**עקרון הכימוס** – לפי עקרון הכימוס, כל מחלקה מסתירה את תכונותיה ודרכי הפעולה הפנימיות שלה, ומספקת מעין מעטפת חיצונית עבור המשתמש. הכימוס מאפשר למשתמש לעשות שימוש באותה מחלקה או שירות, מבלי לדעת מה מתרחש בפנים, שכן גם אינו רלוונטי לאותו משתמש (כך למשל API). כמו כן, זה מאפשר גמישות מסוימת בקוד, כי כאשר יהיה צורך להחליף את אותו שירות ספציפי, או לעדכן את הקיים, פשוט ניתן להחליף את המחלקה במחלקה אחרת.

**הבדל בין מחלקה למחלקה אבסטרקטית** – מחלקה רגילה מגדירה אובייקט. באמצעות המתודות והתכונות המוגדרות בה ניתן ליצור Instance של אותו אובייקט. מחלקות יכולות לרשת תכונות ומתודות אחת מהשנייה. עם זאת, לפעמים אין באמת צורך להגדיר מחלקה בסיסית כמחלקה רגילה, שכן אין הגיון ליצור Instance לאותה מחלקה בסיסית (כך למשל, מחלקה בסיסית של כלי רכב). לכן, יש צורך במחלקה אבסטרקטית, שכוללת את המתודות והתכונות אך לא ממומשת. המחלקות היורשות מהמחלקה האבסטרקטית יצטרכו לממש את המתודות האבסטרקטיות.

**הבדל בין מחלקה למחלקה סטטית** – מחלקה רגילה מגדירה תכונות ומתודות, שאיתן ניתן ליצור Instance. לעומת זאת, מחלקה סטטית, כוללת מתודות ותכונות סטטיות, אשר אינן שייכות לאובייקט, אלא למחלקה עצמה. דוגמה למחלקה סטטית בc# היא מחלקת Math. מחלקות סטטיות נותנות לנו שירות באמצעות המתודות שלה, אך לא ניתן ליצור אובייקטים. מחלקות סטטיות חשובות ביותר אם אנחנו רוצים ליצור utils כלשהו.

**מי יכול לגשת לmembers בעלי access modifier של internal** – מי שיכול לגשת לinternal members זה רק בתוך אותו assembly. כלומר, רק מה שנמצא באותו פרויקט, לכן אם יש שני פרויקטים ואחת המחלקות של פרויקט 1 היא internal, היא תהיה נגישה רק למחלקות אחרות בתוך פרויקט 1, ופרויקט 2 לא יוכל לגשת.