Name:		
Id:		
Program: (A)	(B) Day class	Night class

SCORE	Q1	Q2	Q3	Q4	TOTAL

KARABUK UNIVERSITY | COMPUTER ENGINEERING DEPARTMENT Object Oriented Programming | Midterm | Fall | 11.22.2021 | 17.25 | Duration: 80 mins

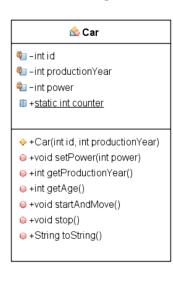
QUESTION.1 [21p]: Write outputs in the given table. All the classes are declared in the same package.

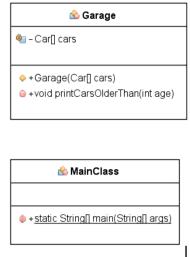
Çıktıları verilen tabloya yazın. Tüm sınıflar aynı pakette yazılmıştır.

```
public class Library {
enum Type{
                                                     static int index;
    ADVENTURE(100, "has an epic journey"),
                                                     Book[] books;
    HISTORY(200, "Facts about history"),
                                                     public Library(int i){
    SCIENCE(300, "Nonfiction");
                                                         books = new Book[i];
    private final int id;
    private final String desc;
                                                     void addBook(Book b){
    private Type(int id, String desc){
                                                         books[index] = b;
        this. id = id;
                                                         index++;
        this.desc= desc;
                                                     public void bookData(Book b){
    public String getDesc(){
                                                         System.out.println("ISBN: "+b.isbn+";"
       return this.desc;}
                                                                 +b.name+";"+b.type.getDesc());
}
                                                 }// End of Library class
public class Book {
    String name;
                                                 public class MainExamClass {
    Type type;
                                                     public static void main(String[] args) {
    String isbn;
                                                      Book b1= new Book("My Adventure", Type.ADVENTURE);
    int year;
                                                      Book b2= new Book("Our History", Types.HISTORY, 2010);
    static int counter;
                                                      System.out.println(Book.counter);
                                                                                                     //(01)
                                                      System.out.println(b1);
                                                                                                     //(02)
    public Book(String bName, Type type){
                                                      Book b3= new Book("Math", Type.SCIENCE);
    this(bName, type, 2021, ++counter); }
                                                      B2.type=b3.type;
                                                      Book b4= new Book("my voyage", Type. ADVENTURE, 2015);
    public Book(String bName, Type type, int y){
                                                      Library ourLibrary = new Library(3);
    this(bName,type,y,++counter);
                                                 System.out.println(b2.type+":"+b2.type.getDesc()); //(03)
private Book(String bName, Type type, int y, int
                                                      System.out.println(b3);
                                                                                                     //(04)
count) {
                                                      ourLibrary.addBook(b1);
       this.name= bName;
                                                      ourLibrary.addBook(b2);
       this.year= y;
                                                      System.out.println(Library.index);
                                                                                                     //(05)
       this.isbn= y+"-"+count;
                                                      ourLibrary.addBook(b3);
       this.type= type;
                                                      System.out.println(b2);
                                                                                                     //(06)
       }
                                                      ourLibrary.bookData(b2);
                                                                                                     //(07)
    @Override
    public String toString(){
        return "ISBN: "+isbn+":"+name+":"+type;
}// End of Book class
```

(01)	(05)	
(02)	(06)	
(03)	(07)	
(04)		

QUESTION.2 [30p]: Write all classes according to given class diagram on the left.





In the main method, create one Garage and 4 Car objects, and print cars that older than **n** years in the garage. (**n**: last digit of your student number) Also, print number of Cars in the garage.

Tüm sınıfları, solda verilen sınıf diyagramına göre yazınız. Main metodda, 4 Car nesnesi ve bir Garage nesnesi oluşturun ve Garage'de **n** yaşından eski arabaları yazdırın. (n: öğrenci numaranızın son hanesi) Ayrıca garajdaki Arabaların numarasını da yazdırın.

QUESTION.3 [24pt]: Write necassary lines of code to perform followings

a. Create a class called **EncapsulatedNFT**, which applies encapsulation (data hiding) to given **NFT** class.

```
public class NFT {
   public double balance; // balance cannot be less than 0 or greater than 10000
   public String owner;
}
```

- b. **EncapsulatedNFT** class should throw an exception (with an error message) if any illegal argument (balance) is set.
- c. Inside the MainClass, create an instance/object from EncapsulatedNFT, set attributes for it, handle any exception, and print the exception message on the screen.

Soru.3

- a. NFT classina encapsulation (kapsülleme, veri gizleme) prensibini uygulayan EncapsulatedNFT adlı bir class oluşturun.
- b. <u>Illegal bir arguman</u> (balance) girilirse, **EncapsulatedNFT** sınıfı bir **exception** (bir hata mesajıyla) firlatmalidir.
- c. MainClass'ta **EncapsulatedNFT** sinifindan bir nesne (instance) oluşturun ve degiskenleri atayın, herhangi bir exception durumuyla başa çıkın (handle edin) ve istisna mesajını ekrana yazdırın.

QUESTION.4 [20p]: Find 10 errors in the following Java code. The errors are both compiler and logical errors. For each error, specify the line number and briefly explain how to fix it. Aşağıdaki Java kodunda 10 hata bulun. Hatalar hem derleyici hem de mantıksal hatalardır. Her hata için satır numarasını belirtin ve nasıl düzeltileceğini kısaca açıklayın.

```
1. public class Patient{
2.
       private int id;
3.
       static int counter;
4.
       private String name;
       private float temperature=36.5;
5.
       public static final String doctor="Jone";
6.
7.
       public patient(String name, double temp,String doctor){
8.
9.
               this(name,temp,++counter,doctor);
10.
       private Patient(String name, float temp,int pId, String doctor){
11.
12.
              this.name=name;
13.
               this.temperature= temp;
14.
               this.id = pId;
15.
               this.doctor= doctor;
16.
17.
       public String getName(){
18.
              return this.name;
19. }
20.
       public getId(){
21.
               return this.id;
22. }
23.
       @Override
24.
       public String toString(){
25.
               return this.id +": name:"+this.name+"; temperature"+this.temperature;
26. }
27. }
28. public class Hospital {
29.
         static void main(String[] args) {
30.
         Patient patient1= new Patient("Ahmed", 37.2f, "Jak");
31.
32.
         System.out.println(patient1);
         Patient patient2= new Patient("Omer", 37.2f, "Jone");
33.
34.
         System.out.println(patient2.getName());
35.
         patient2.name=patient1.name;
         Patient patient2= new Patient("Ali", 37.2f, "Jan");
36.
37.
         System.out.println(patient2);
         System.out.println("number of patients:"Patient.counter);
38.
39.
        }
40.}
```