

## Bil 211 - LAB 2

### SINIF Diagramı:

<<Abstract Class>> Karakter
private String Karakter_adi; private int yetenek_puanı; private int ceviklik_puanı; private int zeka_puanı; private double bonus; private String tanitim_cumlesi;
<b>public String toString();</b> /*Karakterin Adı ve tanitim cümlesi döner */
<b>public abstract double saldır();</b> /* KarakterTipi, Karakter yetenek, çeviklik, zeka puanı, Silah Kullanımı ve Hamle Gücü göz önüne alınarak karakter saldırı puanı hesaplanır. */
<b>public abstract void setProp();</b> /* Karakter Tipine göre bonus değeri ve tanitim cümlesi atanır*/

<<Interface>> Silah
<b>public double hasar();</b> /* Silah Tipine göre silahın hasar puanı hesaplanır ve double olarak döndürülür*/
<b>public String bilgi();</b> /* Silah Tipine göre silah bilgisi atanır ve String olarak döndürülür */

Hamle
private String KarakterTipi; // "villain" or "hero" olabilir. private int HamleGucu; private Silah s = null; private Karakter chr = null;
<b>public void KarakterYarat(String ad, int yetenek, int ceviklik, int zeka)</b> /* Karakter chr2 nesnesi üzerinden Karakter sınıfını extend eden <b>anonymous</b> sınıf oluşturulur, gerekli metodlar override edilir. Hamle sınıfına ait Karakter chr objesine, chr2 objesi atanır. */
<b>public void SilahSec(String SilahTipi, String SilahAdı )</b> /* Silah s2 nesnesi üzerinden Silah interface'ini implement eden <b>anonymous</b> sınıf oluşturulur, gerekli metodlar override edilir. Hamle sınıfına ait Silah s objesine, s2 objesi atanır. */
<b>public void yarışma (Hamle H2)</b> /* yarışma metodunun çağrıldığı Hamle nesnesine ait karakterin saldırı puanı ile parametre olarak alınan Hamle nesnesinin (H2'nin) karakterinin saldırı puanı kıyaslanır, kazanan belirlenir */

**NOT:**

Diagramdaki sınıfları, her sınıf için belirtilen değişken ve metotları oluşturunuz. (Belirtilenlerin dışında bir değişkene ihtiyaç duyuyorsanız tanımlayabilirsiniz.)

**GÖNDERİM:**

**Kodlarınızı Uzak'a yükleyeceksiniz. Yükleyeceğiniz dosyaları klasörün için koymadan, olduğu gibi yükleyiniz. Teslim ettiğiniz dosyaların içinde main metodu bulunmasın. Ekteki Test.java dosyasını çalıştırarak kodunuzu test edebilirsiniz.**

**Kodunuz otomatik değerlendirmeden geçmezse puanın yarısını kaybedersiniz.**

**Gönderim aşamasında dosyalardaki package satırlarını silmeyi unutmayın. Terminalde kodun sadece "javac Test.java" komutuyla derlendiğinden emin olun.**

**AMAÇ:**

Bu labda; abstract class, interface, anonymous inner class yapılarının, polymorphism ve inheritance kavramlarının tekrarı amaçlanmıştır.

Labda sizden kabaca beklenen 2 hamleden oluşan bir oyun tasarlamaktır. Her bir hamlede, o hamle üzerinden bir silah (tercihe bağlı) ve bir karakter (zorunlu) tanımlanacaktır. Tanımlanan silahın ve karakterin özellikleriyle beraber hamlenin gücü göz önüne alınarak o hamle için karakterin atak puanı hesaplanacaktır. Aynı işlemler ikinci hamle için de yapılacak ve ikinci hamlenin karakterinin atak puanı da hesaplanacaktır. İki hamlenin sonunda bu karakterler yarışacak ve yarış sonunda yüksek saldırı puanına sahip karakter oyunu kazanmış olacaktır.

Yukarıda belirtilen kısmın implementasyonunu **Test.java** dosyasında bulabilirsiniz.

**Test.java Çıktısı Aşağıdaki gibi olmalıdır.**

**Ben John Wick, Seni ne pahasına olursa olsun yeneceğim**

**Ben atak silahi, kalem kullanıyorum**

**Ben Neo, Beni hafife alma**

**Hiç silah kullanmadan seni yenebilirim**

**Kazanan John Wick**

**Uyari: Hamleye bir karakter atanmıştır, yeni bir karakter atayamazsınız !**

.....  
**Ben Davy Jones, Seni ne pahasına olursa olsun yeneceğim**

**Ben atak silahi, Kilic kullanıyorum**

**Ben Jack Sparrow, Beni hafife alma**

**Ben savunma silahi Halat kullanıyorum**

**Kazanan Davy Jones**

## **SINIF YAPILARI:**

### **Abstract Class Karakter**

- Private olan bütün değişkenler için get/set metodu tanımlayın.
- **String toString()** metodu “Ben <Karakter Adı>, <Karakter tanitim\_cumlesi>” olarak override edilir.
- **abstract double saldırı()** metodu üzerinden karakterin saldırı puanı hesaplanır ve double olarak döndürülür.
- **abstract void setProp()** metodu üzerinden Karakter Tipine (**Hamle sınıfı - KarakterTipi**) göre, Karakterin tanitim\_cumlesi ve bonus değeri atanır.

**NOT: Detaylı Tanımlara Hamle sınıfı içerisinde bulunan ve dinamik şekilde anonymous inner class oluşturan KarakterYarat(String ad, int yetenek, int ceviklik, int zeka) metodunun açıklaması üzerinden ulaşılabilir.**

### **Interface Silah**

- **double hasar()** metodunda silah tipine göre karaktere kazandıracağı hasar puanı hesaplanır ve double olarak döndürülür.
- **String bilgi()** metodunda silah tipine göre atanan silah bilgisi string olarak döndürülür.

**NOT: Detaylı Tanımlara Move sınıfı içerisinde bulunan ve dinamik şekilde anonymous inner class oluşturan SilahSec(String SilahTipi, String SilahAdı) metodunun tanımı üzerinden ulaşılabilir.**

### **Hamle**

- **KarakterYarat(String ad, int yetenek, int ceviklik, int zeka)** metodunda;

Eğer Hamle sınıfındaki **Karakter chr** değişkenine bir değer atanmamışsa (null ise):

**Karakter chr2** objesi üzerinden dinamik olarak Karakter sınıfını extend eden anonymous sınıf üretilir.

### Oluşturulan Anonymous sınıf yapısı:

ANONYMOUS CLASS

**public void setProp()**

/\*

Eğer Karakter Tipi (KarakterTipi) "villain" ise bonus değeri 3 olarak set edilir.  
Karakter tanitim\_cumlesi değeri "Seni ne pahasına olursa olsun yeneceğim" olarak atanır.

Eğer Karakter Tipi (KarakterTipi) "hero" ise bonus değeri 1.5'e eşitlenir.  
Karakter tanitim\_cumlesi değeri "Beni hafife alma" olarak atanır.

\*/

**public double saldır()**

/\*

İlk olarak objenin toString() metodu çağrılarak Karakter bilgisi ekrana bastırılır.

Eğer Hamle Sınıfındaki Silah s objesine bir değer atanmamış ise (Silah s null ise)  
Karakterin saldırı puanı **(ceviklik+zeka)\*bonus + yetenek\*HamleGucu** olarak hesaplanır.  
Ekрана "Hiç silah kullanmadan seni yenebilirim" yazısı bastırılır.

Eğer Hamle Sınıfındaki Silah s objesine bir değeri atanmış ise  
Karakterin saldırı puanı **(yukarıda belirtilen formül) + atanan silahın hasar puanı** (silah objesinin hasar() fonksiyonu üzerinden) hesaplanır.  
Ekрана "Ben <Silah Bilgisi> kullanıyorum" bastırılır.

Saldırı puanı double olarak döndürülür.

\*/

**Oluşturulan chr2 objesi, Hamle sınıfının değişkeni olan Karakter chr objesine atanır.**

Eğer Hamle Sınıfındaki Karakter chr objesine bir karakter atanmış ise  
Ekрана "**Uyari: Hamleye bir karakter atanmıştır, yeni bir karakter atayamazsınız !**" yazısı bastırılır.

- **public void SilahSec(String SiahTipi, String SilahAdı )** metodunda;

**Silah s2** objesi üzerinden dinamik olarak Silah interface'ini implement eden anonymous sınıf üretilir.

### Oluşturulan Anonymous sınıf yapısı:

ANONYMOUS CLASS

**public void saldır()**

/\*

Silah Tipi (**SilahTipi**) "Atak" ise Silah hasar puanı **HamleGucu\*3** olarak belirlenir.

Eğer Silah Tipi (**SilahTipi**) "Savunma" ise hasar puanı **HamleGucu\*1.5** olarak hesaplanır.

\*/

**public String bilgi()**

/\*

Silah Tipi (**SilahTipi**) "Atak" ise "atak silahi, <SilahAdı> " String'i döndürülür

Eğer Silah Tipi (**SilahTipi**) "Savunma" ise "savunma silahi, <SilahAdı> " String'i döndürülür.

\*/

**Silah s2 objesi Hamle sınıfının değişkeni olan Silah s objesine atanır.**

- **public void yarisma (Hamle H2)** metodunda,

yarisma metodunun çağrıldığı Hamle nesnesine ait karakterin ve parametre olarak alınan Hamle nesnesinin (**H2'nin**) karakterinin setProp() fonksiyonu çağrılarak karakterlerin özellikleri atanır.

Karakterlerin saldırı puanları kıyaslanır (**saldır() metodu üzerinden**), hangi karakterin saldırı puanı daha yüksek ise ekrana "**Kazanan <Karakter adı>**" bastırılır.