

שיעורי בית

1. מהו תהליך ה- **Serialization**? מדוע צריך אותו?
2. מהו תהליך ה- **Deserialization**? מדוע צריך אותו?
3. מהו פורמט **XML**? מה היתרון של פורמט **XML**?
4. מהו **Stream**? מהו **FileStream**?
5. מהו **using**? כיצד הוא עוזר לנו?
6. האם ניתן לעשות **Serialization** למערך?
7. מה היתרון של **AddLink** לפרוייקט?
8. מהי תגית? **[Attribute]**
9. מה נותנת תגית **XmlRoot**?
10. מה נותנת תגית **XmlElement**?
11. מה נותנת תגית **XmlIgnore**?
12. מה נותנת תגית **XmlAttribute**?
13. האם כל שדות המחלקה עוברים **Serialization**? מי כן? מי לא?
14. מה צריכה להיות רמת נגישות של מחלקה אשר רוצים לעשות לה **Serialization**?
מדוע?
15. איזה בנאי חייבים שיהיה במחלקה כדי לתמוך ב **Serialization**? מדוע לדעתך? מה צריכה להיות רמת נגישות?

Car
+ Model {get; set;} + Brand {get; set;} + Year {get; set;} + Color {get; set;} - codan : int # numberOfSeats : int
+ ctor() + ctor(fileName) + ctor(model, brand, year, color, codan, numberOfSeats) + GetCodan() : int + GetNumberOfSeats() : int # ChangeNumberOfSeats(int) : void + static SerializeACar(filename, car) : void + static SerializeCarArray(filename, Car[]) + static DeserializeACar(filename) : Car + static DeserializeCarArray(filename) : Car[] + CarCompare(filename) : bool

- ממש את הדיאגרמה בצורה הנכונה ביותר. הוסף ToString()
- **שלב א':** ממש את כל הפונקציות ללא הפונקציות הסטטיות וללא ctor(fileName) וללא CarCompare
- ייצר ב Main מכונית וכתוב אותה לקובץ (בפורמט XML כמובן). קרא את המכונית מהקובץ ובדוק שבאמת קיבלת את אותם הערכים.
- כעת שנה את תוכן קובץ ה XML והרץ את התוכנית שוב (שים בהערה את קטע הקוד שכתב לקובץ כמובן, כדי לא למחוק את השינוי שביצעת). האם קיבלת אובייקט כמו שכתוב בקובץ?
- ייצר מערך של מכוניות וכתוב את המערך כולו לקובץ בפעולה אחת (רמז: ראה דוגמא מהשיעור) לאחר מכן קרא את המערך כולו
- ***אתגר: שלב ב':**
- פונקציית ctor(filename) מקבלת שם של קובץ כפרמטר, ומייצרת את המופע על פי המידע שקיים בקובץ.
- **כלומר:** בתוך בנאי זה, יש ליפתוח את הקובץ לקריאה ולעשות deserialization. לאחר מכן לקחת את כל הערכים של המכונית שנקראה מהקובץ (מודל, צבע וכו') ולשים אותה בתוך שדות המידע של this.
- פונקציית SerializeACar מקבלת שם של קובץ + מכונית, ומבצעת Serialization של המכונית לתוך הקובץ שהתקבל כפרמטר
- פונקציית SerializeCarArray מקבלת שם של קובץ + מערך מכוניות, ומבצעת Serialization של המערך לתוך הקובץ שהתקבל כפרמטר
- פונקציית DeserializeACar מקבלת שם של קובץ, ומבצעת Deserialization מתוך הקובץ שהתקבל כפרמטר ומחזירה את המכונית שנוצרה בפונקציה
- פונקציית DeserializeCarArray מקבלת שם של קובץ, ומבצעת Deserialization מתוך הקובץ שהתקבל כפרמטר ומחזירה את מערך המכוניות שנוצר בפונקציה
- פונקציית CarCompare מקבלת שם של קובץ, ומבצעת Deserialization מתוך הקובץ שהתקבל כפרמטר. לאחר מכן משווה את כל השדות של המכונית שהתקבלה מהקובץ, מול השדות של המחלקה. אם כל השדות זהים יוחזר אמת אחרת יוחזר שקר (כמובן רק מה ש public)
- כעת שנה את התוכנית שכתבת בשלב א' כדי שתשתמש בפונקציות הסטטיות
- ייצר מכונית תוך כדי קריאה לבנאי המקבל שם של קובץ
- ייצר מכונית שמור אותה בקובץ. קרא ל CarCompare ובדוק שחזר אמת. כעת שנה את ערך אחד השדות של האובייקט וקרא שוב ל CarCompare. האם חזר שקר?

בהצלחה