



Basic SW Design Workshop

יום אחד

1. מבוא

האם הלחץ בפרויקט לא מאפשר לך לצאת לקורס ארוך? האם אתה בכל זאת רוצה לקבל מושג על מהו עיצוב טוב ואיך Design Patterns יכולים לעזור לזרז ולשפר את תהליך פיתוח התכנה?

אם כן, סדנא זו היא בשבילך! בסדנא נלמד:

- העקרונות הבסיסיים של תכן מונחה עצמים.
- עקרונות עיצוב שהופכים מערכת לגמישה ויציבה.
- Design Patterns בסיסיים ושילובם בתכן על מנת לאתחל וליצור אפליקציה שלמה.

2. תוכן הסדנא

<p>Abstract Factory Pattern • Singleton Pattern • Composite Structure Diagram •</p> <p>Design Patterns למימוש ממשק אדם</p> <p>בחלק זה אנו נלמד תבניות עיצוב שיכולות לעזור בבניית ממשק אדם.</p> <p>Strategy Pattern • Command Pattern • Observer Pattern • Model View Controller (MVC) Pattern •</p> <p>Design Patterns למימוש ממשק מכונה</p> <p>בחלק זה אנו נלמד תבניות עיצוב שיכולות לעזור בבניית ממשק לחומרה או למערכות חיצוניות.</p> <p>Proxy Pattern •</p>	<p>עקרונות בסיסיים של תכן מונחה עצמים</p> <p>בחלק זה נדון על המאפיינים הבסיסיים ביותר של עיצוב מונחה עצמים.</p> <ul style="list-style-type: none"> • עקרון הקשר הלא רצוי. • עקרון האחרייות והאצלת האחרייות / סמכות. <p>עקרונות עיצוב (Design Principles)</p> <p>בחלק זה אנו נלמד עקרונות עיצוב בסיסיים הנחוצים כדי להגיע לעיצוב תוכנה שמפחית את כמות ה-Unit Test שחייבים להריץ בכל אינטגרציה.</p> <ul style="list-style-type: none"> • Open Close Principle • Liskov Substitution Principle • Dependency Inversion Principle • Interface Segregation Principle <p>Design Patterns בסיסיים לאתחול מערכת</p> <p>בחלק זה אנו נלמד איך למדל מערכת בעזרת Components, תבניות עיצוב בסיסיות ואיך למדל אתחול מערכת.</p>
---	--

3. קהל יעד ותנאי קדם

מנהלי פרויקטים, ראשי צוותים ואנשי פיתוח שיש להם ידע בסיסי ב-UML וניסיון בכתיבת קוד ++C, Java או #C. הרוצים לשפר את יכולת העיצוב של מערכות מונחות עצמים.

הערות:

- סדנא זו אינה רק סדנא תאורטית. במהלך הסדנא המשתתפים יתרגלו את הנלמד באופן מעשי על ידי בניית מודל UML.