# מעביד הנחפפים

## הקדמה

בתרגיל זה יש שלושה חלקים. בחלק הראשון תלמדו לנתח בעיה ולהכין מסמך "דיזיין" לפני ביצוע המשימה. בחלק השני תיצרו api שיהיה ניתן לפנות אליו והמידע יישמר בקובץ json מקומי. בחלק השלישי תלמדו לעבוד מול mongoDB

## תיאור

אני רוצה לפתח מערכת שבה אוכל לראות את כלל המשימות שיש בחפיפה. ועבור כל נחפף לראות את כמות התרגילים שהוא סיים בכל חלק.

**תיאור המשימות:** רשימה קבועה של מטלות. לכל מטלה יש מזהה שם ורמת קושי. המטלות מחולקות לפי קטגוריות של נושאים (nodejs, mongo, elastic, Jenkins וכו). לצורך פשטות, הניחו כי כל משימה שייכת לקטגוריה אחת בלבד.

**תיאור הנחפפים:** לכל נחפף יש מזהה, שם, דרגה, מספר טלפון ורשימה של מטלות שהנחפף סיים

תפקידכם לייצור api שיתמוך בפעולות הבאות:

עבור המשימות:

* הוספת משימה חדשה
* קבלת כל המשימות לפי קטגוריה
* קבלת כל המשימות בקטגוריה מסוימת
* קבלת מידע על משימה לפי המזהה שלה
* עדכון השם של המשימה
* מחיקת משימה לפי מזהה

עבור הנחפפים

* הוספת נחפף חדש
* קבלת מידע על כל הנחפפים
* קבלת אחוז המשימות שכל נחפף ביצע בנפרד בפנייה אחת
* קבלת מידע על נחפף אחד
* עדכון שנחפף ביצע משימה מסויימת
* קבלת מספר המשימות שנחפף מסוים ביצע בכל קטגוריה ביחד עם מספר המשימות הכולל בכל קטגוריה

כל נחפף יכול לראות רק המידע הרלוונטי אליו. רק המנהל יכול לראות מידע שרלוונטי לכל הנחפפים ולבצע עדכונים.  
לצורך פשטות הניחו שהמזהה של מי שביצע את הפנייה נמצא ב-headers של הבקשה בשדה בשם user\_id. אם היוזר הוא מנהל, יועבר בשדה הערך admin.

## חלק ראשון

בחלק זה עליכם להכין מסמך מפורט לגבי הדרך בה תממשו את הדרישות.

על המסמך להכיל את הדברים הבאים:

* ספריות שתשתמשו בהם
* מודלים של מידע הנשמר – ממשקים, איך תשמרו את המידע, באיזה צורה
* כתובות לפניות, Http Methods (Get,Post,Delete,Put,Patch), פרמטרים שמעבירים לכל בקשה ואיך
* אילו קבצים תיצרו, אילו פעולות תיצרו ואיך הם יתממשקו אחד לשני
* תככנו את העבודה בצורה נכונה ובמינימום תלות, כך שתוכלו בהמשך להתממשק עם DB בקלות

## חלק שני

* בחלק זה תתחילו לבצע את מה לפני הדיזיינים בחלק הראשון.
* בשלב זה אין צורך לעבוד מול DB אמיתי, מספיק לשמור את כל המידע במקום אחד בשרת וכל פעם לעדכן אותו
* ממולץ לבדוק שהפניות עובדות בעזרת POSTMAN
* ניתן לבדוק את הקוד שאתם כותבים ע"י הרצת הפקודה **npm start** שמריצה את הקוד, מאזינה לשינויים שאתם מבצעים בקוד ומריצה את הקוד מחדש אוטומטית

## חלק שלישי

* בחלק זה יש להוסיף את הספרייה mongodb לפרויקט ולממש את הפעולות הנדרשות כך שכל המידע יישמר ל-DB גם אם השרת למטה