

תשובות לחלק התאורטי

1. מחלקה זה ה"מתכון למופע. במחלקה אנחנו כותבים את ההוראות – השדות והפונקציות ואת המופע אנחנו יוצרים לפי ההוראות האלה. למופע יש את השדות והפונקציות שכתובות במחלקה.
2. כי הבן מכיל בתוכו את הפונקציות והמשתנים מהאב
3. צורה ועיגול. עיגול יורש ממחלקה צורה. היתרון שקיבלנו משימוש בירושה הוא שהמחלקה עיגול מקבלת פונקציות ומשתנים מהאב צורה. צורה יכול להוריש את הפונקציות והמשתנים שלו למחלקות נוספות ובכך למנוע שיכפול קוד בכל אחד מהמחלקות האחרות. המחלקה צורה יכולה להיות בנויה בצורה כזאת שיהיו בה פונקציות ומשתנים כללים שמתאימים למימוש צורות שונות.
4. המשפט הזה לא נכון בגלל שירושה היא מאב לבן. הבן צריך להיות סוג של האב. ירושה יכולה לקרות בין רהיט לשולחן וכיסא כי כיסא ושולחן הם סוגים של רהיטים. שולחן הוא לא סוג של כיסא ולכן לא יכול לרשת ממנו.
5. כדי שנוכל לדעת שאילו שדות המידע של המחלקה ולא שדות מקומיים של פונקציה.
6. בנאי הוא פונקציה שמופעלת כאשר יוצרים את האובייקט. נשתמש בה כדי להגדיר דברים שאנו רוצים להגדיר עם יצירת האובייקט- כמו למשל להציב ערכים בשדות המידע של המחלקה.
7. בנאי ריק.
8. בנאי המקבל פרמטרים הוא פונקציה שאנו מגדירים בתוך המחלקה בעלת שם זהה לשם של המחלקה ושמקבלת פרמטרים object initializer זה דרך שבה אנחנו מציבים ערכים לשדות של המחלקה כאשר אנחנו יוצרים את המופע של האובייקט (אנחנו מוסיפים סוגריים מסולסלים שמסתיימים בנקודה פסיק ובתוך הסוגריים אנו מציבים ערכים לשדות של המחלקה)
האחד public void Shape (area)
והשני

```
Author mahesh = new Author()  
{  
    Name = "Mahesh Chand",  
    Book = "LINQ Programming",  
    publisher = "APress",  
    Year = 2013,  
    Price = 49.95 };
```

9. Overloading זה שכותבים פונקציות בעל שם זהה אך עם פרמטרים שונים.
Constructor Overloading זה שמשתמשים בבנאים עם פרמטרים שונים על מנת לתת את האפשרות לבנות את המופע של האובייקט בכל מיני דרכים.
10. מייצרת מחרוזת בהתאם לאובייקט שאליה הפעלנו אותה ולאיק שהגדרנו אותה אם הפעלנו אותה על מחלקה היא תחזיר את השם של המחלקה. אם הפעלנו אותה על מספר היא

- תחזיר את המספר בצורת מחרוזת. וכשאנו מגדירים מחלקה יש לנו אופציה להגדיר את הפונקציה לפי רצוננו.
11. Override שאנו ממשים מחד פונקציה קיימת. זה קורה כאשר בבן אנחנו רוצים לממש פונקציה שכתובה באב. TOSTRING?
12. הוא ידפיס את שם המחלקה.
13. מחלקה אבסטרקטית היא מחלקה שאי אפשר ליצור ממנה אובייקט. כדאי לתכנן מחלקה אבסטרקטית כאשר אנו רוצים לתכנן מחלקה כללית שאפשר יהיה להוריש את תכונותיה למספר מחלקות דוגמא: מחלקה צורה. מהמחלקה צורה אנו יכולים ליצור מחלקות רבות שירשו ממנה כמו: עיגול, ריבוע, מעוין.
14. פונקציה אבסטרקטית היא פונקציה לא ממומשת.
15. כאשר יצרנו פונקציה אבסטרקטית בתוך מחלקה המחלקה חייבת להיות אבסטרקטית וכל המחלקות שירשו ממנה חייבות לממש את הפונקציה אם הן לא אבסטרקטיות.
16. כן
17. לא
18. לא
19. כאשר אנו יוצרים מופע של מחלקה רגילה שירשת ממחלקה אבסטרקטית נוצר מופע של המחלקה האבסטרקטית.
20. שאנחנו מפעילים פונקציה ששייכת למחלקת הבסיס. חובה להשתמש במילה זאת כשאנו רוצים לקרוא לפונקציה האב מתוך פונקציה בעלת שם זהה בבן. וכאשר יש את הפונקציה באב ובבן ואנו רוצים לקרוא מהבן לפונקציה של האב.
21. כאשר אנחנו עושים override לפונקציה ToString אנחנו נרוויח מכך את המחרוזת שמומשה במחלקת האב ולא נצטרך לכתוב קוד שיוצר אותה שוב.
22. עוד מתודה או מופע שצריך לייצר.
23. ע"י חץ מהמחלקה היורשת למחלקה ממנה היא יורשת.
24. למטה המחלקה סלט פירות מכילה את המחלקה פירות
25. שפת ביניים. בין C# לשפת המכונה זאת השפה שבה כתוב הקוד המקומפל. כדאי להכיר אותו כי הוא יכול לעזור לנו להבין את הקוד בצורה טובה יותר.
26. ILDSM
27. לפתוח את ILDSM להמיר את הקובץ לטקסט למחוק את בקשת הסיסמא ואז בעזרת ILSM להמיר בחזרה.
28. פולימורפיזם זה יותר מצורה אחת. זה כשאנו מגדירים פונקציות שונות בעלות אותו שם אך כל אחת מהן שונה. Overloading | overriding הן דוגמאות לפולימורפיזם.