

PROGRAMLAMA DİLİ OLARAK JAVA

MUHAMMET ÇAĞATAY

ÖZET

Bu çalışmada, günümüzün en popüler programlama dili olarak kabul edilen java dilinin ortaya çıkışından günümüzdeki halini alıncaya kadar izlediği yolu en iyi şekilde incelemeye çalışırken, java dilinin yapısını, java dilini diğer dillerden ayıran özelliği de en iyi şekilde anlatmaya çalıştık.

Java dilini anlatırken, tek kaynağa bağlı kalmayıp, birçok internet sitesi ve java adına yazılmış olan rapor ve tezleri inceleyerek, en doğruyu bilgiyi en iyi şekilde aktarmaya çalıştık. Faydalı bir çalışma olması dileği ile...

MUHAMMET ÇAĞATAY

Anahtar kelimeler: Programlama Dili Olarak Java, Java Dilini Tanıyalım, Nedir Bu Java? , Java Sanal Makinesi Nedir?

DANIŞMAN
Yrd. Doç. Dr. MUSTAFA ÇAĞATAY TUFAN

İÇİNDEKİLER

1.	NEDİR BU JAVA?	1
1.1.	Java Nedir?.....	1
1.2.	Java Dilinin Ortaya Çıkışı.....	1
1.3.	Niçin Java ismi?.....	2
2.	JAVA NASIL ÇALIŞIR?.....	2
2.1.	Java Kodunun Yazılması.....	3
2.2.	Java Kodunun Derlenmesi.....	3
2.3.	Çalıştırma ve Java Sanal Makinesi.....	4
3.	JAVA İLE İLGİLİ YAZILARDA KARSILAŞABİLECEĞİNİZ BAZI KAVRAMLAR.....	4
3.1.	Java Hot Spot teknolojisi.....	4
3.2.	JIT.....	4
3.3.	Java APL.....	5
3.4.	Cop toplayıcı (Garbage Collector).....	5
3.5.	Jar.....	5
3.6.	AWT ve Swing.....	5
3.7.	SWT.....	6
3.8.	Applet.....	6
4.	JAVA PROGRAMLAMA DİLİNİNÖZELLİKLERİ.....	6
4.1.	Java Her Ortamda Çalışabilir.....	6
4.2.	Java Basittir.....	7
4.3.	Java Nesne Tabanlıdır.....	7
4.4.	Java Güvenlidir.....	7
4.5.	Java Yüksek Performanslıdır.....	7
5.	KAYNAKLAR.....	8

Resim 1.1.	Sun Microsystems firmasının	logosu
1		
Resim 1.2.	James Gosling	1
Resim 1.3.	Java Logosu	2
Resim 2.1.	java ve c++ dillerinin derleme akış diyagramları	3

SEMBOLLER DİZİNİ

AMD	Advanced Micro Devices (<i>Bir İşlemci Markası</i>)
GB	Garbage Collector (<i>Çöp Toplayıcı</i>)
JDK	Java Development Kit
JIT	Just In Time (<i>Olay Anı</i>)
JNI	Java Yerel Arayüzü (<i>Java Native Interface</i>)
JVM	Java Sanal Makinesi (<i>Java Virtual Machine</i>)
MAC	Macintosh (<i>Apple Şirketinin Ürettiği Bir Bilgisayar</i>)
OS	Operating Ssystem (<i>İşletim Sistemi</i>)
SM	Sun Microsystem (<i>Java Dilini Geliştiren Firma</i>)

1. NEDİR BU JAVA?

1.1. Java Nedir?

Java Sun Microsystems firması tarafından geliştirilmiş olan üst seviye bir programlama dilidir. İlk adı OAK olan Java aslında bilgisayarlar için değil, set üstü araçlar (WEB TV gibi) veya avuç içi cihazlar (PDA'ler gibi) için hazırlanmıştı. OAK 1995 yılında başarısız olunca, Sun firması bu programlama dilinin adını Java olarak değiştirdi ve Internet üzerinde popüler bir programlama dili haline getirmeye çalıştı.



Resim 1.1. Sun Microsystems firmasının logosu

Java, C++ dili gibi nesne yönelimli bir programlama dilidir ancak bu dile göre çok daha sadeleştirilmiştir. Soyadı.java olan Java programlarının çalıştırılabilir hale getirildiğinde soyadları.class olur ve Java desteği olan herhangi bir web tarayıcısı ile çalıştırılabilirler. Java programları çalıştırıldığı işletim sisteminden hemen hemen bağımsızdır, çünkü bu programlar için hemen hemen tüm platformlarda (UNIX, Macintosh OS ve Windows gibi) gerekli destek verilmektedir.

1.2. Java Dilinin Ortaya Çıkışı



Resim 1.2. James Gosling

Adı “*Green Project*” olarak bilinen bir projenin içerisinde yer alan **James Gosling**, proje için en uygun dilin belirlenmesi isini üstlenmişti. 1984’de Sun Microsystems’de göreve başlayan Gosling, çalışmalarına C++ ile başlamış ancak bu dilin proje için yeterli olmayacağı düşüncesiyle, yine bu dilin bir türevi olan ve ilk adı “Oak” olan yeni bir dili geliştirmeye başlamıştı. Daha sonra yerel bir kefedeki çağrışım yaparak bu dilin adını Java olarak değiştirmiştir. O günlerden bu güne Java halen geliştirilmekte olan ve günümüzde popüleritesi tartışılmaz olan bir dil olarak kullanılmaktadır.

1.3. Niçin Java İsmi?

Neden Java olmasın? Java ismi Web de insanların aklında hemen yerleşebilecek bir kelimenin arandığı beyin fırtınasında ortaya çıkmıştır. İlk olarak HotJava ve Java isimleri akla gelmiştir. Daha Java ile ilgili her programa kahve isimleri verilmiştir. Bu yüzden ki Java'nın sembolü kahvedir.



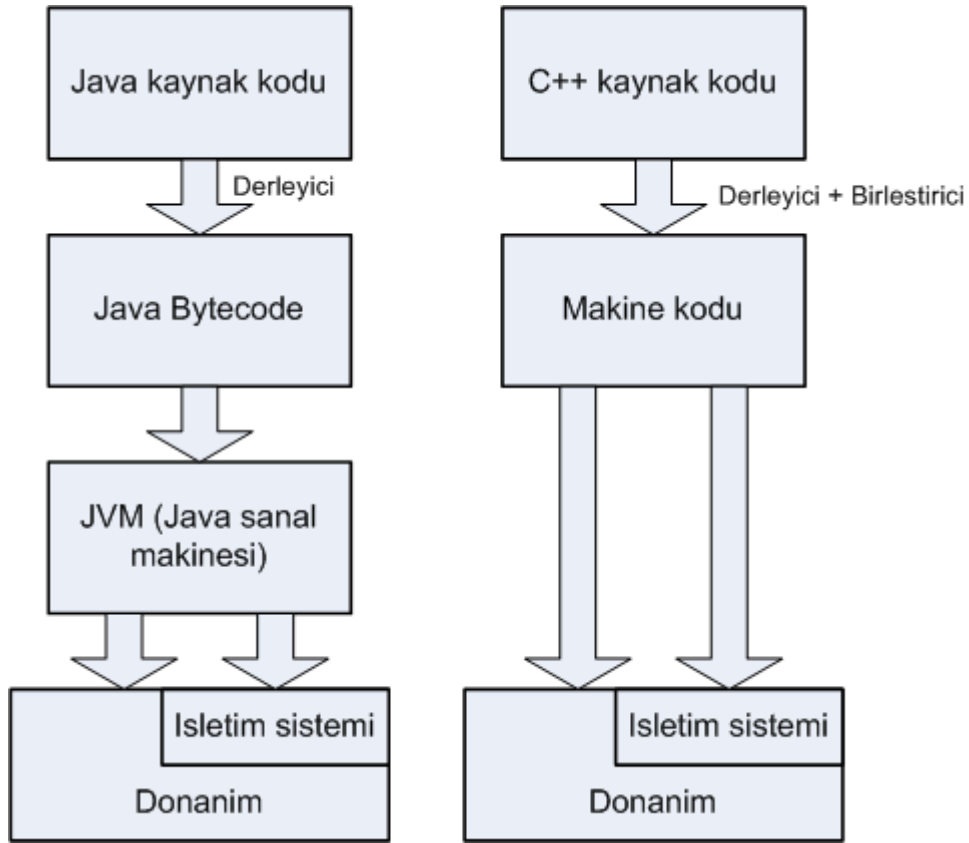
Resim 1.3. Java Logosu

2. JAVA NASIL ÇALIŞIR?

Bir Java yazılımı su şekilde geliştirilir:

1. Programcı Java kodunu yazar.
2. Bu kod bir Java derleyicisi ile derlenir. Sonuçta byte-kod adı verilen bir tür makine kodu ortaya çıkar. Platform bağımsızlığını sağlayan şey byte-code'dir. Çünkü bir kere byte-code oluştuktan sonra yazılım tüm işletim sistemlerinde çalışabilir.
3. Bu byte kod Java virtual Machine (Java Sanal Makinesi) tarafından adım adım işletilir.

Aşağıda Java ve C++ kodunun geçirdiği aşamalar gösterilmiştir.



Resim 2.1. java ve c++ dillerinin derleme akış diyagramları

2.1 Java Kodunun Yazılması.

Java nesneye yönelik bir dil olduğundan tüm yazılım sınıflar ve nesneler üzerinden yürütülür. Sınıflar uygulamadaki nesnelerin tanımlandığı kod parçalarıdır. Java'da her bir sınıf bir dosya içerisinde yer alır. Dosyaların uzantıları Java seklindedir. Dosya adı ise içinde tanımlanan sınıf ile aynıdır. Örneğin; Benim Sınıf.cagatay gibi.

2.2 Java Kodunun Derlenmesi

Derleyici kısaca herhangi bir editör ile yazılan Java kaynak kodlarını (yani Java uzantılı sınıfların yer aldığı dosyaları) Java sanal makinesinin çalıştırabileceği bir tür makine dili (assembler) olan Byte-code'a dönüştürür. Bu dönüştürülen byte kod ise (class) dosyaları içerisinde saklanır. Java kodunu derlemek için bir Java derleyicisine ve Java kütüphanelerine ihtiyacınız var. Şu anda iki derleyici yaygın olarak kullanılmaktadır. Bir tanesi Sun'ın SDK'si ile birlikte gelen Java diğeri ise IBM'in açık kodlu derleyicisi jikes.

Jikes, çok hızlı derlemesi ile ünlü olsa da en son Java yazılımlarını derlemek için javac'ye ihtiyaç duyulabilir. Ayrıca Linux altında derleme yapmak için gnu lisansı ile geliştirilen açık kodlu Gnuj da kullanılabilir. Derleyici ve kütüphanelerin bulunduğu (Java API) uygulamaya "J2SE SDK" adi verilmiştir.

2.3 Çalıştırma ve Java Sanal Makinesi

Sanal makine, donanımdan bağımsız yazılım geliştirme ihtiyacına cevap verme amacıyla geliştirilen bir teknolojidir. Java'nın temel felsefesi olan "bir kere yaz, her yerde çalıştır" sanal makine sayesinde var olmuştur. Sanal makineyi bir yönden bir tür hayali bir mikroişlemci gibi düşünebiliriz.

Gerçek tüm mikroişlemciler (Intel Pentium, AMD Athlon, Sun Sparc vs.) belirli bir grup komutu işlemek üzere tasarlanmıştır. Bu komutlara işlemcinin komut kümesi adı verilir. Örneğin x86 komut kümesi gibi. Tüm yazılımlar çalışabilmek için önce bu komut kümesine dönüştürülür, daha sonra işlemci bu komutları sıra ile gerçek işlemci komutlarına dönüştürüp işletir. Java Sanal makinesi de Byte-kod komut kümesini tıpkı işlemci gibi adım adım işletir. Java'nın interpreted bir dil olarak adlandırılmasının nedeni budur. Byte-kod ilkel işlemlerin yanında (ilkel işlemler, mikroişlemci seviyesi komutlardır, aritmetik işlemler, bit işlemleri, bellek ve yığın işlemleri vs.) sanal makinenin üzerinde çalıştığı işletim sistemine yönelik işlemler de barındırır. Bu sayede Java Virtual Machine yazıcı, seri port, grafik, dosya servisi, ağ bağlantısı gibi yazılım ve donanım servislerine erişim yapabilir.

Java'nın doğrudan byte code çalıştırması performansının düşük olabileceği izlenimini verebilir. Ancak, JVM tasarımı geçen on yılda çok değişmiş ve geliştirilmiştir. Şu anda Java'nın performansı çoğu alanda C++'in performansına yakın bir seviyededir ve işlemci hızı ve bellek miktarının her geçen yıl katlanarak artması ile performans konusu çoğu uygulamada artık ikinci planda kalmıştır.

3. JAVA İLE İLGİLİ YAZILARDA KARŞILAŞABİLECEĞİNİZ BAZI KAVRAMLAR

3.1 Hot Spot teknolojisi

Java sanal makinesi Hot Spot adı verilen özel bir teknolojiyi içinde barındırır. Hot Spot yani sıcak nokta, bir yazılımda sürekli olarak tekrarlanan ve üzerinden geçilen kod bölümlerine verilen bir isimdir. Java sanal makinesi bir kod çalışmaya başladıktan sonra sıklıkla kullanılan kod bloklarını gözler. Bir süre sonra bu byte code bloklarının çalışılan sistemdeki gerçek işlemci komut karşılıklarını bir tür cep belleğe yazar ve zaman ilerledikçe artık byte kod üzerinden değil doğrudan sistemin öz komutlarını kullanarak yazılımın o bölümlerini işletmeye baslar. Bu şekilde ciddi performans avantajı sağlanmıştır.

3.2 JIT

Java ilk çıktığında byte code işletme hızı çok iyi değildi. Yerine göre sistemin öz yazılımlarından 5–10 kat yavaş çalışıyordu. Bu nedenle bazı yazılım geliştirme şirketleri JIT yani Just-in-time compiler, "anında derleme" araçları üretmeye başladılar. Yapılan şey byte kodu sanal makinenin kurulu olduğu gerçek sistemin diline anında derleme yaparak

dönüştürmesiydi. Bu sayede performansta ciddi artışlar sağlandı. Ama 2000 yılından sonra Hot Spot teknolojisinin gelişmesi ile JIT'in işlevi VM' içinde yer almaya başlamış, işlemci hızı ve bellek miktarının dramatik biçimde artması ile dış JIT yazılımları popülerliğini kaybetmiştir. Bugün halen bir kaç ürün pazarda bulunsa da genellikle bu yöndeki ihtiyaç yok olmuş gibi gözükmemektedir.

3.3 Java API

Java API, java yazılımlarında kullanılan yazılım kütüphanelerine genel olarak verilen isimdir. Java API ile disk, grafik, ağ, veri tabanı, güvenlik gibi yüzlerce konuda kullanıcılara erişim imkânı sunulur. Java API J2SDK'nin bir parçasıdır.

3.4 Çöp toplayıcı (Garbage Collector)

Çöp toplayıcı Java'nın en belirgin özelliklerinden birisidir. C++, C gibi dillerin en büyük handikaplarından birisi dinamik bellek yönetimidir. Yazılımda işaretçi (pointer) kullanarak dinamik olarak bellek ayırdıktan sonra o bellek ile isiniz bittiğinde mutlaka ayrılan belleği bellek yöneticiye özel metotlar yardımıyla (delete, destructor vs.) iade etmeniz gerekir. Yoksa bellek sızıntısı (Memory Leak) oluşur ve bu bir süre sonra yazılımın ve işletim sisteminin beklenenden farklı davranmasına yol açar. Bugünün tüm büyük C ve C++ yazılımları az da olsa bellek sızıntısı içerir (işletim sistemleri dâhil). Sızıntıların tespiti oldukça güçtür ve bulunması zor hatalara yol açar. Çöp toplayıcı sayesinde Java'da bir nesne oluşturulduktan sonra o nesne ile isiniz bittiğinde hiç bir şey yapmanız gerekmez. Sanal makine akıllı bir biçimde kullanılmayan bellek bölümlerini belirli aralıklarla ya da adaptif metotlarla sisteme otomatik olarak temizler ve sisteme iade eder. Bu işleme Çöp toplama, ya da Garbage collection adı verilir. Çöp toplama sistemlerinin yapısı oldukça karmaşıktır ve geçen yıllar içinde büyük gelişmeler kaydedilmiştir. Çöp toplayıcının varlığı Java'da bellek sızıntısı olmayacağı anlamına gelmez, ama bellek sızıntıları daha ender olarak ve farklı şekillerde karşınıza çıkar ve genellikle tedavi edilmesi daha kolaydır.

3.5 Jar

Jar, aslında bir tür sıkıştırma formatıdır. Jar ile derlenen Java kodları ile oluşan yazılımın paketlenip taşınması kolay bir hale getirilir. Jar dosyaları temelde bytekod blokları içerir. Jar dosyaları genellikle kütüphane oluşturmada ya da uygun biçimde hazırlanırsa işletim sisteminden doğrudan çalıştırılabilecek bir şekilde kullanılabilir (Executable Jar, işletilebilir jar) jar dosyalarının içeriğini sıkıştırma yazılımları ya da Java yazılım geliştirme araçları ile inceleyebilirsiniz. Java 1.5 ile yeni bir tür jar oluşturma metodu da kullanıma girecek. Pack200 adı verilen hiper-compression algoritması ile jar dosyaları 8 kata varan oranlarda daha az yer kaplayacak. Bunun özellikle uzak uygulamaların kullanımını ciddi biçimde kolaylaştırması bekleniyor.

3.6 AWT ve Swing

AWT, ilk Java ile birlikte geliştirilen temel grafik ara yüz oluşturma kütüphanesine verilen isimdir. Ancak Java 2 platformu ile birlikte AWT'nin yetersiz görülmüş ve çok daha geniş ve gelişmiş özelliklere sahip Swing Kütüphanesi sisteme eklenmiştir. Özellikle çok platform destekleyen yazılımlarda kullanıcı ara yüzü geliştirme aracı olarak Swing halen önemini korumaktadır. Swing, önceleri işletim sisteminin kullandığı donanım grafik hızlandırma araçlarını kullanmadığından yavaşlığı ile eleştirilere hedef olmuştu. Özellikle

Java 1.4 ile Swing, hem genel olarak sanal makinenin hızlanması ve kimsen donanım hızlandırmayı kullanması ile bu kötü şöhretinden sıyrılmaya başladı. Java 1.5 ile donanım, özellikle OpenGL kullanımı ve yeni ara yüz gösterim şekli ile Java'nın masaüstü uygulama geliştirmede popülerleşmesi bekleniyor. AWT halen swingin bir alt katmanında, temel 2 boyutlu grafik işlemlerinde kullanılmaya devam ediyor.

3.7 SWT

SWT swing'e bir alternatif olarak IBM tarafından geliştirilen bir gösterim sistemidir. Swing'den en büyük farkı çalıştığı işletim sisteminin grafik kütüphanesi ve komutlarını kullanmasıdır. Bu nedenle SWT uygulamaları Swing'e göre çoğu yerde daha hızlı ve işletim sistemindeki diğer uygulamaları andıran bir şekilde çalışmasını sağlar. Ancak yapı itibari ile SWT kullanımı Swing kadar efektif olamayabiliyor (özellikle olay mekanizması, tablo ve ağaç yapılarındaki yavaşlığı, ayrıca Linux performansı ile SWT eleştirilmiştir.) Swing in Java 1.5 ile performans açığını kapatacağı iddia edilse de SWT'nin de artık Java camiasında kabul görmüş bir sistem olduğu aşıkardır. SWT'nin dezavantajı ise Java'nın bir parçası olmamasıdır. yani SWT uygulamaları SWT kütüphanesi ile birlikte dağıtılmaktadır. En bilinen SWT uygulaması ünlü Java yazılım geliştirme aracı Eclipse' tir. Bununla birlikte son yıllarda Swing ile profesyonel derecede ara yüze sahip masaüstü yazılımları da ortaya çıkmıştır. Sonuçta her şey yazılımcının aracı ne kadar efektif kullandığına bakıyor.

3.8 Applet

Applet, uzaktaki sistem üzerinden indirilip internet tarayıcı üzerinde çalıştırılabilinen Java uygulamalarına verilen isimdir. Java'nın son kullanıcılar tarafından tanınması applet sayesinde olmuştur dersek yanlış olmaz herhalde. Applet'ler sisteme zarar veremeyecek bir şekilde tasarlanmıştır ve bugün özellikle oyun sitelerinde halen yaygın olarak kullanılmaktadır. İçerisinde Applet olan bir sayfayı açmaya çalıştığınızda tarayıcınız otomatik olarak Java sanal makinesini çalıştırıp ekranın applet'e ayrılan bölümünde uygulamanın çalışmasını sağlar.

4. JAVA PROGRAMLAMA DİLİNİN ÖZELLİKLERİ

4.1 Java Her ortamda çalışabilir

Java teknolojisinin en önemli özelliği her ortamda, en küçük bilgisayarlardan süper bilgisayarlara kadar, çalışabilmesidir. Java teknolojisi bileşenleri nasıl bir bilgisayar, telefon, televizyon veya işletim sistemi olduğuna bakmaksızın Java platformunu destekleyen her türlü ortamda çalışır. Java teknolojisinin yaratılış amacı daha önce hiç kolay olmadığı kadar kolay bir şekilde bilgisayar ve diğer iletişim araçları arasında etkileşim kurmaktır. Java ilk ortaya çıktığında asıl amacı elektronik ev aletlerinin birbirleriyle iletişim kurabilmesini sağlamaktır. Java hızlı bir şekilde gelişerek bugün ki halini almıştır.

4.2 Java Basittir

Java ile program hazırlamak oldukça kolay ve zevklidir. Daha önce bir programlama dili ile çalışmış veya en az bir nesneye programlama dili ile çalışmak yeterlidir. Ayrıca Java programlama dilinin C++ programlama diline olan benzerliği sebebiyle C++ programlama dili çalışmış olanlar için çok zevkli yeni bir deneyim olacaktır.

4.2 Java Nesne Tabanlıdır

Java nesne yönelimli bir programlama dilidir. Bir programlama dilinin nesne yönelimli olması, dilin kendi özelliği itibarıyla aynı amaca yönelik görevlerin sınıf (class) denilen yapılar içerisinde toplanmasına ve bu prensibe göre kullanımına izin vermesidir. Nesne yönelimli programcılığın birçok avantajı vardır. Bunlardan en önemlisi, bir projede birbirinden bağımsız olarak tasarlanan sınıf nesneleri kullanılmasıdır. Böylece projeye belli görevlerin eklenmesi veya çıkarılması projenin bütününe dokunmadan mümkün hale gelebilmektedir.

4.3 Java Güvenlidir

Java programlama dili hazırlanan programlar bugün güvenliğin çok önemli üst düzey devlet kuruluşlarında ve NASA da kullanılmaktadır. Java ile hazırlanan programlara virüslere ve hacker'lara karşı geliştirilen en güvenli programlar olarak düşünülmektedir.

4.4 Java Yüksek Performanslıdır

Java ile hazırlanan programlar hızlı çalışmaları ve bilgisayara az yük getirmeleri nedeniyle İnternet ve intranet uygulamalarında tercih edilmektedir.

6. KAYNAKLAR

1. Academytech_java_tr.pdf modülü
2. Zafer DİNÇ “Java programlama dilinde veri tabanına bağlantı için JDBODBC ara yüzü” isimli tezi
3. <http://www.bilgiustam.com/java-nedir-nasil-calisir//print>
4. [http://tr.wikipedia.org/wiki/Java_\(programlama_dili\)#Java_programlar.C4.B1_nas.C4.B1l_.C3.A7al.C4.B1.C5.9Ft.C4.B1r.C4.B1l.C4.B1r.3F](http://tr.wikipedia.org/wiki/Java_(programlama_dili)#Java_programlar.C4.B1_nas.C4.B1l_.C3.A7al.C4.B1.C5.9Ft.C4.B1r.C4.B1l.C4.B1r.3F)
5. [http://tr.wikipedia.org/wiki/Java_\(programlama_dili\)#Java_API](http://tr.wikipedia.org/wiki/Java_(programlama_dili)#Java_API)
6. [http://tr.wikipedia.org/wiki/Java_\(programlama_dili\)#At.C4.B1k_veri_to_play.C4.B1c.C4.B1](http://tr.wikipedia.org/wiki/Java_(programlama_dili)#At.C4.B1k_veri_to_play.C4.B1c.C4.B1)
7. <http://www.1923turk.com/forumdisplay.php?f=30>
8. <http://www.javaturk.org/?cat=10>
9. <http://www.csharpnedir.com/>
10. http://www.bilisimterimleri.com/bilgisayar_bilgisi/bolum/1.html
11. <http://www.forumalev.net/diger-kodlama-dilleri/8833-java-nedir.html>