

Java HAFTA 4

Paket Eriřimleri

-Paket kavramları ve paketlerin kullanılması

-Jar Dosyaları

-Eriřim belirleyicileri

-Kapsülleme

Paket Kavramı:

Kütüphaneyi oluřturan öğelere denir. Bu öğelerin tamamına verilen tanımlamadır diyebiliriz.

Geliřtirme sürecinde farklı bir kiři tarafından yazılmıř yapıları kullanmada kütüphanedeler sayesinde olur. Bu kütüphanelere verilen isimler dođrultusunda paket tanımı oluřur ve biz bu tanımlamaları kullanarak ilgili kütüphaneyi kullanmak için ilk adımı atmıř oluruz.

Örneđin; Not hesaplaması için oluřturulmuř bir kütüphaneyi farklı yazılımcılar standart ve hızlı geliřtirme aısından kullanılabilir. Kullanımı için `import com.obs.hesap;` tanımlaması verilebilir.

ÖDEV:.....

Jar Dosyaları

Jar dosya (The Java Archive File) formatı dosyaların arřivlenme ve sıkıřtırılmasına olanak tanır. Olađan durumlarda JAR dosyaları ierisinde dosyaları ierisinde sınıf dosyaları (*.class) bulunur; bazen, özellikle Appletler de, yardımcı dosyalarda (gif, jpeg) Jar dosyalarının ierisine konulabilir.

JAR dosyasının sađladığı yararlar

Güvenlik:

Sıkıřtırma:

İndirme Zamanının Azalması:

Paketleme:

Tařınabilirlik:

Eriřim Belirleyicileri

Java dilinde dört tür eriřim belirleyicisi vardır; bunlar friendly, public, protected, private.

	AYNI PAKET	AYRI PAKET	Ayrı Paket Türemiş
Public	Erişebilir	Erişebilir	Erişebilir
Protected	Erişebilir	Erişemez	Erişebilir
Friendly	Erişebilir	Erişemez	Erişemez
Private	Erişemez	Erişemez	Erişemez

--Statik olan olmayan alanlarda

Örnek: 2 farklı paket oluşturun. Bir paketin içerisinde birden fazla erişim belirleyicisi kullanarak ulaşma/ulaşamama durumunu gözlemleyin.

KAPSÜLLEME-Encapsulation:

Nesneye yönelik programlama özelliklerinden birisi kapsüllemedir. Bu dışardaki başka bir uygulamanın bizim nesnemiz ile sadece public olarak belirtilmiş arabirimlerle iletişim kurması gerektiğini, ancak, arka planda işi yapan esas kısmın gizlenmesi gerektiğini söyler. Olaylara bu açıdan bakıldığında nesnelere ikiye bölmemiz gerekiyor. İletişim kuran ve işlem gerçekleştiren. Bu yapıya aynı zamanda getset yapısı denilmektedir.

```
package javaapplication2;
```

```
public class Kullanici {
```

```
    private int beden;
```

```
    public int bedenget(){
```

```
        return beden;
```

```
    }
```

```
    public void bedenset(int bdn){
```

```
        if(bdn>20 && bdn<40)
```

```
            beden=bdn;
```

```
    }
```

```
    public void bedenhesapla()
```

```
    {
```

```
        if(beden>30)
```

```
        {
```

```
            System.err.println("M");
```

```
        }
```

```
    else
```

```
{  
System.err.println("S");  
}}
```

BAK

https://pastebin.com/
