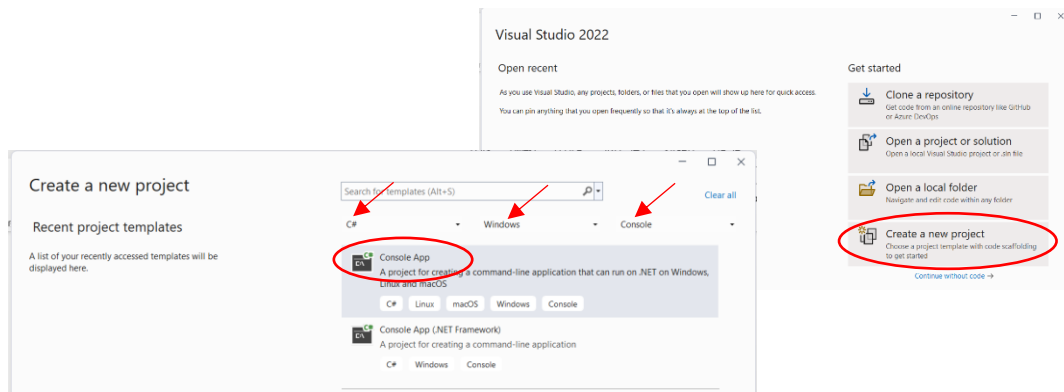


פרויקט C# - שלב 1 א' – יצירת פרויקטים לשכבת DAL

1. יצירת פרויקט

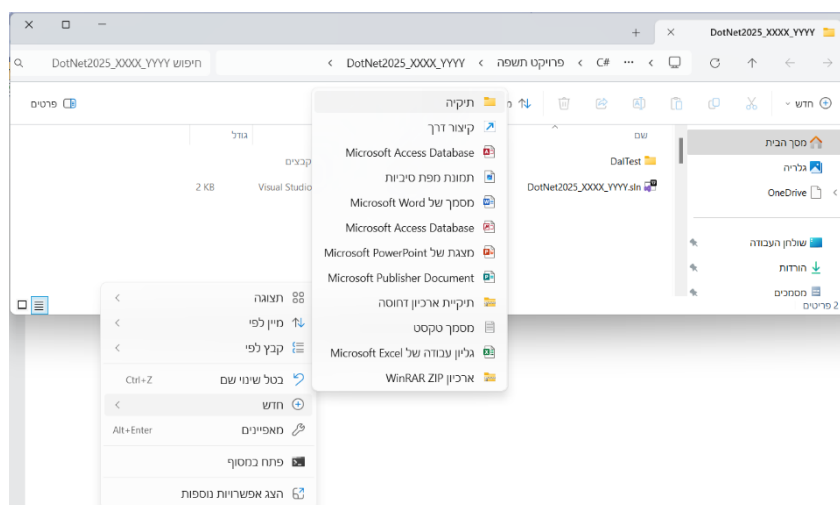
צרו פרויקט חדש **Create a new Project** (או בתפריט **File=>New=>Project...**)
ובחרו בניית פרויקט מסוג מסוג **Console Application**:
נ.ב. בשום פנים ואופן לא לבחור סוג **Console Application (.NET Framework)**



- קנפגו את הפרויקט לפי הסדר כפי שמופיע כאן – קודם כל שם הפרויקט ורק לאחר מכן שם הפתרון!
- שם הפרויקט (**Project**) יהיה **DalTest** (פרויקט קונסול לבדיקת שכבת הנתונים)
שם הפתרון (**Solution**) יהיה **DotNet2025_XXXX_YYYY**
כאשר XXXX, YYYY הן 4 הספרות האחרונות של ת.ז. של כל אחת מבנות הזוג.
- המיקום (**Location**) – איפה תמוקם תיקיית הפרויקט, יש לבחור מיקום נוח עבורכן
- חשוב!! אין לסמן בתיבת סימון **Place solution and project in the same directory!**
לחצו על כפתור **Next**. בחלון הבא וודאו ש-Target Framework מוגדר כ- **NET 6.0** לפחות ולחצו על **Create**.

2. יצירת תיקיית הרצה

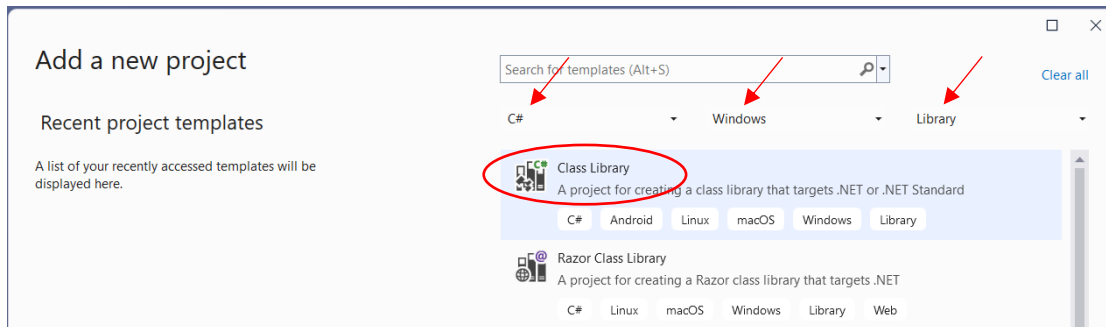
בתיקיה הראשית של ה-Solution יש ליצור תת-תיקיה בשם bin. היצירה חייבת להתבצע בסייר קבצים - File Explorer, בשום פנים ואופן אסור "ליצור" את התיקיה מתוך VS כדי להגיע לתיקיית ה solution מתוך VS, יש ללחוץ במקש ימני על ה solution ולבחור "Open Folder in File Explorer"



הוספת תיקייה bin וחלק מההגדרות המתוארות בהמשך נעשות על מנת לרכז את כל קבצי הריצה במרוכז במקום אחד בתיקייה הראשית של Solution ולמנוע חלק מהבעיות ומקושי של מעקב אחר קבצי ריצה בהמשך הדרך.

3. הוספת שני פרויקטים נוספים ל solution:
עכשיו יש להוסיף ל-Solution שני פרויקטים נוספים (במהלך הוספה יש לוודא שהפרויקטים נפתחים עם Framework בגרסה NET 6.0. לפחות).

ב Visual Studio עמדו עם עכבר ימני על ה solution ואז **Add → New Project**
הוספת פרויקט **DalFacade** ופרויקט **DalList** מסוג **Class Library**:



4. קינפוג הפרויקטים
עבור כל אחד משלושת הפרויקטים שהוספנו, יש לבצע את השלבים הבאים:
לחיצה כפולה על שם הפרויקט ב-Solution Explorer,
יפתח לעריכה קובץ ניהול הפרויקט עם סיומת csproj,
יש להוסיף לו את השורות הבאות (המודגשות בצהוב) בסוף האלמנט **PropertyGroup**:

עבור פרויקט **DalFacade**

```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <TargetFramework>net7.0</TargetFramework>
    <ImplicitUsings>enable</ImplicitUsings>
    <Nullable>enable</Nullable>
    <BaseOutputPath>$(SolutionDir)\bin\</BaseOutputPath>
    <OutputPath>$(SolutionDir)\bin\</OutputPath>
    <AppendTargetFrameworkToOutputPath>>false</AppendTargetFrameworkToOutputPath>
    <AppendRuntimeIdentifierToOutputPath>>false</AppendRuntimeIdentifierToOutputPath>
    <UseCommonOutputDirectory>true</UseCommonOutputDirectory>
    <RootNamespace />
  </PropertyGroup>
</Project>
```

נרצה לוודא שהטקסט שהוספנו בקובץ המאפיינים השפיע על המאפיינים. לכן, בתפריט ההקשר של הפרויקט (לחיצה על כפתור ימני כאשר הסמן מעל שם הפרויקט) בחרו מאפיינים (**Properties**), ובחלון המאפיינים שנפתח:

א. תחת **General → Application**, וודאו שהשדה **Default namespace** הוא **יק**.
ב. תחת **Output → Build**, וודאו שהשדה **Base output path**, מכיל: **\$(SolutionDir)\bin**.
יתר ההוספות שעשינו לעיל לא משתקפות בממשק המאפיינים של הפרויקט ולכן עשינו אותם ע"י עריכה ישירה בתוך הקובץ csproj

עבור פרויקט **DalList**

```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <TargetFramework>net7.0</TargetFramework>
    <ImplicitUsings>enable</ImplicitUsings>
    <Nullable>enable</Nullable>
    <BaseOutputPath>$(SolutionDir)\bin\</BaseOutputPath>
    <OutputPath>$(SolutionDir)\bin\</OutputPath>
    <AppendTargetFrameworkToOutputPath>>false</AppendTargetFrameworkToOutputPath>
    <AppendRuntimeIdentifierToOutputPath>>false</AppendRuntimeIdentifierToOutputPath>
```

```
<UseCommonOutputDirectory>true</UseCommonOutputDirectory>
<RootNamespace>Dal</RootNamespace>
</PropertyGroup>
</Project>
```

נרצה לוודא שהטקסט שהוספנו בקובץ המאפיינים השפיע על המאפיינים. לכן, בתפריט ההקשר של הפרויקט (לחיצה על כפתור ימני כאשר הסמן מעל שם הפרויקט) בחרו מאפיינים (Properties), ובחלון המאפיינים שנפתח:

- תחת **Application → General**, וודאו שהשדה **Default namespace** הוא DAL.
 - תחת **Build → Output**, וודאו שהשדה **Base output path**, מכיל: \$(SolutionDir)\bin\
- יתר ההוספות שעשינו לעיל לא משתקפות בממשק המאפיינים של הפרויקט ולכן עשינו אותם ע"י עריכה ישירה בתוך הקובץ csproj

עבור פרויקט DalTest

```
<Project Sdk="Microsoft.NET.Sdk">
  <PropertyGroup>
    <OutputType>Exe</OutputType>
    <TargetFramework>net7.0</TargetFramework>
    <ImplicitUsings>enable</ImplicitUsings>
    <Nullable>enable</Nullable>
    <BaseOutputPath>$(SolutionDir)\bin\</BaseOutputPath>
    <OutputPath>$(SolutionDir)\bin\</OutputPath>
    <AppendTargetFrameworkToOutputPath>false</AppendTargetFrameworkToOutputPath>
    <AppendRuntimeIdentifierToOutputPath>false</AppendRuntimeIdentifierToOutputPath>
    <UseCommonOutputDirectory>true</UseCommonOutputDirectory>
  </PropertyGroup>
</Project>
```

נרצה לוודא שהטקסט שהוספנו בקובץ המאפיינים השפיע על המאפיינים. לכן, בתפריט ההקשר של הפרויקט (לחיצה על כפתור ימני כאשר הסמן מעל שם הפרויקט) בחרו מאפיינים (Properties), ובחלון המאפיינים שנפתח תחת **Build → Output**, וודאו שהשדה **Base output path**, מכיל: \$(SolutionDir)\bin\

יתר ההוספות שעשינו לעיל לא משתקפות בממשק המאפיינים של הפרויקט ולכן עשינו אותם ע"י עריכה ישירה בתוך הקובץ csproj

5. קישור בין הפרויקטים

- נרצה שפרויקט DalList יוכל לגשת בקוד לישויות ולמחלקות הציבוריות שיוגדרו בפרויקט DalFacade ולכן ניצור הפניה (reference) ממנו אליו:
עמדו עם עכבר ימני על הפרויקט DalList ואז Project Reference
בחלון שיפתח סמנו את הפרויקט DalFacade, ואז לחצו OK
ודאו בקובץ המאפיינים של הפרויקט שהטקסט שלו התעדכן לאחר הוספת הרפרנס:

```
<ItemGroup>
  <ProjectReference Include="..\DalFacade\DalFacade.csproj"/>
</ItemGroup>
```

חשוב שתבינו את המשמעות של רפרנס בין פרויקט לפרויקט:
פרויקט DalList מכיר את ההגדרות שנמצאות ב-DalFacade ולא ההיפך!
וכן, שכאשר נבנה/נקמפל את פרויקט DalList אז באופן אוטומטי הוא קודם כל יבנה/יקמפל את הפרויקט שהוא תלוי בו (DalFacade).

- נרצה שפרויקט DalTest יוכל לגשת בקוד לישויות ולמחלקות הציבוריות שיוגדרו בפרויקט DalFacade וכן יוכל לגשת לממשקים הציבוריים שמוגדרים ב-DalList, לכן ניצור הפניה (reference) ממנו אליהם:
עמדו עם עכבר ימני על הפרויקט DalList ואז Project Reference
בחלון שיפתח סמנו את 2 הפרויקטים DalList, DalFacade, ואז לחצו OK
ודאו בקובץ המאפיינים של הפרויקט שהטקסט שלו התעדכן לאחר הוספת הרפרנס:

```
<ItemGroup>
  <ProjectReference Include="..\DalFacade\DalFacade.csproj"/>
  <ProjectReference Include="..\DalList\DalList.csproj"/>
</ItemGroup>
```

הסבר כללי - 3 הפרויקטים של שכבת ה-DAL

1. פרויקט בשם **DalFacade** - חזית (facade) הנתונים. הפרויקט יכול:
 - a. הגדרות של הישויות הקיימות, הגדרה נפרדת לכל ישות בקובץ נפרד.
 - b. הגדרות של המתודות שניתן לבצע על הנתונים. נגדיר את אוסף המתודות בצורה מסודרת, כממשק (Interface).
2. פרויקט בשם **DalList** - בסיס הנתונים, שכבת הנתונים. הפרויקט יכול עבור כל ישות שמוגדרת ב-DalFacade:
 - a. אוסף הנתונים שיש מכל ישות (כל אוסף שמור בתור רשימות בזיכרון הפנימי מסוג **List**). בשלב זה הנתונים יהיו שמורים ברשימות (Lists) בתוך הפרויקט DalList. בשלבים הבאים הנתונים יהיו שמורים בזיכרון חיצוני (קבצי XML), בפרויקט נפרד בשם DalXml.
 - b. מחלקה שתממש את המתודות שניתן לבצע על כל ישות ע"י פנייה לאוסף נתונים מסוג **List**, של כל ישות. מימוש הממשק שמוגדר עבור אותה ישות ב-DalFacade.
3. פרויקט בשם **DalTest** - פרויקט שיכיל תוכנית ראשית שמטרתה:
 - a. לאתחל את בסיס הנתונים השמור בפרויקט DalList (תמלא את כל רשימות הישויות בנתונים)
 - b. לבדוק את תקינות כל המתודות שניתן לבצע על הנתונים השמורים ברשימות, דרך מהממשקים המוגדרים בפרויקט DalFacade.
 - c. כשנריץ את שלב 1, בעצם נריץ את הפרויקט DalTest והוא יתקשר עם שני הפרויקטים של שכבת הנתונים (DalFacade, DalList).

בהצלחה רבה!