

מטלה 6

ממשקים | מחלקה אבסטרקטית | פולימורפיזם | מחלקת Object

התוכנית אותה תממשו תעסוק בניהול אפליקציות המותקנות במכשיר הסלולרי של משתמש

1. **תצרו ממשק בשם IApp** והגדירו בו מתודה בשם AddVAT מקבלת אחוז מע"מ. אחראית להוסיף מע"מ למחיר מוצר/שירות.

2. **צרו מחלקה אבסטרקטית בשם AppSystem** המאופיינת ע"י **מספר ייחודי** (מספר רץ החל מ-1), **שם האפליקציה**, **מחיר להורדה ותאריך** מטיפוס **DateTime** המייצג את תאריך הורדת אפליקציה למכשיר.

יש לממש את המתודות:

- תכונות (properties) לכל השדות. יש לבצע בדיקת תקינות ערכים עבור שם האפליקציה ומחיר. במידה והתקבל null עבור שם האפליקציה, זרקו חריגה מסוג `ArgumentNullException` עם הודעה מתאימה. במידה והתקבל ערך שלילי עבור המחיר יש לזרוק חריגה מסוג `ArgumentException` עם הודעה מתאימה.
- בנאי שמקבל את שם האפליקציה ומחיר. אתחלו את השדות שם ומחיר בערכים שהתקבלו, תאריך הורדת אפליקציה יש לאתחל בתאריך הנוכחי של המחשב ואת המספר הייחודי של האפליקציה אתחלו בערך ייחודי (חשבו כיצד לעשות זאת).
- דריסה למתודה `ToString` אשר תחזיר מחרוזת המייצגת את כל פרטיו של האובייקט.
- מתודה אבסטרקטית בשם `AppSystemPurpose` המחזירה מחרוזת המייצגת את מטרת האפליקציה
- ממשו את הממשק `Comparable` כך: המתודה `CompareTo` תבצע השוואה בין שם האפליקציה של האובייקט שהפעיל את המתודה לבין שם האפליקציה של האובייקט שהתקבל כפרמטר.

3. **צרו מחלקה בשם Navigation** המייצגת אפליקציית ניווט המאופיינת ע"י **מספר ייחודי** (מספר רץ החל מ-1), **שם**, **מחיר להורדה**, **תאריך** מטיפוס **DateTime** המייצג את תאריך הורדת אפליקציה למכשיר, ומנהל הניווט (אובייקט מסוג `NavigationManager`).

a. **צרו מחלקה בשם NavigationManager** המאופיינת ע"י מיקום נוכחי (מחרוזת המייצגת כתובת) ומערך כתובות יעד (`string[]`) אליהם ניווט המשתמש, מספר כתובות יעד שיש בפועל במערך וסוג רכב: '1' מייצג רכב פרטי, '2' מייצג אופנוע, '3' מייצג מונית (הגדירו `enum` עבור הרשימה של הסוגים)

- הגדירו `properties` לכל אחד מן השדות.
- הגדירו **בנאי** שמקבל את כל השדות **למעט השדות** מערך כתובות היעד ומספר כתובות היעד בפועל. אתחלו את השדות שהתקבלו כפרמטר. את מערך כתובות היעד יש לייצר בגודל 0 ואת מספר כתובות היעד יש לאתחל ב-0.
- דריסה למתודה `ToString` המחזירה מחרוזת המייצגת את שדות האובייקט

- הגדירו מתודה בשם ShowRecentLocations () המדפיסה את המיקומים האחרונים אליהם ניווט המשתמש. אין להדפיס כתובת יותר מפעם אחת.
- הגדירו מתודה בשם AddAddress המקבלת כתובת יעד לניווט (מחרוזת), דאגו להגדיל את המערך באחד והוסיפו את הכתובת שהתקבלה כפרמטר למערך במקום הפנוי לפי מספר כתובות היעד בפועל. יש לבצע הוספה של הכתובת רק אם היא לא קיימת במערך. במידה וקיימת אין צורך להוסיפה שנית.

דגש חשוב! אין לשכפל שדות/מתודות שהוגדרו במחלקת AppSystem

- b. במחלקת Navigation יש לממש:
 - בנאי שמקבל את השדות, שם האפליקציה, מחיר להורדה ומנהל ניווט. אתחלו את השדות בערכים שהתקבלו.
 - דריסה למתודה ToString אשר תחזיר מחרוזת המייצגת את כל פרטיו של האובייקט. אין לשכפל קוד.
 - את הממשק IApp כך שהמתודה תוסיף למחיר אפליקציית ניווט מע"מ של 12% (ערך זה צריך להתקבל כפרמטר מהתוכנית - יש להגדירו כקבוע).
 - ממשו את המתודה האבסטרקטית AppsystemPurpose אשר תחזיר את המחרוזת הבאה: "Catch The Road-Choose The Best Way"

4. צרו מחלקה בשם Social המייצגת אפליקציות חברתיות המאופיינת ע"י מספר ייחודי

(מספר רץ החל מ-1), שם האפליקציה, מחיר להורדה, תאריך **DateTime** מטיפוס המייצג את תאריך הורדת אפליקציה למכשיר, דירוג (ערכים בין 1 ל-5) ושדה בוליאני האומר האם האפליקציה החברתית מיועדת לארגון.

דגש חשוב! אין לשכפל שדות/מתודות שהוגדרו במחלקת AppSystem

יש לממש במחלקה:

- תכונות (properties) עבור כל שדה. יש לבצע בדיקת תקינות עבור הדירוג שנמצא בטווח הערכים 1 עד 5. במידה ולא זרקו חריגה מסוג ArgumentException עם הודעה מתאימה.
- בנאי המקבל את השדות שם האפליקציה, מחיר להורדה, דירוג והאם האפליקציה מיועדת לארגון. אתחלו את השדות בערכים תקינים.
- את הממשק IApp כך שהמתודה תוסיף למחיר אפליקציית ניווט מע"מ של 13% (ערך זה צריך להתקבל כפרמטר מהתוכנית - יש להגדירו כקבוע)
- את המתודה האבסטרקטית AppsystemPurpose אשר תחזיר את המחרוזת הבאה: "Far away and talking close".

5. **צרו מחלקת בשם MobileDevice** המאופיינת משם משתמש (מחרוזת), סיסמה (מחרוזת), האם

פעיל (משתמש מחובר למכשיר), מספר ניסיונות התחברות למכשיר, מערך הטרוגני של

אפליקציות (מסוג **AppSystem**) ומספר איברים בפועל במערך.

יש לממש במחלקה את המתודות:

- תכונות (properties) לכל השדות. שם המשתמש יכול להכיל אותיות בלבד - במידה והשם מכיל תווים שאינם אותיות יש לזרוק חריגה מסוג `ArgumentException`.
- בנאי המקבל שם משתמש וסיסמה. אתחול את השדות בערכים שהתקבלו. אתחול את המערך בגודל 0 ואת מספר האיברים בפועל ב-0.
- `AddApp` המקבלת הפניה מסוג **AppSystem** ומוסיפה את האפליקציה למערך. יש לוודא כי האפליקציה לא קיימת כבר במערך (השוואה עפ"י שם האפליקציה). במידה והאפליקציה קיימת במערך יש לזרוק חריגה מסוג `ArgumentException` עם הודעה מתאימה. בהתאם לסוג האובייקט שמועבר בזמן ריצה שמרו אותו במערך במקום הפנוי.
- `showListAppNavigation()` המתודה תדפיס את כל מספרי האפליקציות ניווט ושמות האפליקציות ניווט.
- דריסה למתודה `ToString` אשר תחזיר מחרוזת המייצגת את כל פרטיו של האובייקט. כולל רשימת האפליקציות.
- `PopularNavigationApp()` המתודה מחזירה הפניה לאפליקציה הפופולרית. אפליקציה פופולרית הינה האפליקציה עם הכי הרבה כתובות יעד אליהם ניווט המשתמש. במידה ואין ברשימה אפליקציות ניווט יש להדפיס הודעה מתאימה.
- מתודת `login` המקבלת שם משתמש וסיסמה. הגדילו את השדה מספר ניסיונות התחברות באחד והחזירו אמת **רק אם** שם המשתמש והסיסמה תואמים את שם המשתמש והסיסמה המוגדרים באובייקט ואם החשבון אינו חסום, חשבון חסום יהיה חסום במידה והשדה מספר ניסיונות התחברות עולה על 9. **בנוסף**, במידה ומספר הניסיונות מעל 3, יש להמתין 15 שניות לפני סיום המתודה (המחזירה שקר). במידה ומספר הניסיונות מעל 9, יש לחסום את המכשיר, זרקו חריגה מסוג `Exception` עם הודעה מתאימה.

המתנה של 15 שניות ניתן לבצע באמצעות המתודה `sleep(15000)` הממומשת בספריית

System.Threading (ראו לינק עם הסברים :

<https://docs.microsoft.com/en-us/dotnet/api/system.threading.thread.sleep?view=net-5.0>)

6. **הגדירו תוכנית ראשית (main)** בה הגדירו את המכשיר הסלולרי שלכם (**MobileDevice**) אתחולו

את שם המשתמש והסיסמה. יש לאפשר למשתמש לבצע התחברות: בקשו מהמשתמש להזין שם

משתמש וסיסמה. יש לאפשר למשתמש לבצע התחברות עד אשר תיתפס חריגה המציינת חסימת

מכשיר. במקרה של חסימה יש לסגור את התוכנית.

רק לאחר התחברות, יש להציג למשתמש תפריט עם האופציות הבאות:

1. הורדת אפליקציה חדשה
2. צפייה באפליקציית הניווט הפופולרית (האפליקציה שהמשתמש עשה בה הכי הרבה שימוש)
3. ניווט באמצעות אפליקציה
4. הדפסת מפרט נתוני המכשיר
5. מיון אפליקציות
6. סגירת המכשיר

בלחיצה על 1 – שאלו את המשתמש האם ברצונו להוריד אפליקציה ניווט או אפליקציה חברתית, הורדת אפליקציה מחייבת את המשתמש להזין פרטים אודות האפליקציה. בנוסף יש להוסיף מע"מ למחיר האפליקציה. בהתאם להחלטת המשתמש הוסיפו את האפליקציה לרשימה המתאימה (הפעילו את המתודה AddApp).

בלחיצה על 2 – הציגו למשתמש את פרטי האפליקציה הפופולרית ביותר.

בלחיצה על 3 – ראשית הציגו רשימה של כל האפליקציות ניווט שקיימות במכשיר (מספר ייחודי ושם אפליקציה בלבד), לאחר מכן בקשו מהמשתמש להזין את שם האפליקציה ברצונו להתחבר. לפי בחירת המשתמש יש להציג את נתוני האפליקציה: המיקום הנוכחי, רשימת כתובות היעד אליהם ניווט המשתמש. בנוסף בקשו למשתמש להזין כתובת יעד אליו ירצה לנווט. לאחר הזנת כתובת היעד הדפיסו למשתמש הודעה "נסיעה נעימה ובטוחה".

בלחיצה על 4 - הציגו למשתמש את כל פרטי המכשיר הסלולרי כולל רשימת האפליקציות

בלחיצה על 5 - הפעילו מתודה Sort ממחלקת Array על רשימת האפליקציות.

בלחיצה על 6 – יש להציג למשתמש הודעת goodbye ולסגור את התוכנית.

דגשים נוספים:

- ניתן להוסיף פונקציות עזר במידת הצורך – אין לשכפל קוד
- חשוב! אין לשכפל קוד, יש לעשות שימוש במידת הצורך במתודות קיימות, גם אם מדובר במתודה של מחלקת אב.
- חשוב! בעקבות עקרון הפולימורפיזם ומחלקה אבסטרקטית ייתכן ותצטרכו לבצע המרות.
- יש לדאוג לתפוס את חריגות ולהדפיס את ההודעה למשתמש, לאחר מכן אפשר למשתמש לבחור אופציה נוספת מהתפריט עד אשר יבחר באופציה 6 (סיום)
- ניתן להוסיף getter או setter במידת הצורך, אך שימו לב שאין צריך בכך לטובת מחלקת הבן בהורשה יש לדאוג לטפל בהרשאות מתאימות.
- יש להגיש את המטלה בפורמט הרגיל. ניתן לצפות בנוהל הגשת מטלות

בהצלחה