**Definitions**

**Design Pattern – תבניות עיצוב**

מבנה קוד שנותן מענה לצורך ספציפי.

המבנה בנוי מפונקציות משתלובת שלהן, מרכיבים ועוד.

**Promise**

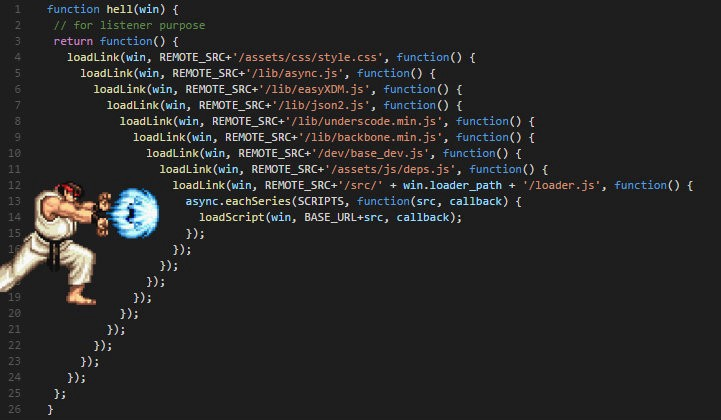
אובייקט JavaScript שיצא ב-ES6 המאפשר לבנות פונקציה אסינכרונית שמדווחת לנו הצלחה או כשלון.

מיועד להחליף את שני ה-Callbacks שתמיד היו שולחים לפונקציה – successCallback ו-errorCallback.

עבודה עם Promise מקל על הקריאה לפונקציה.

**Callback Hell**

מצב בו אנו שולחים callback לפונקציה, ובתוך קוד ה-callback אנו קוראים לפונקציה נוספת עם עוד callback שבתוכה אנו קוראים לפונקציה נוספת עם עוד callback וכו' וכו'... אנו רוצים להימנע ממצב כזה.



**async-await**

צמד מילים שמורות המייעלות מבחינת Design את נושא ה-Callback Hell.

async נכתבת לפני המילה function שאנו בונים, ומציינת שאנו הולכים להשתמש בטכניקת async-await.

await נכתבת לפני הקריאה לפונקציה שמחזירה Promise ומבצעת שלושה דברים:

1. ממתינה עד שפעילות ה-Promise שהתחילה תסתיים – או ע"י הצלחה, או ע"י כישלון.
2. לא תוקעת את פעילות הדפדפן כולו (או פעילות השרת כולו אם אנו ב-Node.js) – רק הפונקציה שאנו בונים כרגע ממתינה בפקודת ה-await, הדפדפן כולו או השרת כולו לא ממתינים אלא ממשיכים לספק פעילות לשאר האפליקציה.
3. מחזירה בחזרה את הערך שדווח ע"י ה-resolve במקום להחזיר את אובייקט ה-Promise!!!