס.

Airpla...	Manufactur...	CabinClass	TotalRevenue
Large	Boeing	economy	1000.0000
Large	Airbus	business	1200.0000
Large	Airbus	economy	400.0000
Small	Airbus	economy	172.5000

עבור כל עובד (טיסים ודילים) שעבד, נחשב את שעות הטיסה המצטברות שלו שבהן עבד בטיסות קצרות (טיסות מתחת ל 6 שעות), ואת שעות הטיסה המצטברות שלו שבהן עבד בטיסות ארוכות (טיסות מעל 6 שעות).

איך עשינו את זה ב-SQL?

1. השכבה הפנימית (הפרדת נתונים גולמיים)

השתמשנו ב-ALL UNION כדי לאוסף נתונים מארבעה מקורות שונים בנפרד:

טיסים מול דילים: פיצלנו את השילפה כדי לפנות לtbodyות שונות (Pilot לעומת FlightAttendant).

קצרות מול ארוכות: לכל קבוצה עובדים יצרנו שתי שליפות: אחת שבודקת משך זמן קטן או שווה ל-360 דקות (6 שעות) ואחת שבודקת משך זמן גדול מ-360.

המרת דקות לשעות: חילכנו את Duration.f. Duration ב-60 כדי שהתוצאה הסופית תוצג בשעות עבודה ולא בדקות.

2. סינון קריטי (Status)

בכל אחת מארבע תתי-השאלות הוספנו את התנאי f.Status = 'active'.

המשמעות: השאלה מתעלמת מכל טיסה בסטטוס 'cancelled'. עובד ששובץ לטיסה

שבוטלה קיבל עלייה 0 שעות.

3. מבנה העמודות ("זמןויות")

כדי שנוכל לסקום את השעות בצורה נפרדת בסיכון הסופי, יצרנו מבנה שבו בכל שורה יש ערך רק בעמודה אחת:

טיסה קצרה: עמודת ShortHours מקבלת את משך הטיסה, ועמודת LongHours מקבלת 0.

טיסה ארוכה: עמודת ShortHours מקבלת 0, ועמודת LongHours מקבלת את משך הטיסה.

4. השכבה החיצונית (אגרגציה)

לאחר שכל הנתונים (מכל העובדים ומכל סוג הטיסות הפעילות) אוחדו לטבלה אחת גדולה

(RawData):

GROUP BY: אנחנו מקובצים את הנתונים לפי מידע העובד והשם שלו.

SUM: אנחנו סכמים את עמודת הקצרות בנפרד ואת עמודת הארכות בנפרד לכל עובד.

EmployeeID	FullName	Role	CumulativeShortHo...	CumulativeLongHo...
P003	Moshe Peretz	Pilot	2.6667	21.0000
P001	Yossi Mizrahi	Pilot	0.0000	21.0000
P002	Avi Goldberg	Pilot	0.0000	21.0000
A004	Yael Berger	Flight Attendant	2.6667	21.0000
A005	Tamar Fischer	Flight Attendant	2.6667	21.0000
A006	Liora Schwartz	Flight Attendant	2.6667	21.0000
A001	Maya Stern	Flight Attendant	0.0000	21.0000
A002	Noa Klein	Flight Attendant	0.0000	21.0000
A003	Shira Wolf	Flight Attendant	0.0000	21.0000

עבור כל חודש שבו הוזמנה לפחות טיסה אחת, נחשב כמה אחוז מסך ההזמנות בוטלו על ידי הליקות.

איך עושים את זה ב-SQL?

1. ייצרת **TotalData**

התת-שאילתת הראשונה מחשבת עבור כל חודש כמה ההזמנות נוצרו בסך הכל (גם תקינות וגם מבוטלות). אלה המוניה שלנו לחישוב האחוזים. השתמשנו ב-HMONTH כדי לקבץ את כל הטיסות לפי החודש בו הן יוצאות.

2. ייצרת "מוניה הביטולים" (**CancelledData**)

התת-שאילתת השנייה דומה לראשונה, אך היא כוללת סינון (WHERE o.Status = "cancelled"). היא סופרת רק את ההזמנות שבוטלו בכל חודש. אלה המוניה שלנו לחישוב.

3. חיבור הנתונים (**LEFT JOIN**)

חיברנו בין שתי הטבלאות לפי מספר החודש. השתמשנו ב-**LEFT JOIN** כדי להבטיח שאם בחודש מסוים היו 0 ביטולים, החודש עדיין יופיע בדוח (הוא יופיע בטבלה הכללית, אך לא בטבלה הביטולים).

4. טיפול בערכים ריקים (**IFNULL**)

מכיוון שהשתמשנו ב-**Left Join**, חודשיים ללא ביטולים יჩזירו ערך NULL. פונקציית **IFNULL** הופכת את ה-NULL למספר 0, כדי שנוכל לבצע עליו חישוב מתמטי מבלית שהשאילתת תישבר.

5. חישוב האחוז והעיגול

בשלב הסופי, השאילתת מבצעת את חישוב על הנתונים שקיבלנו ומחזיר את האחוז ביטולים (עבור כל אחד מהחודשים).

OrderMonth	CancellationRatePerc...
1	66.67
11	0.00

- **סיכון פעילות חודשית לכל מטוס בז' המטוסים** (מספר טיסות שבוצעו, מספר טיסות שבוטלו, אחוז ניצולות בהנחה שיש 30 ימים בחודש, זוג מקור-יעד השולט). עבר כל מטוס, (שהיה בו לפחות טיסה אחת בחודש זה, או לפחות הייתה אמורה להיות נחسب סיכון חדש של הפעולות שלו). איך עשינו את זה ב-SQL?

1. השכבה הפנימית: איסוף נתונים.

בשלב זה אנחנו מקבצים את כל הטיסות לפי **מזהה מטוס וחודש**.
FlightsPerformed (בוצעו): סופרים רק טיסות בסטטוס 'active'. זה עומד בהנחהות המדיניות – אם המנהל ביטל את הטיסה, היא לא נספרת כפעילות שבוצעה.
FlightsCancelled (בוטלו): סופרים את הטיסות שהמנהל ביטל (סטטוס 'cancelled').

2. חישוב אחוז הניצולות (UtilizationRatePercent)

על בסיס הנתונים מהשלב הקודם, אנחנו מחשבים כמה המטוס היה "עסוק" מתוך פוטנציאלי של 30 ימים בחודש.

הנוסחה: {מספר טיסות שבוצעו} / 30 (ימים בחודש)

3. מציאת הנתיב השולט (DominantRoute)

כדי למצוא את זוג המקור-יעד השכיח ביותר, השתמשנו בתת-שאילתת מקושרת (Correlated Subquery):

מעבר כל שורה בדו"ח, השאילתת הולכת לטבלת הטיסות ומ探שת את כל הנתיבים של אותו מטוס באותו חודש.

היא מקצת את אותם (GROUP BY), סופרת כמה פעמים כל נתיב הופיע, ומחזירה רק את זה שהופיע הכי הרבה פעמים (ORDER BY COUNT(*) DESC LIMIT 1).

4. סיכון ותצוגה סופית

השכבה החיצונית לחברת את כל הנתונים (מזהה מטוס, חודש, כמות, אחוז ניצולות והנתיב השולט) לכדי שורה אחת ברורה לכל מטוס בכל חודש.

AirplaneId	FlightMonth	FlightsPerform...	FlightsCancell...	UtilizationRate...	Dominant...
PLANE-002	11	0	1	0.00	LCA-TLV
PLANE-003	11	1	0	3.33	JFK-LCA
PLANE-001	11	1	0	3.33	TLV-JFK
PLANE-004	1	1	0	3.33	TLV-RUH