Document DB Design

ארכיטקטורה:

**הגדרת טבלה:**

הטבלה תמומש כך:

{ key: {}

, key: {}

, key: {}

….}

בדיקט הראשי מפתח מס 0 יביא אותי לדיקט נוסף המכיל את שדות החובה,

בנוסף כל מפתח יביא לדיקט המכיל את כל מאפייני האוביקט של אותו מפתח.

**מחיקת טבלה:**

עבור כל מפתח נמחק את הדיקט אליו הוא מביא כלומר הדיקט המכיל את מאפייניו, לאחר מכן נמחק את הדיקשנרי הראשי.

o(n)סיבוכיות:

**הכנסת רשומה לטבלה:**

כאשר מוסיפים רשומה לטבלה נבדוק אם יש לה את שדות החובה ורק אם כן נכניס את המפתח שלה לדיקשנרי הראשי כך שהוליו של אותו מפתח יהיה בעצם דיקשנרי נוסף המכיל את שאר הנתונים שלו.

o(1) סיבוכיות:

**מחיקת רשומה מטבלה:**

כאשר מוחקים רשומה מהטבלה נקבל את המפתח שלו למחיקה ניגש לפי המפתח בדיקשנרי הראשי ונמחק את הדיקשנרי אליו המפתח מוביל.

o(1) סיבוכיות:

**עדכון רשומה בטבלה:**

ניגש בדיקשנרי הראשי למפתח שנקבל וממנו נגש דיקשנרי בו מעודכנים נתוני הרשומה ונעדכן עפ"י הנתונים החדשים.

o(1) סיבוכיות:

**חיפוש רשומות העונות על תנאי/ים מסוימים בטבלה:**

מעבר על פני כל המפתחות בדיקשנרי הראשי ובדיקה האם דיקשנרי הנוסף אליו הוא מפנה מקיים את התנאי א"כ

נחזיר את הרשומה הנוכחית –כך לכל רשומה.

o(n)סיבוכיות:

**מחיקת מספר רשומות לפי תנאי/ים מסוימים מטבלה:**

מעבר על פני כל המפתחות בדיקשנרי הראשי ובדיקה האם דיקשנרי אליו הוא מפנה מקיים את התנאי א"כ נמחק את הקובץ ואת המפתח כפי שנבצע במחיקת רשומה מהטבלה.

o(n)סיבוכיות:

**מעלות**:

גישה מהירה לכל אובייקט בפני עצמו וע"כ יכולת ניתוח מהירה של כל הנתונים שלו, הוספת ומחיקת אובייקט במהירות, גמישות- לכל אובייקט יש נתונים שונים הנצרכים לו אין צורך במבנה יחיד לכל האובייקטים.

**חסרונות:**

אינו תומך בקשרים בין אובייקטים שונים – חיפוש או מחיקת רשומות לפי תנאי מסוים יהיה בסיבוכיות גבוהה.

**השיקולים לבחירה זו:**

מתאים למקרים כגון:

בהם חשוב ניתוח פנימי של כל אובייקט בפני עצמו וכן יש חשיבות לאפשר לכל אובייקט שדות אחרים,

יש פחות חשיבות לניתוח נתונים בין כמה אובייקטים.