OOP

Object Oriented Programming

תכנות מונחה אובייקטים. תכנות מונחה עצמים. עצם = אובייקט.

רוב הזמן אנחנו נעבוד עם דאטה (נתונים) ופעולות על הדאטה. שיטת ה-OOP מקלה עלינו בזאת.

בבסיס השיטה, הרעיון לייצר "תבניות" של אובייקטים, ואז ליצור את האובייקטים מתוך התבניות.

היחידה הבסיסית, כלומר ה"תבנית" נקראת Class.

Class לא במשמעות של כיתה....

אלא קיצור של המילה Classification שמשמעה "סיווג".

לפעמים מתרגמים את המילה Class בעברית למילה "מחלקה".

שם של קלאס ייכתב תמיד בסגנון UpperCamelCase

ולא upperCamelCase

ולא uppercamelcase

כמעט תמיד ניצור קלאס בקובץ נפרד, ולא מספר קלאסים באותו קובץ.

**Data Members**

אלו הם מאפייני הקלאס,

**Methods**

אלו הן פונקציות, המתארות את הפעולות שאפשר לעשות על האובייקטים. שימו לב שלא צריך לכתוב את המילה השמורה function מכיוון שפונקציות אלה הן מתודות. מתודה היא פעולה שקלאס יכול לעשות. בפועל, מייצגים את זה באמצעות פונקציות.

**Constructor**

הקונסטרקטור היא פונקציה (לא מתודה) מיוחדת המאפשרת לאתחל את האובייקט בנתונים.

הקונסטרקטור היא פונקציה שמתבצעת אך ורק פעם אחת מיד באיתחול אובייקט מתוך הקלאס, אפשר להגדיר לה חתימת פרמטרים באמצעותם ניתן לאתחל את הקלאס.

לא חובה לאתחל את כל הדטה ממברז דרך הקונסטרקטור, אבל נהוג שכן כי זה בדכ מה שצריך.

יש רק קונסטרקטור אחד לקלאס

**Readonly**

ניתן לסמן דטה ממבר כ-readonly ואז לא ניתן לשנות אותו מבחוץ. ניתן לאתחל משתנה readonly או בזמן ההגדרה או מתוך הקונסטרקטור. אך לא דרך שום שיטה אחרת

Access Modifiers

כמפתח הקלאס, אני מקבל יכולת להגדיר עבור כל אחד ואחד מהממברים ומהמתודות, מה תהיה רמת הנגישות אליהם.

ישנן 3 רמות נגישות:

1. Public – גישה פתוחה לכל אחד. ניתן לדעת על קיום של דטה ממברס ופונקציות וניתן גם לגשת אליהם (ואפילו לשנות אם הם לא מוגדרים readonly)
2. Protected
3. Private – הגישה לנתונים ולמתודות תהיה פתוחה אך ורק לאובייקטים שנוצרו מהקלאס עצמו.