מבחן:

1.1) Dog, Cat, Buldog, Bird יורשים מ animal ו SiamiCat יורש מ Cat

1.2) כן, אנחנו לא יוצרים מופע של animal אבל כן יש לנו תכונות משותפות לכול החיות לכן היא גם לא תיהיה interface

2) לא, בגלל שאם יהיה מחלקה(A) שיורשת משתי מחלקות(בהן יש override) והן יורשות מאותה מחלקה, מחלקה A לא תדע באיזה מתודה להשתמש.

3) כן

4) כן

5) לא

6) כן

7) לא

8) public – המידע נגיש לכול אותו solution

Protected – המידע נגיש רק לאותה מחלקה ולמחלקות יורשות (אפילו לא מאותו solution)

Private - המידע נגיש רק לאותה מחלקה

Internal – המידע נגיש רק לאותו solution

9) לא חובה אבל רצוי.

חובה.

10) override

overload

שגיאת קומפילציה

11) ג

12) א

13) ממחלקה לא סטטית אם שדות ומתודות סטטיות ניתן ליצר יותר ממופע אחד.

14) ברגע שמייצרים מופע שלו, כדי לכתוב שם מתודות שאתה רוצה שירוצו בזמן שמייצרים מופע שלו.

15) GET SET

16) שיהיה גישה למידע הזה, וגם תוכל לתת לכול property access modifier שונה, וכן ניתן לתת ל set ו get modifier שונה.

17) backing field

18)כן, ב cmd להריץ פקודה על ה exe של disassemble ולאחר השינוי להריץ assemble

19) -stack

-list

-queue

- dictionary

20) כי אנחנו לא יורשים מהמחלקה הזו.

21) tostring – ברגע שאנחנו שולחים להדפסה את האובייקט הוא מממש את ה tostring, בלי overriding זה יתן את ה namespace של אותה מחלקה אז עדיף לעשות override בכדי שזה יהיה שימושי.

Gettype- מחזיר את ה type של האובייקט, נרצה לעשות לו override כדי שזה יחזיר לנו משהוא יותר שימושיץ

Equals- משווה בין 2 אובייקטים, ההשווה הדיי לא יעילה ב אובייקטים לכן אנחנו רוצים לתת מימוש שלנו.

Gethashcode- מחזיר את הפויינטר של ה אובייקט, אנחנו נראה שזה יהיה משהוא יותר שימושי ( כמו id של רשומה ב database)

21) virtual ממולץ לדריסה, gettype מחייבים אותנו לדרוס אותה.

22) Icomperable

23) Icomperer

24) אם לא עשינו catch ל exeption שניזרק התוכנית תקרוס אבל ה finally תמיד יבוצע.

חלק 3:

1. להשאיר שאילות בשרת של בסיס הנתונים, בכדי שהשאילתה לא תצטרך לעשות ברשת אם השרת לא יושב במחשב שלוקלי.
2. בכדי שכשניתן את התוכנית ללקוח הוא יוכל לשנות את מיקום בסיס הנתונים לפי בקשתו דרך ה app.config.
3. SQLLITE
4. Firebase
5. Mssql
6. Mssql
7. –Primary key (unique ) שמחובר על ידי foreign key ל primery key (שהוא unique) של טבלה אחרת. יוצר יחס של 1:1

- Primary key (unique ) שמחובר על ידי foreign key לprimery key (לא unique ) של טבלה אחרת. יוצר יחס של m:1 כי יכול להיות הרבה מאותו primery key בטבלה השניה כי הוא לא unique

-- Primary key (לא unique ) שמחובר על ידי foreign key לprimery key (לא unique ) של טבלה אחרת. כך שיכול להיות כפילויות של primary keys בכול טבלה.

8) inner join

9) right join

10) full outer join

11) left join

12) cross join

13) המטרה לייצר מופע בסיסי מאוד של המחלקה בכדי שנוכל לייבא הקלות את המידע מהרשומות בבסיס הנתונים. בכדי שנוכל לעשות פעולות בסיסיות עם המופע הזה, כמו להשוות אותו למופע אחד (equals) או לקבל hashcode יותר יעיל )כמו id מרשומות בבסיסי הנתונים).

14) מטרתה לייבא מידע מהבסיס נתונים ל vs בכדי שנוכל לעשות מניפולציות אם הנתונים האלה דרך ה vs, לא כדי להדפיס מהמחלקה הזה כלום היא נועדה רק לייבא מידע.