



בסכימה קיימות מספר טבלאות, ועל כל שדה נזהה איומים, כיצד יתבטאו בנתונים, מה תהיה ההשפעה, ומה הפתרון.

Yearmonth

לפני שניגע בשדות השונים, נשים לב שהטבלה כולה תוצאה של סיכום של טבלת הטרנזקציות. ככל הנראה בוצע pre-compute כדי להאיץ חישובי סיכומים. המידע בטבלה כפול ולכן נוצר איום אי תאימות.

:CustomerID

ערך NULL או כזה שלא קיים ב-customer. זה יגרום לרשומות יתומות ולחישובי צריכה שגויים ע"פ לקוח.

הפתרון: הגדרת NOT NULL, מפתח זר ל-customer(CustomerID) בכדי למנוע רשומות בלי לקוח, לקוחות שאינם קיימים וכפילויות.

:Date

הסוג הוא int ולא תאריך. זו כבר בעיה כי נדרש תרגום מהמספר לתאריך. חישובים על התאריכים (כמו צריכה לשנה) גם יצטרכו להתייחס לתרגום ויהיו יותר מורכבים. בנוסף, אין לנו את הגנת הסוג תאריך.

הפתרון : הגדרת השדה כDATE

:Consumption

איום : ערכים שליליים או מופרזים (ערכים מופרזים הם עניין של החלטה עסקית, לכן כשנגדיר אילוץ נגדיר רק גדול שווה 0 בכדי לתת נקודת מבט אובייקטיבית ככל שניתן). זה יגרום לעיוות מדדים עסקיים.

הפתרון: $CHECK Consumption \geq 0$.

Customers

CustomerID

כפילויות או NULL. זה יגרום לשיוך כפול או חסר ללקוחות. זאת בעקבות העובדה שזהו מפתח ראשי בטבלה זו.

הפתרון: PRIMARY KEY

Segment

איום: ערך שאינו מאחד הסגמנטים הרצויים.

הפתרון: $CHECK (Segment \in \dots)$ או מפתח זר לטבלת segments.

Currency

איום: קוד לא תקני ("nis", "\$"). זה יגרום להמרות לא נכונות.

פתרון: פתרון: $CHECK (Currency \in ('ILS', 'USD', 'EUR'))$ (NOT NULL +) או מפתח זר לטבלת currencies שמכילה רשימה סופית של קודי מטבעות תקינים

gasstations

GasStationID

איום: כפילויות/NULL.

הפתרון: PRIMARY KEY

ChainID

איום: הפנייה לרשת לא קיימת.

הפתרון: מפתח זר ל-chains(ChainID).

Country

איום: ערכים חופשיים ולא סטנדרטיים ("Israel", "IL").

הפתרון: CHECK או FK לטבלת מדינות (יש צורך ליצור אותה).

Segment

: אי-עקביות מול הגדרת Segment ב־ customers.

מלבד בעיות השימוש בערכים הרגילות, המונח segment מופיע בשתי טבלאות. מהסכמה לבדה לא ניתן לדעת האם הם צריכים להיות זהים או לא.

הם הם צריכים להיות זהים, חייבים להשתמש בטבלה segments אחת כדי לאכוף התאמה. אם הם לא צריכים (לדוגמה לקוח פרטי/עסקי, תחנה בצפון/מרכז/דרום), חובה להשתמש בשמות שונים וטבלאות שונות.

הפתרון: להגדיר מילון נפרד לתחנות, לדוגמה: ('עירוני', 'פריפריאלי', 'כביש מהיר'). ניתן לקרוא לשדה StationSegment לצורך הבחנה.

products

ProductID

איום: כפילויות/NULL.

הפתרון: . PRIMARY KEY

Description

איום: ערכים ריקים/כתיב חופשי היכולים ליצור טעויות ובלבול בין מוצרים דומים שמית.

הפתרון: CHECK + NOT NULL לאורך ותוכן..

transactions

– Date/Time

איום: תרגום הזמן למספר

פתרון - יצירת טבלה של מזהה ייחודי לתאריך.

:CustomerID

איום: NULL או לא קיים. זה יגרום לעסקאות יתומות.

הפתרון:

FK ל-customer.

CardID:

איום: כרטיס שלא קיים.

הפתרון: לאפשר NULL לעסקאות מזומן או לדוגמה להשתמש ב-DEBIT CARD (כרטיס מזומן).

:- GasStationID / ProductID

ערכים לא קיימים.

הפתרון: FK לטבלאות gasstations ו-products.

- Amount

איום: אפס, שלילי.

הפתרון: $CHECK\ Amount > 0$.

:Price

איום: אפס, שלילי או דיוק לא עקבי.

הפתרון:

- $CHECK\ Price > 0$ והמחיר חייב להיות חיובי.

- צריך להחליט על דיוק עקבי שמתאים לנתונים (למשל DECIMAL(12,3) למחירי דלק).

****הערה כללית:** כפי שניתן לראות גם במבחן Example 2 (שממנו נלקחה הסכמה), אחת הבעיות החוזרות היא ייצוג מחיר לא עקבי. בעיה זו עלולה להצטבר לשגיאות משמעותיות בחישובי הכנסות ובפיוס מול מערכות חיצוניות, ולכן חשוב להגדיר את השדה Price כ-DECIMAL עם דיוק עקבי (לדוגמה DECIMAL(12,3) למחירי דלק), לצד אילוצים נוספים שיבטיחו שלמות נתונים.**

Transactions_1k

ככלל, מטרת הטבלה עצמה לא ברורה. ככה"נ הכוונה היא לדגימה, ולאחר מכן נוצרת כפילות שעשויה להוות איום. אם הסכמה לא מתעדכנת, האיום לא יתקיים.

בנוסף, כל יתר האיומים המופעים בטבלה המקורית, קיימים גם כאן.

