

Java İle İlk Uygulama

Java Uygulama Geliştirme

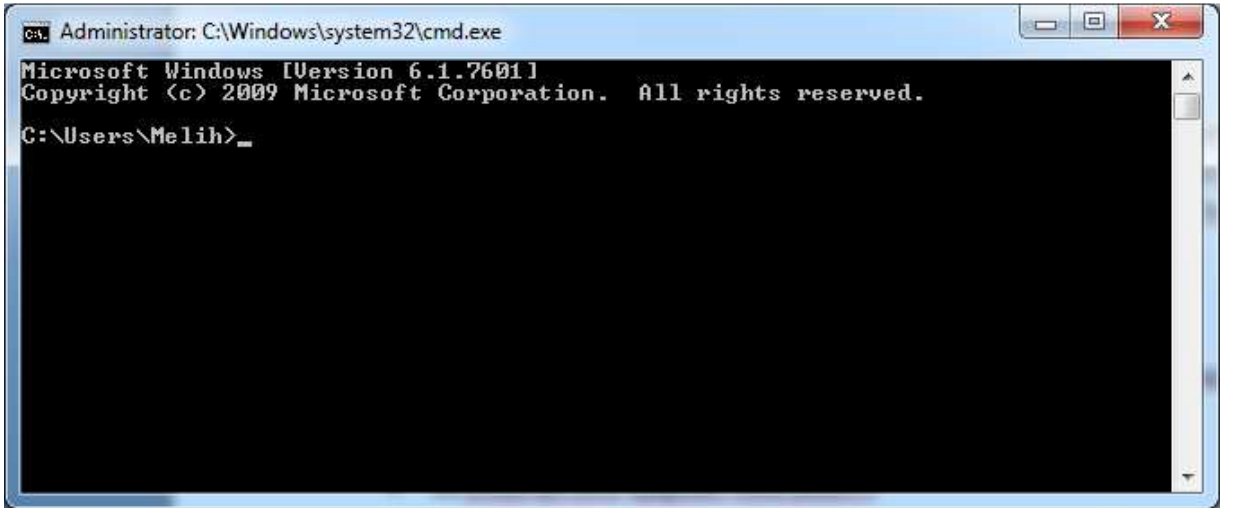
Java uygulamamızı daha öncede bahsettiğimiz üzere JDK ile derleyip JVM üzerinde çalıştırıyoruz. Tüm bu ortam ile ilgili kütüphaneler ise JRE ile geliyor. Peki burada basit bir uygulama istersek kodu nerede yazmamız gerekiyor ? Bunun için Eclipse ve NetBeans gibi geliştirme ortamlarını kullanabiliriz ancak biz ilk olarak bunu sadece JDK ile kendi ortamımızda kullanıyor olacağız.

Not : Java uygulamalarını derlemek için JDK' ya ihtiyacımız vardır.

1. Öncelikle windows kullananlar için c:/ dizini altında dev isimli bir klasör açalım. Yani c:/dev/ isimli bir dizin altında olduğumuzu varsayıyoruz.
2. Bu dizin altında Ornek.java isimli bir text dosyası oluşturalım.
3. Bu dosya içerisine aşağıdaki kodu yazalım.

```
public class Ornek {  
    public static void main(String[] args) {  
        System.out.println("Merhaba Dünya");  
    }  
}
```

4. Şimdi bu kodu derlemek için konsol ekranımızı açalım. Bunun için Windows ta Başlat -> Çalıştır içerisine cmd yazalım siyah console ekranımız aşağıdaki gibi açılacaktır.



5. Bu ekranda ilgili dizine geçmek için "cd c:/dev" yazalım. Bu bizi c:\dev dizinine yönlendirecektir.
6. Şimdi kodumuzu derlemek için aynı konsol ekranına "javac Ornek.java" yazalım.
7. İlgili dizinde Ornek.class isimli bir dosya göreceksiniz. Bu ilgili Java kodumuzun byte kod haline çevrilmiş şeklidir.
8. Bu kodu gibi bir editor ile inceleyecek olursak anlamsız karakterler görebiliriz.

9. Şimdi kodumuzu çalıştıralım bunun için yine konsol ekranına "java Ornek" yazalım.
10. Bu durumda ekranımıza "Merhaba Dünya" çıktısını almış olacağız.



Bu adımlarla ilk Java kodumuzu yazmış, derlemiş ve çalıştırmış oluyoruz. Tabi geliştiricilerin dünyası bu şekilde değil. Daha öncede bahsettiğimiz gibi bir çok Java uygulama geliştiricisi Eclipse ve NetBeans gibi geliştirme ortamlarını tercih etmektedirler.

Main Metodu

Şimdiye kadar gördüğümüz örneklerde çalışan alanları hep **main** isminde bir metod içerisine yazıyoruz. Bunun aslında önemli bir nedeni var.

Java başladığı anda bir thread mekanizması çalışır ve çalıştırılmak istenen başlangıç sınıfında ilk olarak main metoduna girer. Bu metod adından da anlaşılabilir gibi o sınıf içerisindeki ana methodtur.

Not : Şimdi yabancı geliyor olsada main metodunun neden public static ve void olduğunu daha sonra görüyor olacağız.

Not : Bundan sonra yazacağımız uygulamalarda asıl sınıflar içerisinde main metodu kullanacağız.

Çıktı Üretme

Java uygulamalarında kullanacağımız örnekler genelde konsol ekranında olacağından çıktıları da yine konsol ekranlarına veriyor olacağız. Çıktı için ayrı metod kullanabiliriz.

```
public class Ornek {  
    public static void main(String[] args) {  
        System.out.println("1. çıktı");  
        System.out.print("2. çıktı");  
        System.err.println("3. çıktı");  
    }  
}
```

System.out.print metodu standart çıktı veriyor.

System.out.println metodu standart çıktı veriyor ve bir satır aşağıya gönderiyor.

System.out.println metodu hata çıktısı veriyor ve bir satır aşağıya gönderiyor.

Not : Bundan sonraki uygulamalarda bu çıktıları kullanıyor olacağız.

Dışarıdan Değer Okuma

Uygulamalarımızda dışarıdan değer alıp bunları işlemek istiyorsak bir kaç yöntemimiz bulunuyor. Bizim kullanacağımız yöntem Eclipse ortamında da işe yarayan Scanner sınıfı olacak.

Burada uygulama akışı sırasında da konsol ekranından bilgi okuyup uygulamayı devam ettirebiliyor olacağız.

Aşağıdaki örnekte kullanıcıdan ad ve soyad bilgisini isteyim ekrana yazdırıyoruz.

```
import java.util.Scanner;

public class Ornek {
    public static void main(String[] args) {
        Scanner okuma = new Scanner(System.in);
        String ad = okuma.next();
        String soyad = okuma.next();
        System.out.println("Adiniz : " + ad);
        System.out.println("Soyadiniz : " + soyad);
    }
}
```