

הנחיות להגשת פרוייקט שלב 1: אפליקציית ניהול משימות

מועד הגשה: לפי לוח"ז באנייבלי

מטרת הפרוייקט בשלב ראשון היא לפתח אפליקציית CRUD להוספה, עריכה, חיפוש\סינון, ומחיקה של משימות (ראו פירוט בהנחיות להלן).

1. ההגשה היא חובה ותהווה חלק מהציון הסופי. אפשר להגיש בזוגות.

2. יש לכתוב את כל הקוד בעצמכם ללא עזרה או העתקה ממקורות חיצוניים. מותר להשתמש בקוד מתרגילים שעשינו בשיעור. אסור להשתמש בכלים שמייצרים קוד כמו chatgpt.

אפיון:

האפליקציה נועדה לניהול משימות: הוספה, עריכה, חיפוש\סינון, מחיקה של משימות.

השליטה באפליקציה תתבצע דרך תפריט בטרמינל כפי שלמדנו.

הנתונים יישמרו בקבצי pickle.

הנחיות:

1. (20 נקודות) יש לחלק את הקוד לפונקציות ומודולים כפי שלמדנו (...app, functions, classes)

תזכורת לכללים לכתובת פונקציות בצורה נכונה שלמדנו:

<https://dev.to/levivm/coding-best-practices-chapter-one-functions-4n15>

2. (55 נקודות) יש להשתמש ב OOP. צריכה להיות לפחות מחלקה אחת בשם Task עם מינימום התכונות הבאות:

- שם המשימה (טקסט חופשי)
- תאריך בו נוצרה המשימה (יכול להיות מחרוזת או שימוש בספרייה datetime או מחלקה משלכם).
- לדוגמא:

```
today=datetime.date.today()
date=datetime.date(day=21, year=2022, month=12)
```

- תאריך יעד לסיום (כנ"ל)
- שם האחראי לביצוע המשימה (טקסט חופשי או 10 נקודות: לבחור אחראי מרשימה של משתמשים שלכל אחד יש משימות משלו)
- תיאור המשימה בטקסט חופשי
- קטגוריה (למשל עבודה, קניות, חופשה וכו'). טקסט חופשי או 5 נקודות: לבחור מרשימה של קטגוריות.

קישורים ל-OOP:

<https://www.geeksforgeeks.org/python-oops-concepts/>
https://www.w3schools.com/python/python_classes.asp

3. בדיקות (25 נקודות)

20 נקודות: יש להוסיף קובץ tests עם פונקציה שיוצרת משימות אקראיות לצורך בדיקות. המשימות לבדיקה צריכות להיות כמה שיותר דומות למשימות אמיתיות כפי שעשינו עם תרגיל המוסך.

5 נקודות: יש לכלול ב tests בדיקה לכל אחת מהפעולות CRUD. לדוגמא, בדיקה שמוסיפה משימה ואז מוודאת שאכן המשימה נשמרה ב-pickle.

פרוייקט אישי

עבור התלמידים המעוניינים להגיש פרוייקט אישי בקורס יש לעמוד בתנאים הבאים:

1. סיום משימת פרוייקט ניהול משימות הנ"ל בציון לפחות 90

2. הגשת אפיון עם תיאור הפרוייקט המוצע

3. קבלת אישור מהמרצה.

בהצלחה :