

BLM5111 NESNEYE DAYALI TASARIM VE MODELLEME DERSİ
TEMEL KAVRAMLARI ÖLÇME AMAÇLI KÜÇÜK SINAVI

Öğrenci No, Ad, Soyad:

Soru 1: Şarjör adlı bir sınıf tasarlayınız. Bu sınıf aşağıdaki üyelere sahip olmalıdır:

- a) Şarjörün mermi kapasitesini ve şarjördeki mevcut mermi sayısını saklayan üye alanlar
- b) Üye alanların ikisini de parametre olarak alan tek bir yapılandırıcı
- c) Şarjörden bir mermi ateşleyen bir metot
- d) Şarjörü doldurmaya yarayan bir metot

Tasarladığınız sınıfın kaynak kodunu yazınız. Bu sırada gerekli erişim (görünürlük) kuralları, parametreler, geri dönüş tipleri ve uygulama mantığını kodlamayı unutmayınız.

Soru 2: Şarjör türünden bir üyeye sahip olan Silah adlı bir sınıf tasarlayınız. Her silahın bir etkin menzili olduğunu da göz önünde bulundurunuz. Silah sınıfı aşağıdaki metotlara sahip olmalıdır:

- a) Parametre olarak etkin menzili alan bir yapılandırıcı
- b) Bir el ateş etmeye yarayan bir metot. Bu metot parametre olarak hedefin menzilini almalıdır.
- c) Silaha bir şarjör nesnesi atamaya yarayan bir metot
- d) Silahı yeniden doldurmaya yarayan bir metot

Tasarladığınız sınıfın kaynak kodunu yazınız. Bu sırada gerekli erişim (görünürlük) kuralları, parametreler, geri dönüş tipleri ve uygulama mantığını kodlamayı unutmayınız.

Soru 3: Seri atış yapabilen OtomatikSilah adlı yeni bir silah sınıfı oluşturulması gerekmiştir. Seri atış demek, tetiğe bir kez basıldığında birden fazla kez (3 el, 4 el, vb.) ateş edilmesi demektir. Seri atım sayısı fabrika çıkışında belirlenir ve bir daha değiştirilemez. Bu yeni sınıfı kodlayınız.

Soru 4: Yukarıdaki sorularda kodladığınız sınıfların tümünü deneme amacıyla main metodu olan bir sınıf kodlayınız.

Soru 5: Yukarıdaki sorularda kodladığınız sınıfların tümünü içeren UML sınıf şemasını çiziniz.

Soru 1: Şarjör adlı bir sınıf tasarlayınız.

```
public class Sarjor {
    private int mermiKapasitesi, mevcutMermi;

    public Sarjor(int mermiKapasitesi, int mevcutMermi) {
        this.mermiKapasitesi = mermiKapasitesi;
        this.mevcutMermi = mevcutMermi;
    }
    public boolean atesEt( ) {
        if( mevcutMermi > 0 ) {
            mevcutMermi--; return true;
        }
        return false;
    }
    public void doldur( ) {
        mevcutMermi = mermiKapasitesi;
    }
    public int getMermiKapasitesi() { //optional
        return mermiKapasitesi;
    }
}
```

Soru 2: Şarjör türünden bir üyeye sahip olan Silah adlı bir sınıf tasarlayınız.

```
public class Silah {
    private int menzil;
    private Sarjor sarjor;

    public Silah( int menzil ) {
        this.menzil = menzil;
    }
    public boolean atesEt( int hedefMenzil ) {
        if( sarjor == null )
            return false;
        if( hedefMenzil <= menzil )
            return sarjor.atesEt();
        else
            return false;
    }
    public void setSarjor(Sarjor sarjor) {
        this.sarjor = sarjor;
    }
    public void doldur( ) {
        if( sarjor != null )
            sarjor.doldur( );
    }
    public Sarjor getSarjor() { //optional
        return sarjor;
    }
}
```

Soru 3: Seri atış yapabilen OtomatikSilah adlı yeni bir silah sınıfı oluşturulması gerekmiştir.

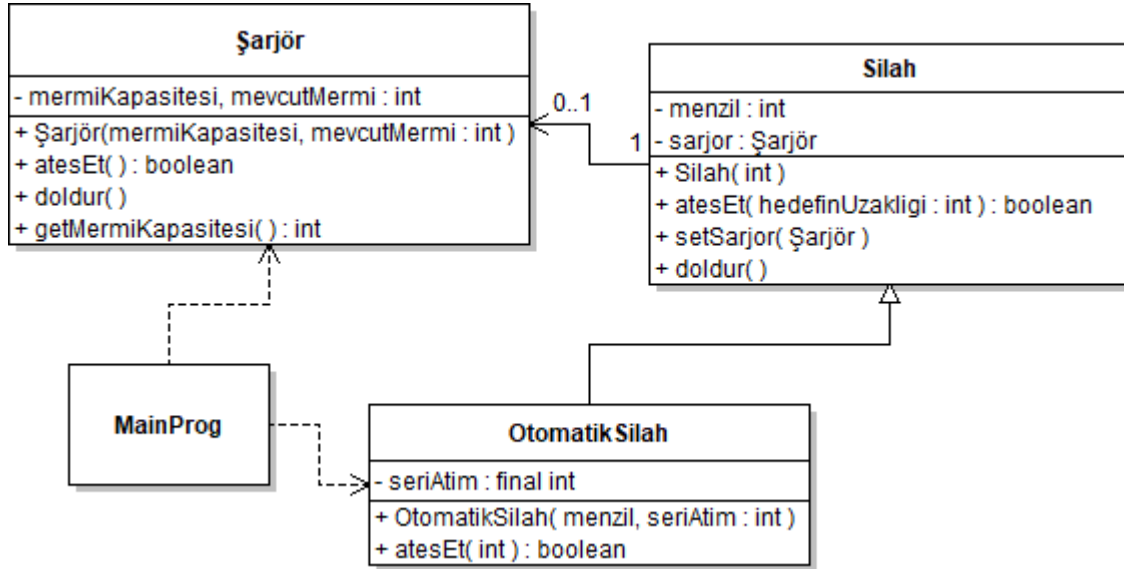
```
public class OtomatikSilah extends Silah{
    private final int seriAtim;

    public OtomatikSilah(int menzil, int seriAtim) {
        super( menzil );
        if( this.seriAtim < 1 || this.seriAtim > getSarjor().getMermiKapasitesi() )
            this.seriAtim = 3;
        else
            this.seriAtim = seriAtim;
    }
    public boolean atesEt( int hedefMenzil ) {
        boolean sonuc = false;
        for( int i = 0; i < seriAtim; i++ )
            sonuc = sonuc || super.atesEt(hedefMenzil);
        return sonuc;
    }
}
```

Soru 4: Kodladığınız sınıfların tümünü deneme amacıyla main metodu olan bir sınıf kodlayınız.

```
public class MainProg { //Scanner da kullanılabilir, zorunlu tutmadım.  
    public static void main(String[] args) {  
        Sarjor sarjor = new Sarjor(30, 30);  
        OtomatikSilah ak47 = new OtomatikSilah(400, 3);  
        ak47.setSarjor(sarjor);  
        if( ak47.atesEt(200) )  
            System.out.println("Hedef vuruldu.");  
        else  
            System.out.println("Hata: Iskalamamam lazımdı!");  
    }  
}
```

Soru 5: Yukarıdaki sorularda kodladığınız sınıfların tümünü içeren UML sınıf şemasını çiziniz.



BLM5111 OBJECT ORIENTED DESIGN AND MODELING

QUIZ FOR TESTING FUNDAMENTAL OBJECT ORIENTATION & UML CLASS DIAGRAMS

Student Number, Name, Surname:

QUESTIONS

Question 1: Design a class named AmmoClip. This class must have the following members:

- a) Two member fields for storing the bullet capacity of the ammo clip and the current number of bullets in the ammo clip.
- b) A constructor for initializing those members.
- c) A method for firing one bullet from the ammo clip.
- d) A method for filling the ammo clip.

Write the source code of the class you have designed. Do not forget to code the necessary visibility rules, method parameters, return types and application logic.

Question 2: Design a class named Rifle which has a member of type AmmoClip. Each rifle also have a range; a rifle cannot shoot beyond its range. A typical range for a rifle is several hundred meters. In addition to those two members, the Rifle class must have the following methods:

- a) A constructor for initializing the range.
- b) A method for firing a bullet. This method must take the range of the target as its only parameter.
- c) A method for inserting an ammo clip into the rifle.
- d) A method for reloading the rifle.

Write the source code of the class you have designed. Do not forget to code the necessary visibility rules, method parameters, return types and application logic.

Question 3: Design a class named AutomaticRifle. Such rifles shoot multiple amount of bullets after a trigger pull, where this amount is set at the factory and cannot be changed afterwards. Write the source code of the class you have designed.

Question 4: Please write the source code of a class with a main method where the classes you have designed so far are tested.

Question 5: Draw the UML class diagram of all the classes you have designed so far.

Question 1: Design a class named AmmoClip.

```
public class AmmoClip {
    private int bulletCapacity, bulletCount;

    public AmmoClip(int bulletCapacity, int bulletCount) {
        this.bulletCapacity = bulletCapacity;
        this.bulletCount = bulletCount;
    }
    public boolean fire( ) {
        if( bulletCount > 0 ) {
            bulletCount--; return true;
        }
        return false;
    }
    public void fill( ) {
        bulletCount = bulletCapacity;
    }
    public int getBulletCapacity() { //optional
        return bulletCapacity;
    }
}
```

Question 2: Design a class named Rifle which has a member of type AmmoClip.

```
public class Rifle {
    private int range;
    private AmmoClip clip;

    public Rifle( int range ) {
        this.range = range;
    }
    public boolean fire( int targetRange ) {
        if( clip == null )
            return false;
        if( targetRange <= range )
            return clip.fire();
        else
            return false;
    }
    public void setClip(AmmoClip clip) {
        this.clip = clip;
    }
    public void reload( ) {
        if( clip != null )
            clip.fill( );
    }
    public AmmoClip getClip() { //optional
        return clip;
    }
}
```

Question 3: Design a class named AutomaticRifle.

```
public class AutomaticRifle extends Rifle{
    private final int serialRound;

    public AutomaticRifle(int range, int serialRound) {
        super( range );
        if(serialRound < 1 || serialRound > getClip().getBulletCapacity() )
            this.serialRound = 3;
        else
            this.serialRound = serialRound;
    }
    public boolean fire( int targetRange ) {
        boolean result = false;
        for( int i = 0; i < serialRound; i++ )
            result = result || super.fire(targetRange);
        return result;
    }
}
```

Question 4: Please write the source code of a class with a main method.

```
public class MainProg { //Scanner da kullanılabilir, zorunlu tutmadım.  
    public static void main(String[] args) {  
        AmmoClip clip = new AmmoClip(30, 30);  
        AutomaticRifle ak47 = new AutomaticRifle(400, 3);  
        ak47.setClip(clip);  
        if( ak47.fire(200) )  
            System.out.println("Target is hit.");  
        else  
            System.out.println("Bug: I shouldn't miss!");  
    }  
}
```

Question 5: Draw the UML class diagram of all the classes you have designed so far.

