

BÖLÜM 1

Java Programlamaya Giriş

Java programlama dili ilk olarak başında James Gosling`n bulunduğu Sun Microsystems tarafından 1995 yılında, Sun Java Platformunun ana unsuru olarak geliştirilmeye başlandı. (Java 1.0 [J2SE])

8 Aralıkta javanın son sürümü olan 6. Sürüm yayınlandı. (J2SE). Javanın gelişmiş oluşu ve popüleritesi sayesinde; farklı platformlar için birkaç farklı konfigürasyon yapıldı. Örn: J2EE for Enterprise Applications, J2ME for Mobile Applications.

Sun microsystems yeni j2 versiyonlarını sırasıyla Java SE,Java EE ve Java ME olarak yeniden isimlendirdi. Java,"bir kere yaz her yerde çalıştır" mantığını garanti eder.

1.1 Java :

Nesneye yönelik: Javada herşey bir nesnedir. Java nesne yönelimli olduğu için kolayca geliştirilebilir.

Platformdan Bağımsız: C ve C++ dahil olmak üzere diğer birçok dilin aksine, java platform bağımlı değildir; platformdan bağımsız bayt kodu çalışır. Bu bayt kodu internet üzerinden dağıtılır ve java sanal makinesi (JVM) tarafından çalıştığı platformda yorumlanır.

Basitlik: Java kolay olarak öğrenilebilmek için dizayn edilmiştir. Nesne yönelimli programlamayı anladıysanız, javada uzmanlaşmanız zor değildir.

Güvenlik: Javanın güvenliği sayesinde virüssüz, şişmeyen sistemler oluşturulabilir. Kimlik doğrulama yöntemleri, public-key encryption sistemi tabanlıdır.

Taşınabilirlik: Serbest mimarisi ve uygulama zorunluluğu olmaması sebebiyle java taşınabilir. Derleyici ve Java Posix tabanlı kolay taşınabilir olan ANSI C ile yazılmıştır.

Sağlamlık: Java esas olarak derleme zamanı hata denetimi ve çalışma zamanı denetimini kullanarak hata eğilimli durumları ortadan kaldırır.

Çok kanallılık(Multi-threaded): Java'nın çok kanallılık özelliği sayesinde; birden fazla işlem yapabilen programlar yazmak mümkündür. Bu dizayn; geliştiricilerin akıcı çalışan, etkileşimli programlar yazmasına yardımcı olur.

Yorumlanmışlık: Java bayt kodu çalıştığı anda makine diline çevrilir ve hiç bir yerde saklanmaz. Bu geliştirme süreci, bağlantılar daha az yorucu işlemler olduğu için, daha hızlı ve analitiktir.

Yüksek performans: Anlık derleyiciler (Just-In-Time(JIT)) kullanıldığı için, Java yüksek performanslıdır.

Dağıtık: Java internetin dağıtık ortamları için dizayn edilmiştir.

1.2 Javanın Tarihi

James Gosling, kendisinin bir masa üstü cihaz projesinde kullanılması için, 1991 yılının Haziran ayında Java dili projesine başladı. İlk başlarda Gosling'in ofisinin önünde duran meşe ağacından dolayı dilin adı "oak" olarak anılıyordu. Daha sonraları "Green" olarak telaffuz edilmeye başlandı. En son olarak geliş güzel kelimeler listesinden biri olan "Java" olarak isimlendirildi.

Sun firması 1995 yılında halka açık uygulama olarak Java 1.0'ı piyasaya sundu. "Write once, Run Anywhere" (WORA)(Bir kere yaz, her yerde çalıştır) taahhütü ile, popüler platformların tümünü desteklemektedir. 13 Kasım 2006 yılında, Sun firması; GNU General Public Licence(GPL) şartlarına uygun olarak, Java'nın kodlarını ücretsiz ve açık kaynaklı olarak dağıtmaya başladı.

1.3 Basit Java Syntax'ı

Java programlarından şu şekilde bahsedebiliriz; birbiriyle iletişim kurup, birbirlerinin metotlarını çağıran objeler bütünü. Şimdi kısaca sınıf, obje, metot ve değişken ne anlama gelmekte bir bakalım.

- **Nesne:** Nesnelerin davranışları ve nitelikleri vardır. Örneğin: Bir köpeğin renk,isim,doğurganlık gibi niteliklerinin yanı sıra, yemek yeme,havlama, kuyruk sallama gibi davranışları vardır. Bir nesne, bir sınıfın oluşumudur.
- **Sınıf:** Sınıf, kendi tipinden bir objenin davranışlarını/niteliklerini anlatan bir şablondur.
- **Metot:** Metot basit açıdan bir davranıştır. Bir sınıf birden fazla method içerebilir.Metotlar; programın mantığının yazıldığı, verinin ayarlandığı ve bütün durumların çalıştırıldığı yerdir.
- **Değişkenler:** Her nesne kendine has değişkenlere sahiptir. Nesnenin nitelikleri, içindeki değişkenlere atanan verilerle oluşur.

1.4 İlk Java Programı

Aşağıda “Hello World” kelimelerini ekrana yazan kod bloğuna bakalım.

```
public class MyFirstJavaProgram {  
  
    /* This is my first java program.  
     * This will print 'Hello World' as the output  
     */  
  
    public static void main(String []args) {  
        System.out.println("Hello World"); // prints Hello World  
    }  
}
```

Dosyanın nasıl kaydedileceği, derleneceği ve programın nasıl çalıştırılacağı aşağıdaki şekilde belirtilmiştir.

- Notepad'i açın ve yukarıdaki kodu ekleyin.
- Dosyayı MyFirstJavaProgram.Java şeklinde isimlendirip kaydedin.
- Komut satırını açıp sınıfı kaydettiğiniz dizine gelin. Şimdi C:/ olduğunu varsayalım.

- 'javac MyFirstJavaProgram.java ' yazıp derlemek için Enter'a butonunu seçin. Eğer hata yoksa komut satırı sizi yeni satıra geçirecek.
- ' java MyFirstJavaProgram ' yazıp çalıştırın
- “Hello World ” yazısını ekranda göreceksiniz.

```
>  
C : > javac MyFirstJavaProgram.java  
C : > java MyFirstJavaProgram  
Hello World
```

1.5 Basit Sözdizimi (Syntax)

- Java Programlamada aşağıdaki durumları bilmek çok önemlidir.
- **Büyük-Küçük Harf Duyarlılığı:** Java Büyük-Küçük Harf Duyarlıdır. Yani “Hello” ve “hello” farklı anlamlara sahiptir.
- **Sınıf isimleri:** Bütün sınıf isimlerinde baş harf büyük olmalıdır.Eğer birkaç tane kelime kullanılacaksa her kelimenin ilk harfi büyük yazılmalıdır.

Örnek: `class MyFirstJavaClass`

- **Metot isimleri:** Bütün metot isimleri küçük harfle başlamalıdır. Eğer birkaç tane kelime kullanılacaksa ilk kelime dışındaki bütün kelimelerin ilk harfleri büyük olmalıdır.

Örnek: `public void myMethodName()`

- **Program dosya İsmi:** Programın dosya ismi kesin bir şekilde sınıf ismiyle aynı olmalıdır. Kaydederken sınıfın ismini kullanarak kaydetmelisiniz(Büyük-küçük harf duyarlılığını unutmayın.) ve sonuna .java uzantısını eklemelisiniz.(Eğer programın ve sınıfın ismi eşleşmezse programınız derlenmez.)

Örnek : 'MyFirstJavaProgram' sınıf ismi olsun. Dosyanın ismi de 'MyFirstJavaProgram.java' olmalıdır.

-- **public static void main(String args[]):** Java programlarının işlenmesi main() metodundan başlar. main() metodu bütün java programlarında gereklidir.

1.6 Java Tanımlayıcıları (identifiers)

Bütün java bileşenlerini isimlendirmek gerekir. Sınıflar,değişkenler ve methodlar için kullanılan isimlere tanımlayıcı(identifier) denir.

Javada identifier'lar için unutulmaması gereken birkaç tane kural vardır;

- Bütün identifier'lar (A - Z veya a-z) harfleri, (\$) veya (_) işaretleriyle başlamak zorundadır
- İlk harften sonra karakterler istenildiği biçimde kullanılabilir.
- Anahtar kelimeler identifier olarak tanımlanamaz.
- En önemlisi, identifier'lar büyük-küçük harfe duyarlıdır.
- Kullanılabilecek identifiers:age, \$salary, _value, __1_value
- Kullanılamayacak identifiers : 123abc, -salary

1.7 Java Erişim Denetleyicileri

Diğer dillerdeki gibi, Java'da da sınıfları, methodları vb. denetleyici kullanarak değiştirmek mümkündür. İki tür denetleyici vardır.

- **Access Modifiers (Erişim Denetleyicisi)** : default, public , protected, private
- **Non-access Modifiers** : final, abstract

1.8 Java Anahtar Kelimeleri

Aşağıdaki liste Java'daki kaydedilmiş anahtar kelimeleri gösterir. Anahtar kelimeler değişken veya identifier ismi olarak kullanılamaz.

abstract	assert	boolean	break
byte	case	catch	char
class	const	continue	default
do	double	else	enum
extends	final	finally	float
for	goto	if	implements

import	instanceof	int	interface
long	native	new	package
private	protected	public	return
short	static	strictfp	super
switch	synchronized	this	throw
throws	transient	try	void
volatile	while		

1.9 Yorum Satırları

C ve C++ daki gibi, Java'da da tekli ve çoklu yorum satırları kullanılabilir. Yorum satırındaki karakterler derleyici tarafından yok sayılır

```
public class MyFirstJavaProgram{  
  
    /* This is my first java program.  
     * This will print 'Hello World' as the output  
     * This is an example of multi-line comments.  
     */  
  
    public static void main(String []args){  
        // This is an example of single line comment  
        /* This is also an example of single line comment. */  
        System.out.println("Hello World");  
    }  
}
```