

1 - ירושה היא פונקציונאליות של שפות מונחות עצמים המאפשרת לנו לכתוב מחלקות אם, שמעבירות את המאפיינים והמתודות שלהן למחלקות אחרות - מחלקות שיורשות מהן. כלומר אם מחלקת האם היא מחלקת Object שמממשת מתודה כמו ToString, כל מחלקה שתורש ממחלקת Object תקבל את ToString בחינם עם מימוש בסיסי (נתון לדריסה).

הפונקציונליות הזאת עוזרת לנו להמנע משיכפול קוד - מספיק לכתוב את המאפיינים במחלקת האם וכל המחלקות היורשות יקבלו אותם, הקוד מודולרי וקל לתחזוק - אם אנחנו רוצים לשנות משהו מספיק לשנות אותו במחלקה אחת וכל המחלקות היורשות יקבלו את התיקון, הקוד קריא יותר וקל יותר להבנה.

2 - מחלקה אבסטרקטית היא מחלקה שבדרך כלל לא מכילה מימוש או מימוש מאוד מינימלי, היא מחלקה שאי אפשר לייצר מופע שלה וחייבים לרשת ממנה כדי לממש אותה - לכן היא חייבת מבחינה הגיונית להיות Public. היא יכולה להכיל פונקציות רגילות פונקציות אבסטרקטיות.

פונקציות אבסטרקטיות חייבות להימצא במחלקה אבסטרקטית, היא פונקציה חסרת מימוש - חתימה בלבד והיא חייבת להיות Public. כל מחלקה שיורשת ממחלקת האם אבסטרקטית חייבת לממש אותה. אנחנו נשתמש בפונקציה אבסטרקטית כשאנחנו רוצים לכפות על כל המחלקות היורשות מימוש. אם מחלקת האם (האבסטרקטית) היא בעל חיים, הגיוני שתהיה לה פונקציה movement, וכיוון שכל בעל חיים מתנייד אחרת אנחנו נכפה את המימוש על המחלקה היורשת, כלב יממש את הפונקציה בעזרת הליכה עד ריצה על 4 כשתוכי יממש את הפונקציה על ידי תעופה (ושימוש ב interface IFlyable לדוגמא).

3 - פולימורפיזם היא היכולת של מחלקה אחת לתפקד בתור מחלקה/ות אחרת/ות. כלומר כאשר משתמשים במחלקת האם כדי להתייחס למחלקת הבן (ברגע שמחלקה עומדת עומד ביחס is a כלפי פחות מחלקה אחת היא מחלקה פולימורפית). למשל אנחנו יכולים להשתמש במחלקת Animal (מחלקת האם) כדי להתייחס למחלקות Dog, Cat, Bird וכו כיוון שכל המחלקות האלה הן Animal (מממשות את יחס is a).