

Typescript

מבוא

הגדרות:

- שפה שהינה Strongly Typed – שפה שמכילה טיפוסים נתונים עבור המשתנים שלה, לדוגמה C#, Java ועוד.
- שפה שהינה Untyped – שפה שאין בה טיפוסים נתונים עבור המשתנים, או שיש טיפוס כללי אחד. לדוגמה: JavaScript, PHP ועוד.
- קומפיילר- מגנון שהופך את ה-Source Code לתוכנית הסופית. לדוגמה, לקובץ הרצה.
- הסבר על סוגי השגיאות שקיימות במערכת:
- שגיאת קומפילציה (כל שגיאה שמנגנון הקומפיילר מגלה, לדוגמה שגיאת תחביר)
- שגיאת ריצה (כל שגיאה שגורמת לקריסת התוכנית במהלך הריצה, לדוגמה נסיון לקרוא מקובץ שלא קיים)
- שגיאה לוגית (כל שגיאה חישובית שלא גרמה לקריסת התוכנית, לדוגמה חישוב ממוצע של 100 תלמידים ע"י חלוקת הסכום ב-99 ולא ב-100)
- שפה שהינה Strongly Typed טובה בהרבה משפה שהינה Untyped בגלל מספר סיבות:
 - שגיאות עוברות לשלב הקומפילציה במקום לשלב הריצה
 - הגנה על המתכנתים
 - Intellisense (הגדרה: חלונית ההצעות שסביבת הפיתוח נותנת)
 - ועוד

קיצורים נפוצים:

- TS - type script
- JS - java script
- VS - visual studio

מסקנה:

- JS (שהינה) קשה מאוד לפיתוח ו-TS מכילה Types וקלה יותר לפיתוח.

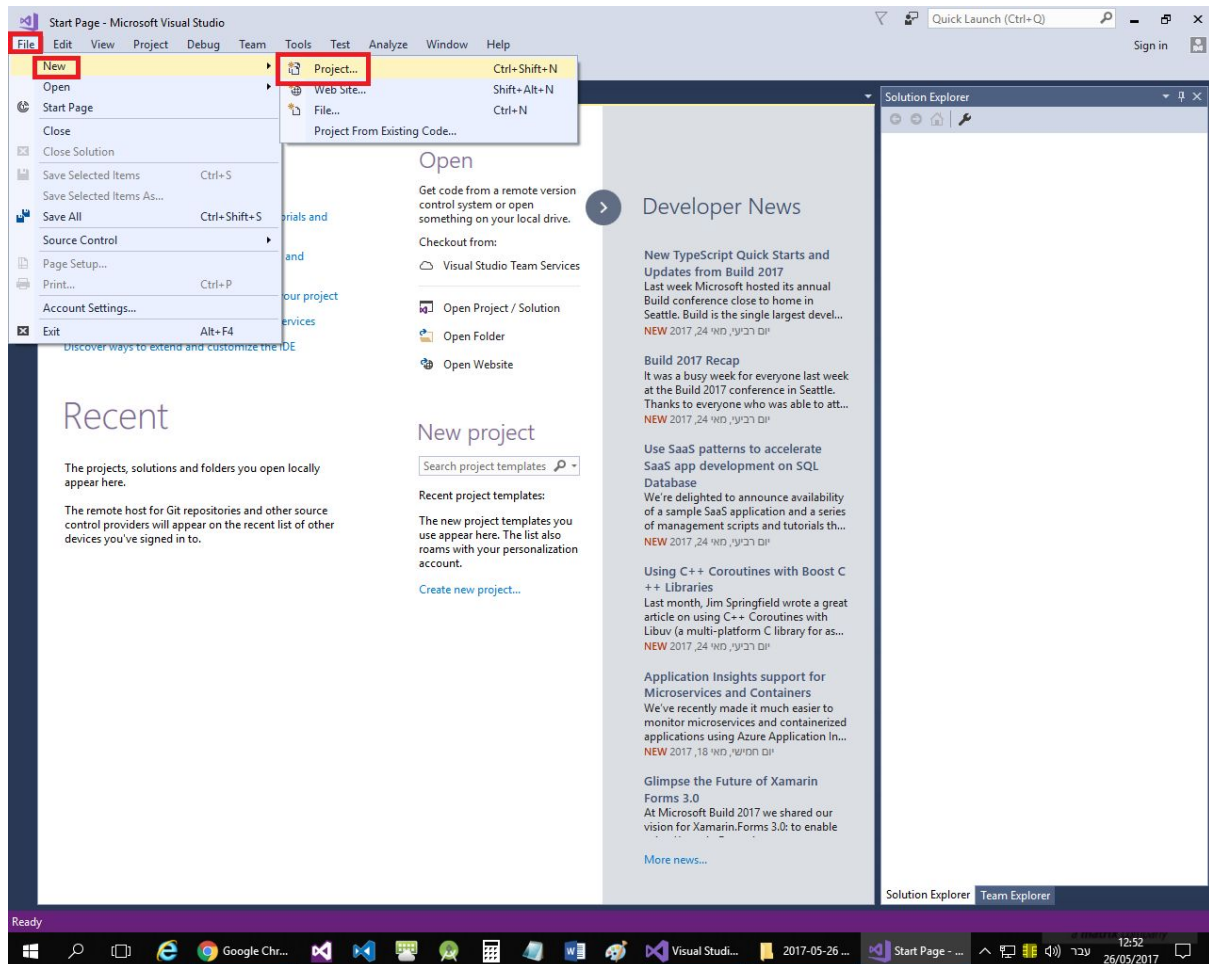
הערה:

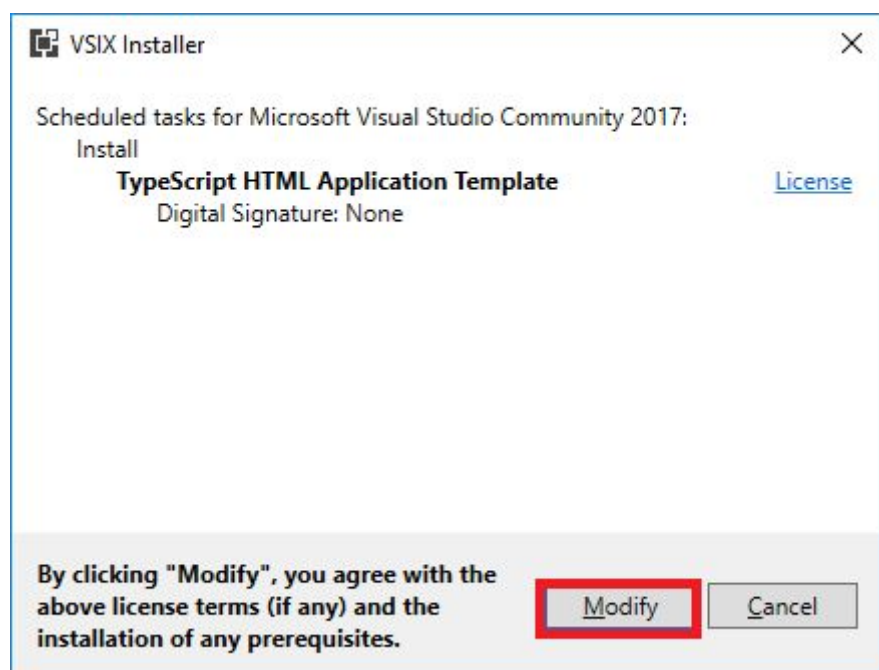
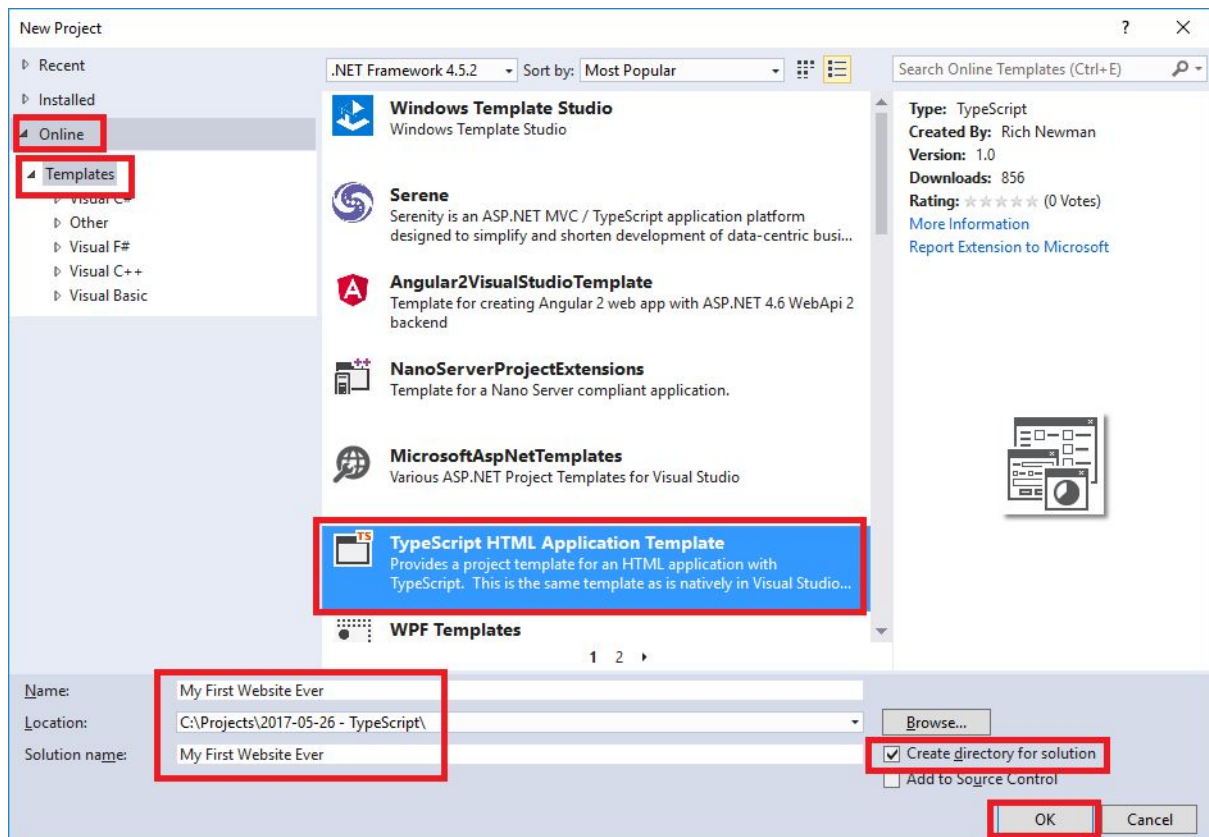
- TS הופכת ל-JS בצד השרת.

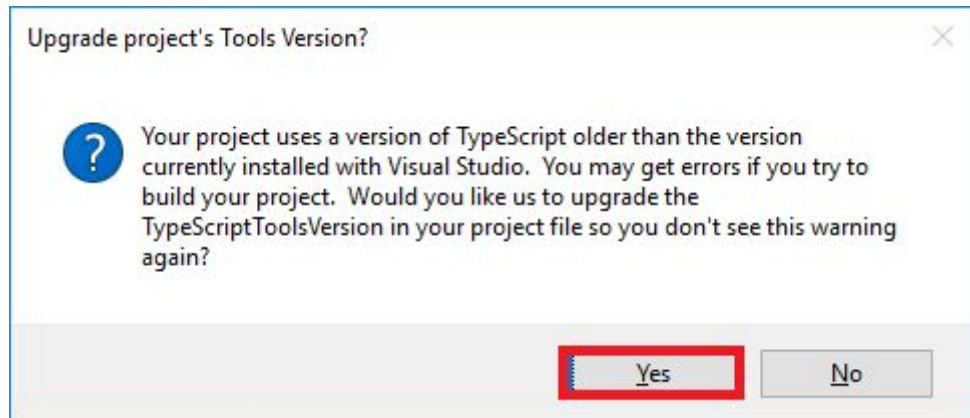
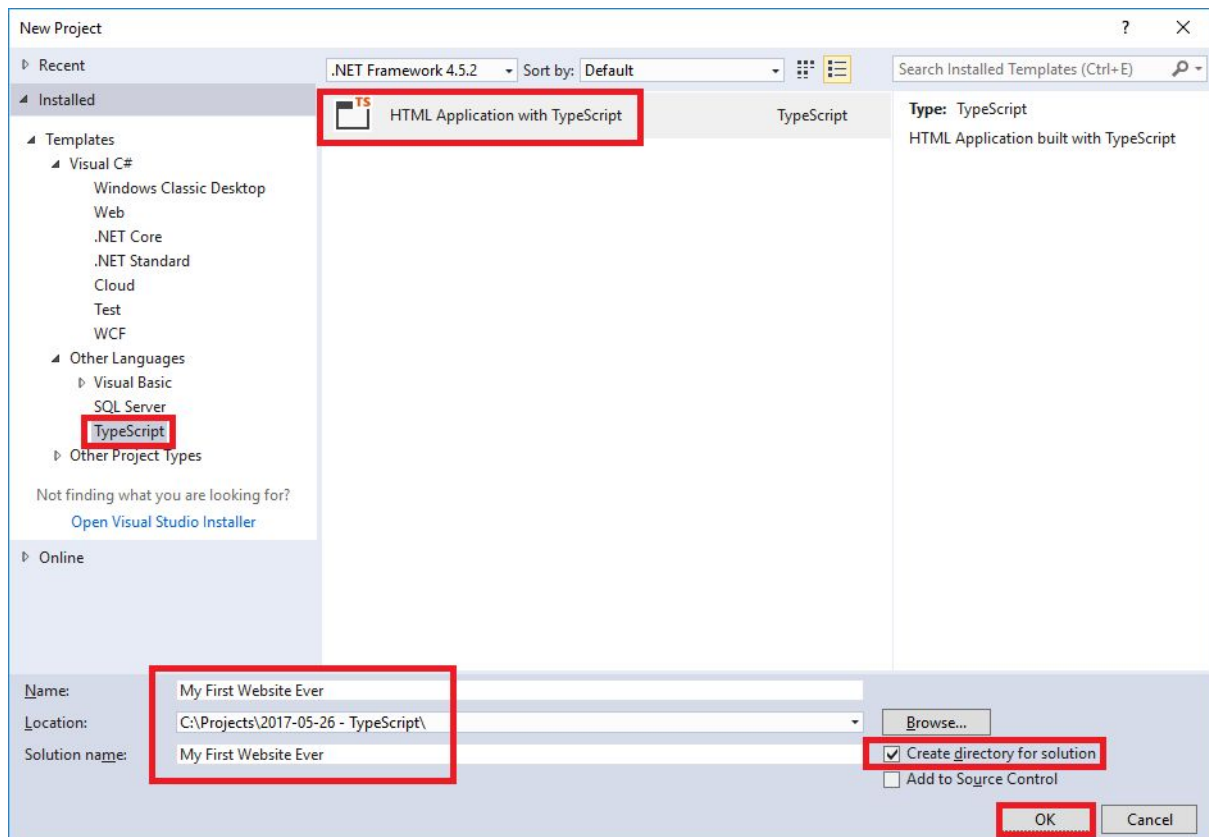
גישה לקובץ JavaScript בפרוייקט TypeScript

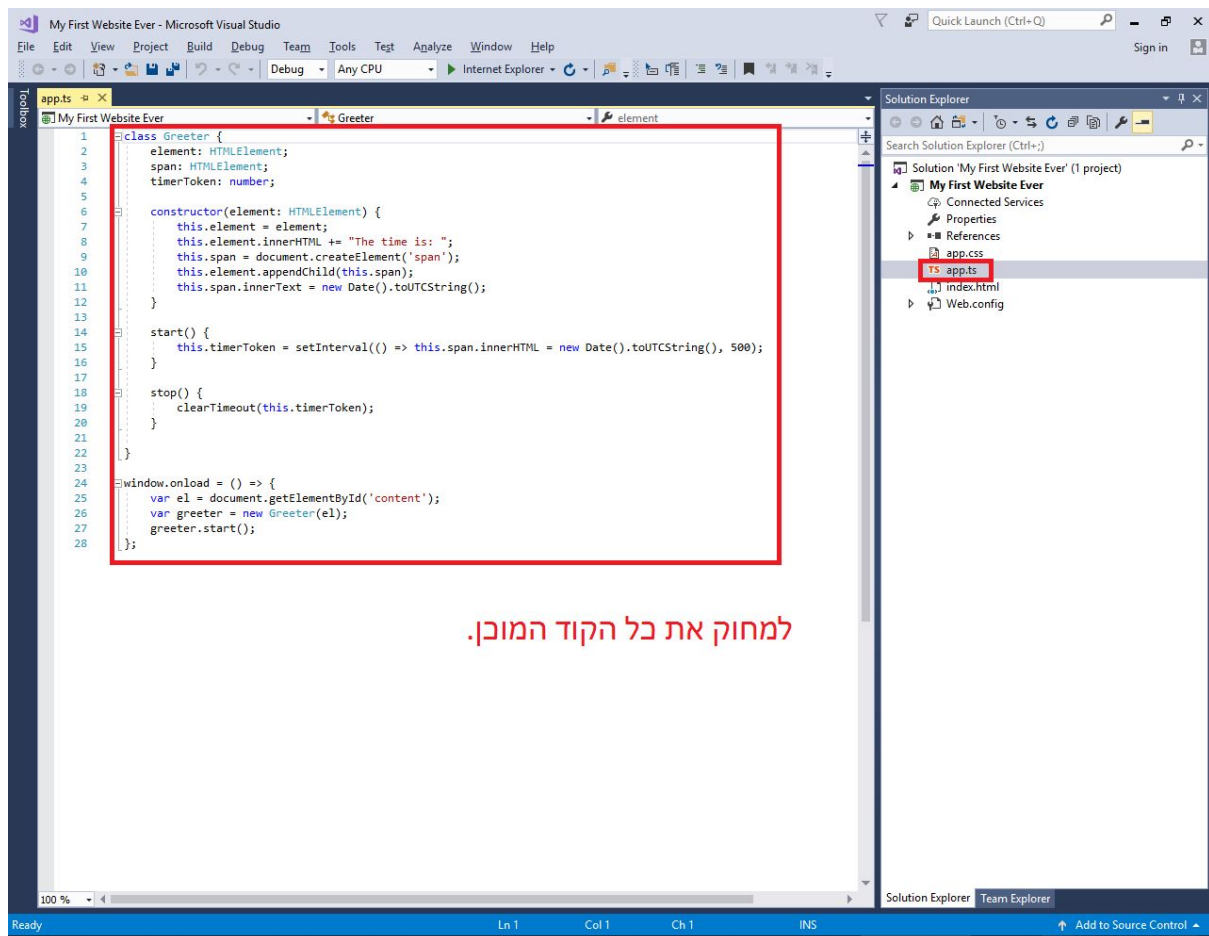
פתיחת פרויקט TypeScript

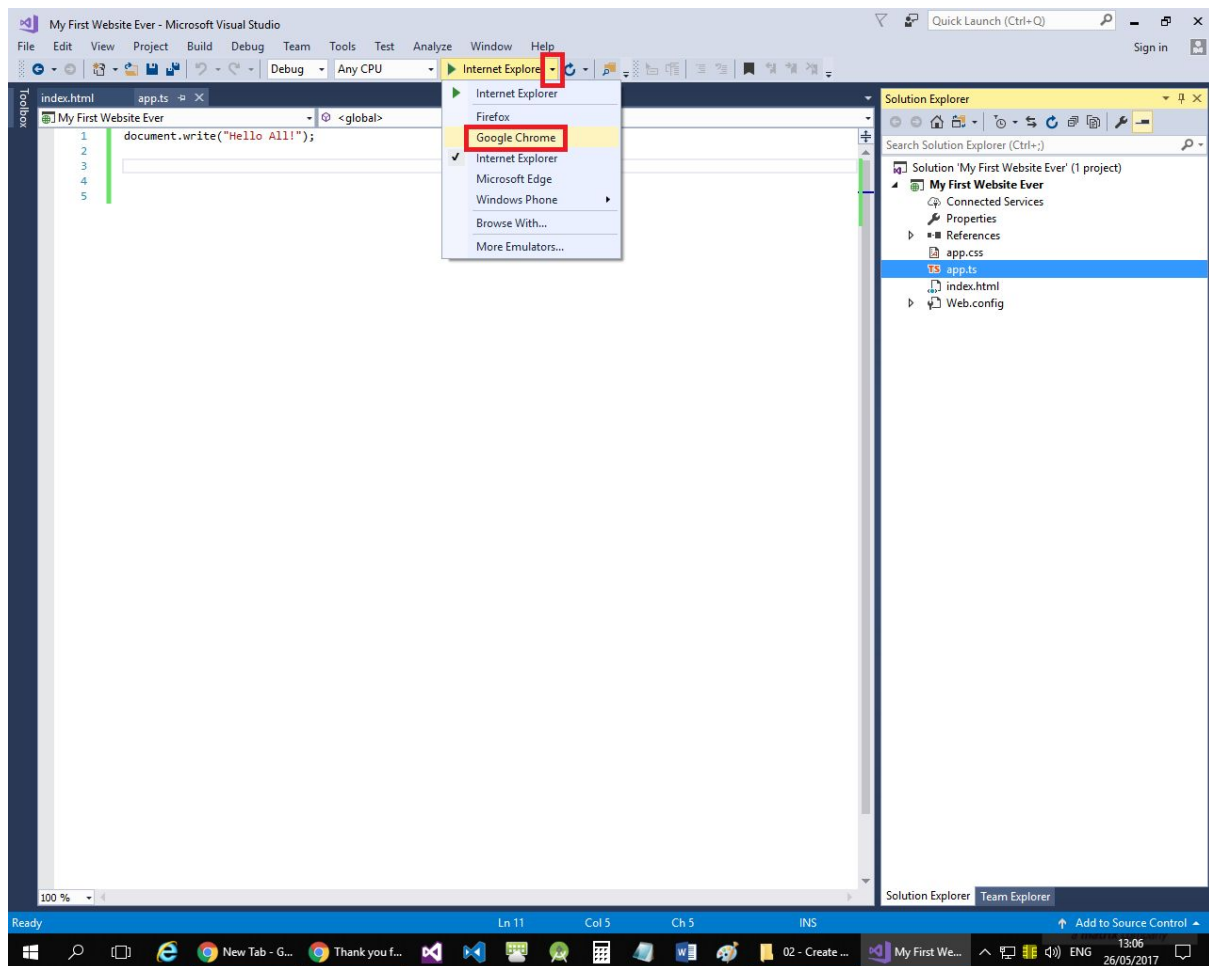
- צילום תמונה או להראות את המסמך שנמצא בתיקיה ברשת.
- יש להיפטר מכל הקוד שקיים ב-app.ts
- יש למחוק את שתי השורות שקיימות ב-body שב-HTML
- יש להריץ ולראות שהאתר רץ כאתר ריק.





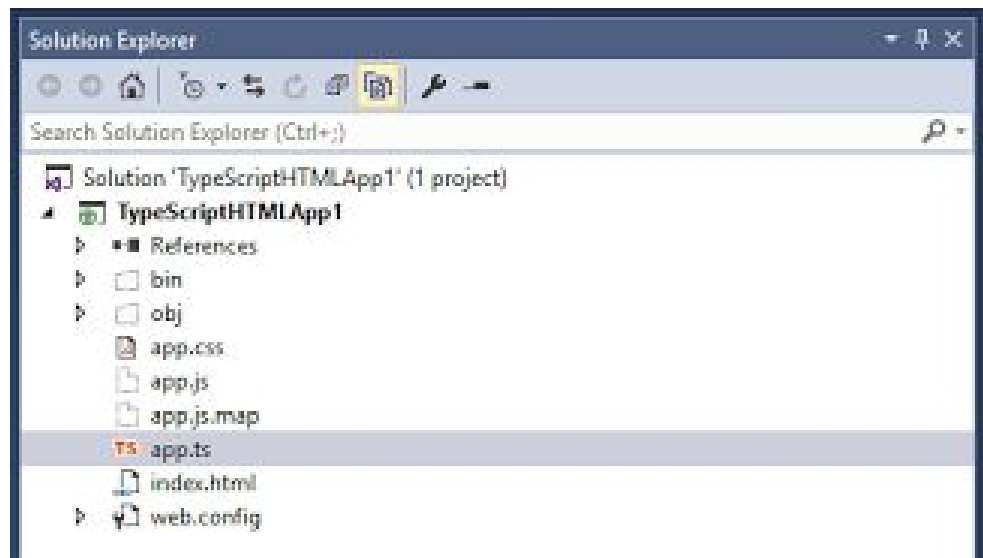




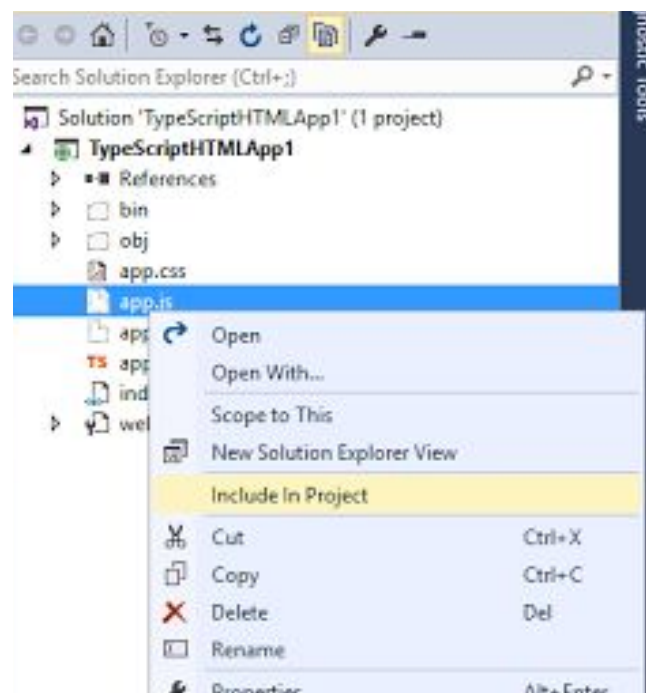


כאשר ניצור פרוייקט חדש מסוג TS ב- visual studio, נראה בצד את כל הקבצים שנוצרו. אבל, כאשר נרצה לראות את הקובץ js שנוצר באופן אוטומטי בתהליך הקומפילציה בהתאם לתוכן שאנו נרשום בתוך קובץ type script, לא נמצא אותו.

הדרך להציג את קובץ ה-js תהיה ע"י לחיצה על אופציית ה"Show All Files" שבחלון ה-solution explorer, ואז נראה את כל התיקיות / הקבצים המוסתרים, מקווקוים.



נוכל לבחור את קובץ הts, ללחוץ על לחצן ימני, ולבחור את האופציה "Include In Project", ואז הקובץ יתווסף להיות כמו כל קובץ אחר שמוצג בפרוייקט.

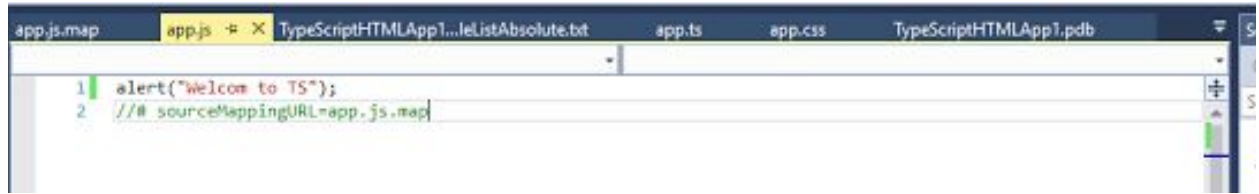


עכשיו נרשום בתוך קובץ ts את הקוד הבא:



לאחר מכן, נלחץ על **control + F5** והתוכנית שלנו תרוץ בדפדפן.

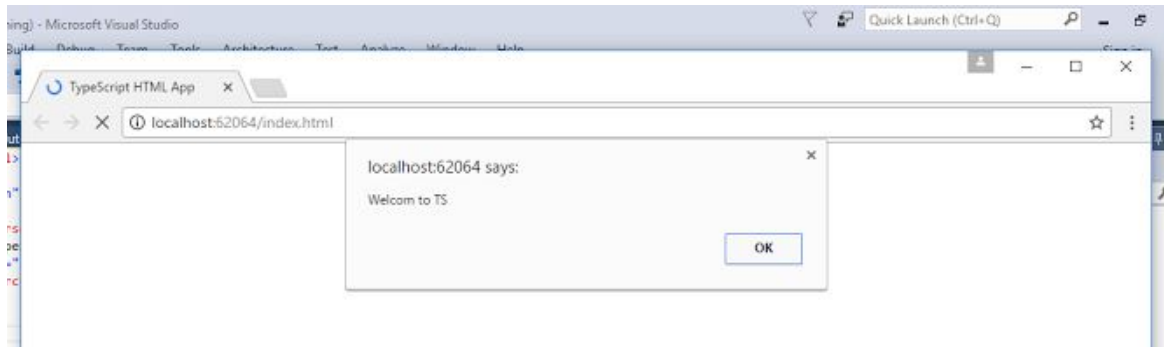
נחזור לקובץ ה-js ונראה שהתוכן שלו התעדכן ל:



```
1 alert('Welcom to TS');
2 //# sourceMappingURL=app.js.map
```

גישה לקובץ JavaScript מתוך הדפדפן

כאשר דפדפן chrome מריץ את הסקריפט שכתבנו, הדף אותו אנו מריצים יראה כך:



נוכל ללחוץ על מקש F12 ולהגיע לעמוד של כלי המפתחים. ניגש לטאב של "Application" ושם נוכל למצוא את הסקריפט שיצרנו

Elements Console Sources Network Performance Memory Application Security Audits

Application

- Manifest
- Service Workers
- Clear storage

Storage

- Local Storage
- Session Storage
- IndexedDB
- Web SQL
- Cookies

Cache

- Cache Storage
- Application Cache

Frames

- top
- Scripts
 - app.js**
- Stylesheets
 - index.html

```
1 alert("welcom to TS");
2 //# sourceMappingURL=app.js.map
```