**תשובות לשיעורי בית 0506**

**1. Resume -** עוצר את העובד עד שמישהו אחר יעשה לו Resume... היתרון שזה מקל על מהירות הפעולה באותו רגע במקום לעשות Start והם נקראים **Deprecated**(פונקציות ישנות) בגלל שהם לוקחים איתם את המפתח ויכול ליצור **DeadLock**

**2. Wait -** לוקח את העובד ששלח את הפונקציה להמתנה עד שעובד אחר יעשה לו Pulse עם אותו מפתח או **PulseAll** שמשחרר את כולם.

**3. RaceCondition –** תחרות בין עובדים.. לדוגמא כשעושים **PulseAll** יש תחרות בין העובדים מי מגיע ראשון למפתח כי זה אף פעם לא מדויק(למרות שבדרך כלל הראשון שנכנס הוא הראשון שיוצא).

**4. DeadLock –** בכל מקרה כאשר עובד תופס מפתח ולא יכול לשחרר אותו כי הוא תפוס עם מפתח אחר ומחכה שישחררו אותו (קורה הרבה כאשר שני עובדים נשלחים לפונקציה של עובד אחר ושניהם תופסים את המפתח של הפונקציה שהעובד השני נשלח אליו).

**5. א.** שזה יהיה ב**Critical Section**

**ב.** שזה יהיה עם אותו מפתח

**6. Design Patterns –** כ25 דוגמאות לפתירות בעיות נפוצות והם הומצאו ע"י קבוצת אנשים שנקראת **Gang Of Four**

**7.** כאשר רוצים לייצר מופע ממחלקה שיופיע רק פעם אחת והמופע יהיה נגיש לכולם (ושיהיה (ThreadSafe

**8. א.** שדה מידע Private מסוג של המחלקה(בדרך כלל נקרא instance) שבהתחלה הוא Null וכאשר מפעילים את הפונקציה  **GetInstance (פעולה ד')** אנו שולחים New ומכניסים אותו לשדה מידע.

**ב.** מפתח מסוג object

**ג.** בנאי **Private** או \***Protected**.

**ד.** פונקציה Public שמחזירה את instance שבדרך כלל נקראת **GetInstance** (צריך לבדוק שהשדה מידע הוא עדיין Null לפני הLock ואחריו כדי שלא יכביד על המערכת ויחכו הרבה עובדים למפתח רק כדי לבדוק אם instance == Null)

**\*** היתרון ב**Protected** הואשרק כך יהיה אפשר לרשת מאותה מחלקה. לדוגמא- אם ארצה שיהיו עוד מחלקות שיורשות מה**Singleton** שלי חייב שהבנאי שלי יהיה Protected כי היורשים תמיד ניגשים לבנאי של האבא לפני שהם ניגשים לבנאי של עצמם ואם הבנאי הוא על Private אז גם היורשים לא יכולים לגשת אליו (בנאי זה פונקציה, ואי אפשר לגשת לפונקציה Private חוץ מהמחלקה עצמה)