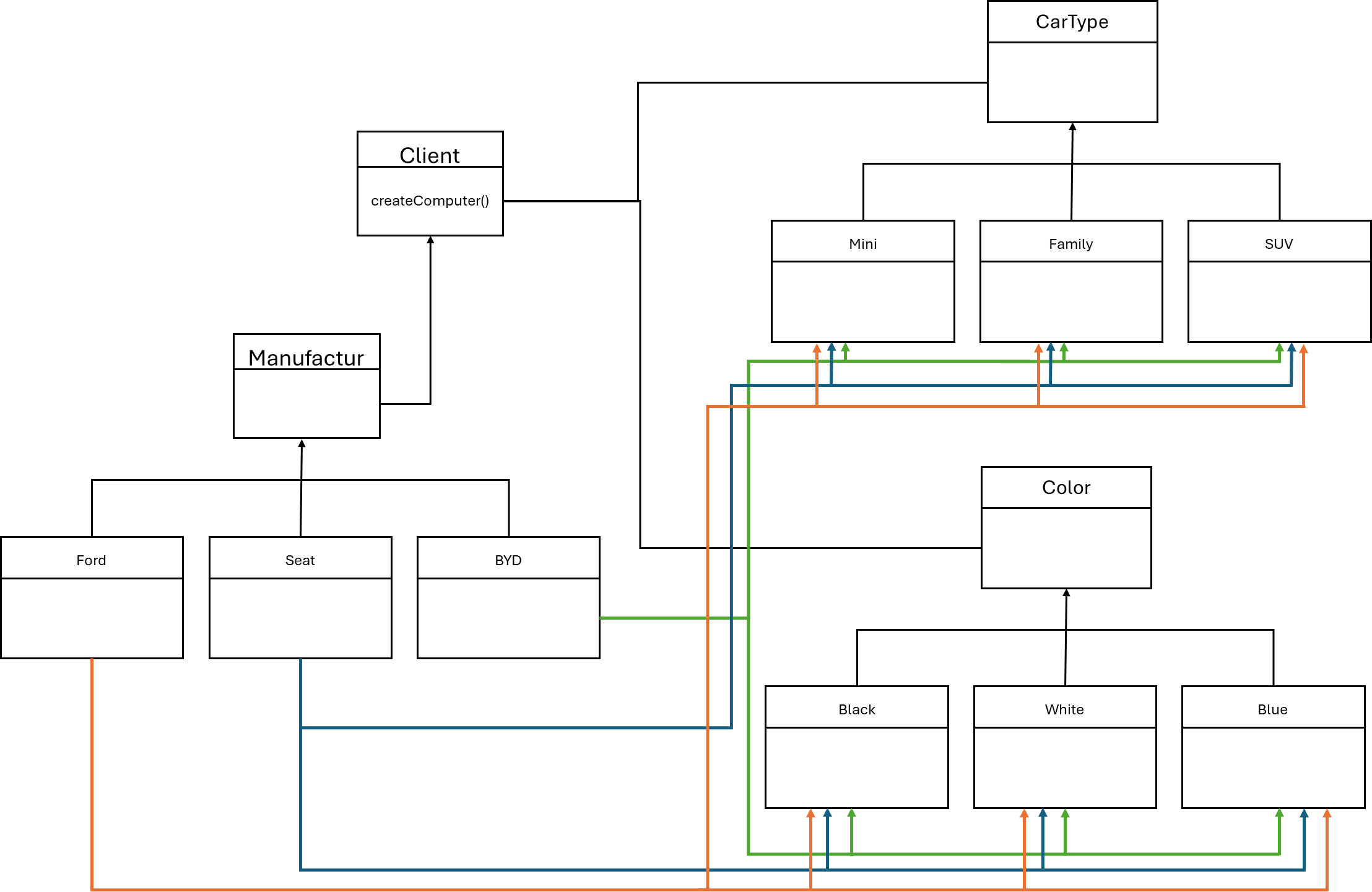
בס"ד

**פתרון תרגיל 4**

# **שאלה 1**

**1.**

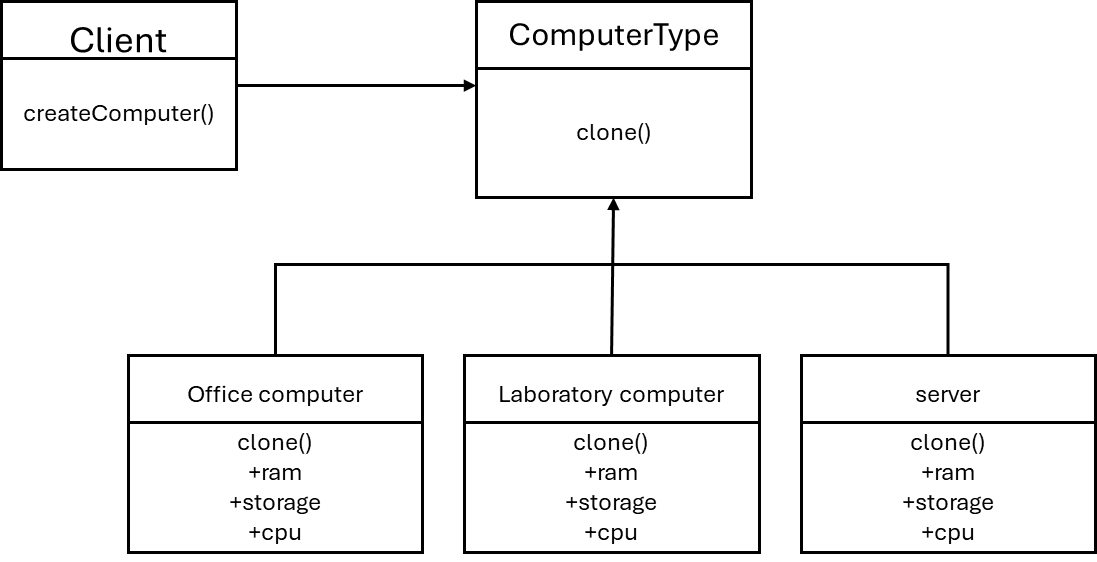
1. אם הולכים על דרך פיתוח של ממשקים המהלך מאפשר לנו גמישות רבה. כי ממשק היא תבנית או מתאר ויש מס' אפשרויות לממש. כך אפשר לקבל או להוסיף מס' אובייקטים שונים ובלי לשנות את הקוד כל פעם.
2. תבנית abstract factory. משפחת כלי רכב. הכולל מס' יצרנים, מס' סוגי כלי רכב (מיני, משפחתית, SUV וכד') ומס צבעים. בזכות תבנית זאת אפשר לייצר מגוון אובייקטים מסוג כלי רכב ונוכל בקלות להוסיף יצרן מסוים או עוד סוג כלי רכב.



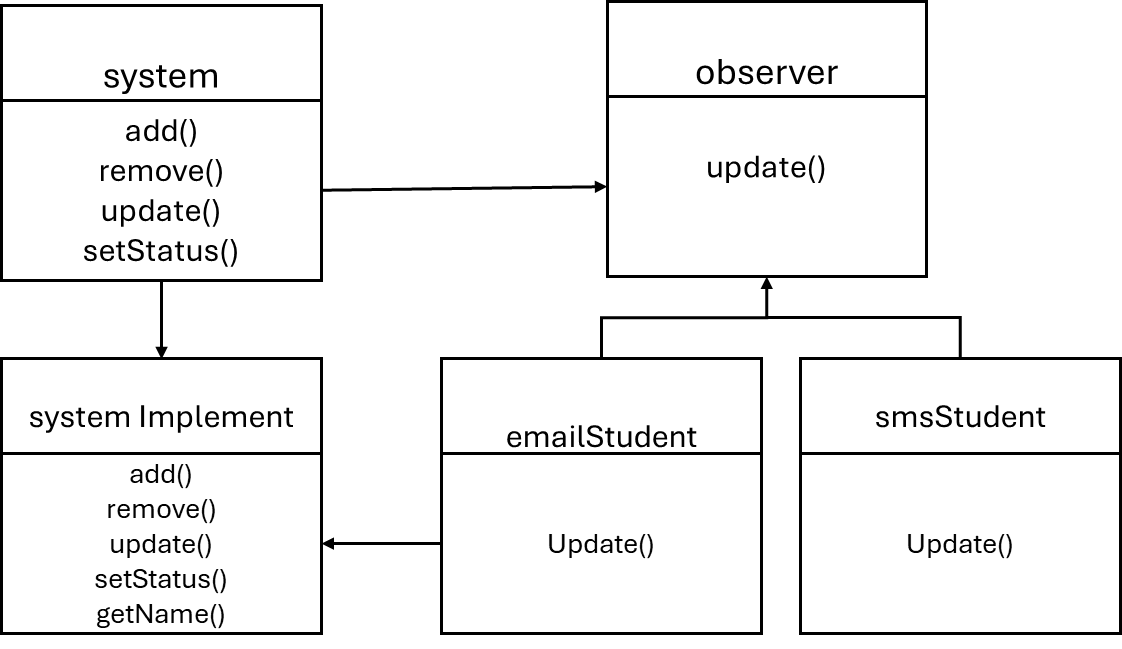
**2.**

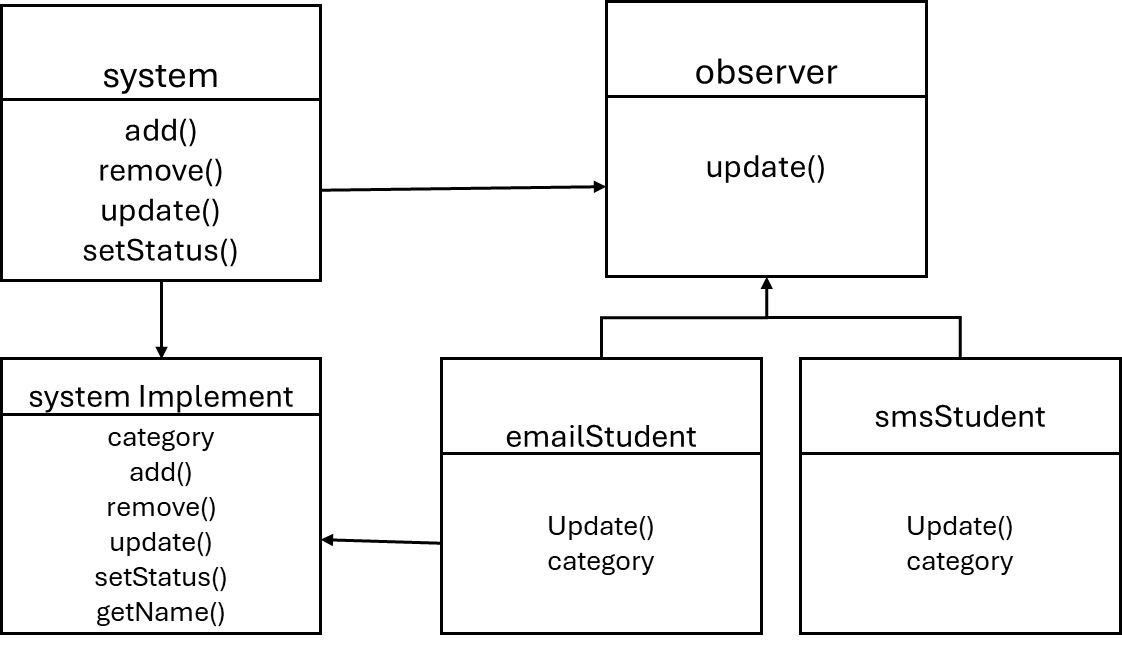
1. צימוד רופף הוא קשר בין אובייקטים בעזרת ממשק ולא מחלקה שאז הם יהיו תלויים זה בזה.
2. אי תלות ומה שנותן גמישות. אם נרצה מימוש אחר לאותה תכונה לא נצטרך לשנות את הקוד של האובייקט.
3. תבנית bridge. יש את הממשק ומצד שני את המימוש. את החלק של המימוש אפשר לממש בכמה דרכים ולהוסיף יכולות וזה לא ישפיע על הממשק כך שהאובייקט שישתמש באותו ממשק ולא יצטרך לשנות את הקוד של האובייקט על כל שינוי במימוש של הממשק.
4. כן כי יש לנו ממשק שמקשר בין אובייקטים ואין תלות בין האובייקטים.

# **שאלה 2**

1. הבעיה עוסקת במקרה שיש אובייקטים ומורכבים ויקרים לייצור.
2. ע"י יצירת אבי טיפוס לאבוייקטים היקרים יהיה ניתן "לשכפל" וליצור בעלות זולה את האובייקטים. במקרה שלנו ניתן ליצור אביטיפוס למחשבים היקרים בעלות זולה במקום ליצור אותם בעלות היקרה.
3. 

# **שאלה 3**

1. תבנית observer הוא מתווך שכל תקשורת בין אובייקטים תעבור דרך ה observer והיא תעדכן את כל השאר. כך שאם הפרויקט מוכן לרישום או שינוי בסטטוס שלו הוא יעדכן את הסטודנטים.
2. 
3. מצורף קובץ ZIP
4. לכל סטודנט ופרויקט נגדיר את התחום/נושא. ובפונקציה update נבדוק האם תחום של הפרויקט והסטודנט תואמים.



# **שאלה 4**

1. Replace Temp with Query אנחנו מחליפים משתנה זמני במתודה. במקרה שלנו יש לנו משתנה X בשורה 20 אבל בהמשך אחרי יש עוד פעם שימוש באותו X כמו ב if יש שוב שימוש במשתנה הזמני X מה שיגרור שוב קריאה למתודה וזאת פגיעה בביצועים. במקום אפשר ליצור משתנה ספציפי עבור שורה 20.
2. מפני שאם נקצה משתנה עבור שורה 20. נוכל להשתמש ב Replace Temp with Query ותוצאת המתודה תשמר במשתנה ששייך רק לשורה 20.
3. מצורף לקובץ ZIP בשם quest4.3.
4. מצורף לקובץ ZIP בשם quest4.4.