

<b><u>CGI</u></b> .....	<b>2</b>
1.1 <b><u>GİRİŞ</u></b> .....	2
1.2 <b><u>CGI PROGRAMCIĞI NEDİR?</u></b> .....	2
1.3 <b><u>CANLI VERİNİN YÖNETİLMESİ</u></b> .....	4
1.4 <b><u>CGI PROGRAMCIKLARINIZ İÇİN BİR PROGRAMLAMA DİLİ SEÇME</u></b> .....	4
1.4.1 <b><u>UNIX Shell Kullanma</u></b> .....	6
1.4.2 <b><u>C/C++ Kullanma</u></b> .....	8
1.4.3 <b><u>Perl Kullanma</u></b> .....	9
1.4.4 <b><u>JavaScript Kullanma</u></b> .....	10
1.4.5 <b><u>VBScript Kullanma</u></b> .....	11
1.5 <b><u>NEDEN CGI PROGRAMCIKLARI?</u></b> .....	12
1.6 <b><u>CGI PROGRAMCIKLARI NASIL ÇALIŞIR?</u></b> .....	13
1.7 <b><u>CGI PROGRAMCIKLARINA GİRDİ</u></b> .....	14
1.8 <b><u>ORTAM DEĞİŞKENLERİ</u></b> .....	14
1.9 <b><u>STANDART ORTAM DEĞİŞKENLERİ</u></b> .....	15
1.9.1 <b><u>AUTH_TYPE</u></b> .....	16
1.9.2 <b><u>CONTENT_LENGTH</u></b> .....	17
1.9.3 <b><u>CONTENT_TYPE</u></b> .....	17
1.9.4 <b><u>Basit MIME Tipleri</u></b> .....	17
1.9.5 <b><u>Sık Kullanılan MIME Tipleri</u></b> .....	18
1.9.6 <b><u>GATEWAY_INTERFACE</u></b> .....	21
1.9.7 <b><u>HTTP_ACCEPT</u></b> .....	21
1.9.8 <b><u>HTTP_USER_AGENT</u></b> .....	21
1.9.9 <b><u>PATH_INFO</u></b> .....	22
1.9.10 <b><u>PATH_TRANSLATED</u></b> .....	22
1.9.11 <b><u>QUERY_STRING</u></b> .....	23
1.9.12 <b><u>REMOTE_ADDR</u></b> .....	24
1.9.13 <b><u>REMOTE_HOST</u></b> .....	24
1.9.14 <b><u>REMOTE_IDENT</u></b> .....	24
1.9.15 <b><u>REMOTE_USER</u></b> .....	24
1.9.16 <b><u>REQUEST_METHOD</u></b> .....	24
1.9.17 <b><u>SCRIPT_NAME</u></b> .....	25
1.9.18 <b><u>SERVER_NAME</u></b> .....	25
1.9.19 <b><u>SERVER_PORT</u></b> .....	25
1.9.20 <b><u>SERVER_PROTOCOL</u></b> .....	26
1.9.21 <b><u>SERVER_SOFTWARE</u></b> .....	26
1.10 <b><u>CGI KOMUT SATIRI</u></b> .....	26
1.11 <b><u>STANDART CGI GİRDİSİ</u></b> .....	29
1.12 <b><u>CGİ GİRDİSİNİ TEMİZLEME</u></b> .....	29
1.13 <b><u>CGİ PROGRAMCIKLARINDAN ÇIKTI</u></b> .....	32
1.14 <b><u>CGİ BAŞLIKLARI</u></b> .....	32
1.14.1 <b><u>CGİ Başlıklarında Kullanılan Content-Type'lar</u></b> .....	32
1.14.2 <b><u>CGİ Başlıklarında kullanılan Location'lar</u></b> .....	33
1.14.3 <b><u>CGİ Başlıklarında kullanılan Status'lar</u></b> .....	33
1.14.4 <b><u>CGİ Başlıklarını Temizleme</u></b> .....	34

# CGI

## 1.1 GİRİŞ

CGI programcıklarını (script) kullanarak okuyucu ile gerçek bir etkileşim içinde güçlü, kişisel ve profesyonel Web yayımları yaratabilirsiniz. CGI programcıkları Web sunucusu ile diğer uygulamalar arasında bir ağgeçit gibi davranan dış kaynaklı programlardır. CGI programcıklarını okuyuculardan gelen girişleri işlemek için kullanabilir, böylelikle okuyucu ile çift yönlü bir iletişim yolu koyabilirsiniz. Okuyucu girişleri doldurma formlarının verilen şeklinde olabildiği gibi veri tabanı sorgulamaları için anahtar sözcükler ya da okuyucunun inceleyicisinin ya da bağlantının tanımladığı değerler şeklinde de olabilir.

CGI programcıklarınızı bu girişleri bir indekse veri olarak eklemek, veritabanında sorgulama yapmak, kişisel dökümanlar yaratmak için kullanabilirsiniz. CGI programcıklarının en mükemmel özelliği karmaşıklıklarını kullanıcıdan gizlemeleridir. Web üzerinde bir doldurma formunu ya da bir haritayı kullanırken ne olduğunu bilmeseniz de bir ağgeçit programcığı kullanıyor olabilirsiniz. Bu nedenle herşey otomatik yapılıyor gibi görünür. Veriyi girin, farenin düğmesine basın ve bir an sonra sonuç görüntülensin. Bu bölümde fareye tıkladıktan sonra sonuç görülünceye kadar gerçekte neler olduğu anlatılmaktadır. CGI programcıkları hakkında neler bilmeniz gerekiyorsa hepsini, CGI programcıklarının ne olduğunu, nasıl ve neden kullanıldıklarını bu bölümde öğrenebilirsiniz.

## 1.2 CGI Programcığı Nedir?

CGI programcıkları Web sunucusu üzerinde çalışan dış kaynaklı programlardır. CGI programcıklarını yüksek etkileşimli Web yayınları yaratmak için kullanabilirsiniz. Web sunucuları üzerinde kullanılan yabancı kaynaklı programların standartları ve ortak ağgeçit arayüzünün diğer uygulamalarla etkileşiminin nasıl olması gerektiği belirlenmiştir. Ortak, ağgeçit ve arayüz sözcükleri bu standartların anahtar sözcükleridir ve nasıl çalıştığını tanımlar.

CGI; programcığın erişilebilir olmasını sağlamak için ortak bir yol tanımlayarak, herhangi bir kimsenin kullandığı ortam ne olursa olsun bir CGI programcığına bilgi

geçirmesini sağlar. CGI; programcık, sunucu ve diğer uygulamalar arasında bir bağlantı ya da ağgeçit tanımlayarak dış kaynaklı programları genelleştirilmiş girişler olarak kabul edilebilmesini ve diğer uygulamalara bilgi aktarılabilmesini mümkün kılar. CGI; kullanıcıların dış kaynaklı programlara erişebilmesi için bir arayüz ya da bir yol tanımlayarak dış kaynaklı programlar için gerçekleştirilen karmaşık arayüz işlemlerini birkaç basit prosedüre indirger.

CGI'yi geliştirenler Web yayımcılarının okuyucularını karmaşıklıklardan korumasını sağlamak için güçlü ve eklenebilir nitelikte özellikler yaratmak üzere bir CGI standardı oluşturmak amacıyla bu anahtar kavramlar doğrultusunda çalışmışlardır. Okuyucunun tek yapması gereken iş, bir görüntü haritasındaki bir alana tıklamak ya da doldurma formunu tamamladıktan sonra teslim etmek olacaktır. Herşey farenin tuşuna dokunur dokunmaz otomatik yapıyor gibi görünür ve kullanıcı nasıl veya niçin gibi sorular sorma durumuyla karşılaşmaz, herşey kendi kendine yürür.

Bir Web yayımcısı için özellikle güçlü Web yayımları yaratırken CGI programcıklarının sunduğu avantajlardan faydalanmak istiyorsa, CGI programcığının nasıl çalıştığını bilmek çok önemlidir.Kullanıcı sadece yaptığı işlerin sonuçlarını görür arkadaki manzaradan habersizdir.

Programcıklar yürütülürken arka planda gerçekleşen olaylar şöyledir:1.Okuyucunun inceleyicisi girişleri Web sunucusuna aktarır.

2.Sunucu sistem sırası geldiğinde girişleri CGI programcığına aktarır.

3.CGI; programcığı girişi işler ve eğer başka bir uygulamaya gerekiyorsa uygulamayı programcığa geçinir ve çıkışı Web sunucusuna gönderir.

4.Web sunucusu okuyucunun cevabını onun inceleyicisine geri yollar. Bir CGI programcığının çıkışı; bir veri tabanı sorgulamasının sonucundan, okuyucunun cevabına göre yaratılmış yeni bir dökümana kadar herşey olabilir.

UNIX sistemlerinde CGI programcıkları usr dosya sistemi içinde cgi-bin olarak adlandırılan kataloglara yerleştirilmiştir ve CGI uygulamaları da yine usr dosya sistemi içinde cgi-scr olarak adlandırılan kataloglara yerleştirilmiştir. Diğer sistemlerde CGI programcıklarınızı ve uygulamalarınızı nerelere yenleştirmeniz gerektiği Web sunucu dökümanında açıklanır.

### 1.3 Canlı Verinin Yönetilmesi

HTML dosyalarının bir Internet hizmet birimine yüklenmeden önce üretilmeleri bu hizmet biriminin kullanıcıdan gelen bir talebi okuyarak etkileşimli bir biçimde bir HTML dosyası oluşturmasını ve bunu kullanıcıya sunmasını sağlamaktan çok farklı bir şeydir.

Bunu gerçekleştirmek için temel olarak başvurabileceğiniz iki teknik vardır: Common Gateway Interface'i (CGI) enkarnasyonlarından birinde kullanmak veya hizmet birimine bağlanan bir DLL oluşturmak. Her iki durumda da hizmet birimine aradığınız dosyanın yolunu içeren bir parametre geçmek yerine, ona programın (veya DLL'in) adını ve bazı parametreler geçersiniz.

Delphi bu iki hizmet birimi ekini oluşturmanıza da, Web hizmet biriminin bir Windows bilgisayarında çalışması koşuluyla olanak verir. Aslında Delphi'nin Client/Server Suite edisyonu bu beceri için özel destek de dekler. Uygun sihirbazı kullanarak CGI, WinCGI veya bir hizmet birimi DLL'ine dayalı bir uygulamanın yapısını üretebilirsiniz. Delphi'nin bu sürümüne sahip değilseniz, muhtemelen üçüncü parti bileşenler kullanabilirsiniz

CGI, ya da Common Gateway Interface, istek birimi girişini işlemek için son derece geniş bir kullanımı olan bir Internet tekniğidir ve ÇOĞU Web hizmet birimi tarafından desteklenir. Temel olarak hizmet birimi CGI programıyla ortam değişkenlerini kullanarak iletişim kurar ve kullanıcı girişini (komuta bağlı olarak) komut satırı parametreleri aracılığı ile sunar. Bu teknik UNIX giriş/çıkış modeli üzerine kurulmuştur ve pek de etkin değildir. Bir CGI uygulamasını, bağımsız bir konsol uygulaması olarak oluşturursunuz.<sup>1</sup>

### 1.4 CGI Programcıklarınız için bir Programlama Dili Seçme

CGI programcıkları ağgeçit programcıkları (gateway scripts) olarak adlandırılır. Programcık terimi shell programcığının arkasındaki UNIX ortamından gelmektedir, ancak ağgeçit programcıkları bir programcık formatında olmak zorunda değildir. Ağgeçit programcıklarını çalıştırılabilir bir dosya yaratabilen herhangi bir programlama dilinde yazabilirsiniz. En çok kullanılan diller şunlardır:

---

<sup>1</sup> CANTU,Marco,Delphi 3,sayfa 991-998,1997

Bourne Shell

C Shell

C/C++

Perl

Python

Tel

Visual Basic

Programcık yazma dillerinin önde gelen ikisi:

JavaScript

VBScript

Programcıklarınızı yazmak için kullanabileceğiniz en iyi programlama dili tabii ki Web sunucunuz üzerinde kullanılabilen ve ihtiyaçlarınıza cevap verebilen dil olacaktır. Tercihen bu dil Web sunucunuz üzerinde zaten mevcut olmalıdır ve siz bu dili profesyonel bir şekilde kullanabiliyor ya da en azından bu dil hakkında bir bilgi birikimine sahip olmalısınız. Şunu da unutmayın ki, çoğu kullanıcı bilgilerini metin olarak girer. Bu nedenle kullanacağınız dil karakter katarlarını desteklemeli ve onlarla ilgili önemli işlemleri yapabilmelidir.

Dilin uygun olup olmadığını öğrenmenin en kolay yolu bunu sunucu sistemden sorumlu sistem yöneticisine ya da Webmaster'a sormaktır. UNIX sistemleri üzerinde çoğu Web sunucusunda olduğu gibi aşağıdaki UNIX komutlarını kullanarak hangi dillerin sisteminizde kullanılabilir olduğunu öğrenebilirsiniz.

which

where is

which ve whereis komutlarının her ikisi de UNIX sistemlerinde kullanılabilir. Bu komutları shell komut satırından girebilirsiniz, komutun ardından aramayı ne üzerine yapacağınızı belirleyen bin anahtar sözcüğü, örneğin kullanmayı istediğiniz

programlama dilinin adını, girmelisiniz. UNIX sunucunuzun Perl'i destekleyip desteklemediğini belirlemek için aşağıdaki komut satırlarından birini girmelisiniz:

```
which perl
```

ya da

```
whereis perl
```

Perl, C/C++ ve UNIX shell gibi sık kullanılan popüler programcık yazma dilleri ilendeki bölümlerde, neden ve ne zaman kullanılmaları gerektiği üzerinde durularak anlatılacaktır. Her bölüm niteliklerin ve sistem desteklerinin bir listesini içerir. Sistem destekleri şöyle özetlenebilir:

İşletim Sistemi Desteği— Dilin kullanıldığı işletim sistemi

Programlama Düzeyi— Dili kullanma ve öğrenme zorlukları

İşleme Karmaşıklığı— Dilin bin işlemi gerçekleştirme karmaşıklığı

Metin İşleme Yetenekleri— Metin ve katarları işleme yeteneği

Ortak programcık dilleri ile ilgili bu başlıktan sonra en yeni iki programcık yazma dillerinden olan JavaScript ve VBScripts incelenmektedir. Şu sıralar hem JavaScript hem de VBScript Net'te en sık konuşulan konulardır. İnternet teknolojisinin ardında kalmak istemiyorsanız gözünüzü bu dillerden ayırmamalısınız.

#### **1.4.1 UNIX Shell Kullanma**

UNIX işletim sistemi iş dünyasında, eğitim ve araştırma alanlarında geniş çapta kullanılmaktadır. Kullanılan ortama göre UNIX işletim sisteminin çok çeşitli değişkenleri bulunmaktadır. Aynı üretici tarafından üretilen ortamlarda bile UNIX işletim sisteminin farklı değişkenleri kullanılmış olabilir. Örneğin DEC'te Dec-Alpha, Decstation ve Dec OSF değişkenleri bulunur.

Bu işletim sistemlerinin ortak noktası aynı çekirdeği kullanmalarıdır. Çoğu UNIX işletim sistemi Benkeley UNIX (BSD), AT&T System V ya da BSD ve System V'in bin birleşimine dayanmaktadır. BSD ve System V'in ikisi de şu üç shell programcığı yazma dilini kullanabilmektedir

Bourne Shell

C Shell

Korn Shell

Tüm UNIX shelleri programcık yaratmak için kullanılan ve derlenmesine gerek duyulmayan diller olarak yorumlanabilir. Bourne Shell en temel shell'dir. C shell ise C programlama dilinin birçok niteliğini taşıyan geliştirilmiş bir shell'dir. Bourne Shell C Shell'den çok farklı yazım kuralları kullandığından Bourne Shell'de yazılmış bir programcık C Shell'de çalışmayacaktır. C Shell'in bu programcığı yorumlaması için programcığı yeniden ve C Shellde yazmak gerekir.

Bazı programcılar C Shell'in geliştirilmiş niteliklerini Bourne Shell'in basit yapısı ile birlikte kullanmak istemektedir. Bu ihtiyaçtan Korn Shell doğmuştur. Korn Shell en fazla tercih edilen shell'dir. Bourne Shell'in işlevselliğine sahiptir ve C Shell'in bazı niteliklerini desteklemektedir. Bourne Shell'de yazılmış bir programcık Korn Shell tarafından doğrudan yorumlanabilir. Böylece Korn Shell'i kullanabilme imkanına kavuştuğunuzda Bourne Shell programcıklarınızı çevirmek için zaman harcamanız gerekmeyecektir. Korn Shell her geçen gün daha da popülerleşmesine rağmen Bourne Shell ve C Shell en çok kullanılan iki UNIX shell'idir.

Belli bir shell'i kullanıyorsanız Bourne, C ve Korn Shell'ler arasındaki farklar yalnızca görüntüseldir. Kullandığınız shell'i aşağıdaki komutları yazarak istediğiniz anda değiştirebilirsiniz:

`/bin/sh` Bourne Shell'e geçmek için

`/bin/csh` C Shell'e geçmek için

`/bin/ksh` Korn Shell'e geçmek için

Çeşitli sheller arasındaki gözlenebilir farklılıkları anında görmek mümkündür. Örneğin Bourne Shell'in önceden belirlenmiş komut yorumlayıcısı `$` işaretidir. C shell'in önceden belirlenmiş komut yorumlayıcısı genellikle ana bilgisayar adı ve bin boşluğun takip ettiği kullanıcı adıdır. Bunun ötesinde C Shell, Bourne Shell'de olmayan, tarih fonksiyonunu, komutların takma adlarını ve bir çok diğer kontrolü

destekler. Fakat genellikle bu tür farklılıklar CGI programcıları için önemli değildir. Sizi asıl ilgilendiren shell'in doğrudan desteklediği nitelikler ve işletildiğinde programcıların davranışlarının nasıl Olduğu olmalıdır.

Bu nedenle Bourne Shell daha çok tercih edilir. Bourne Shell diğer shell'ler içinde en küçüğü ve en etkilisidir. Bu yüzden, bir Bourne Shell programcısı genellikle daha hızlı çalışır ve daha az sayıda sistem kaynağı kullanır. Diziler gibi daha ileri nitelikler kullanmak isterseniz Korn Shell'i kullanmalısınız. Korn Shell, Bourne Shell'den daha fazla niteliğe sahiptir ve biraz daha fazla sayıda sistem kaynağına ihtiyaç duyar. Fonksiyon çağrılarını yada ilişkilendirmeleri yapmak için C Shell'i kullanmalısınız. C Shell, Bourne ve Korn Shell'den daha büyük olduğundan genellikle C Shell'de yazılmış programcılar daha fazla yetenek ve sistem kaynağı kullanımı gerektirir.

UNIX shell yeni metin işlemek için; sed, avk ve grep gibi hazır uygulamalar bulundurmasına rağmen bunlar geleneksel programlama dilleri gibi güçlü ve geliştirilebilir değildir. Basit işler gerçekleştirmek ve nispeten ileri düzeyde bir metni ya da dosyayı işlemek isterseniz shell programcılarını kullanmayı tercih etmelisiniz.

İşletim Sistemi Desteği: UNIX

Programlama Düzeyi: Temel

İşleme Karmaşıklığı: Basit

Metin İşleme Yeteneği: Orta Derecede Gelişmiş

#### **1.4.2 C/C++ Kullanma**

Karmaşık işleri gerçekleştirmek için kendi programcılarınızı kullanmak istiyorsanız büyük silahlarla donatılmışsınız demektir. CGL programcıları için en çok kullanılan iki gelişmiş dil C ve C++'tır. C, bugün kullanılan en popüler programlama dilidir. C-ı-+ ise C'nin nesne-tabanlı kuşağıdır. Hem C ve hem de C++ ile kullanılmadan önce derlenmesi gereken programlar yaratılabilir. C ve C++'ın en büyük avantajı her türlü ortamda kullanılabilir olmalarıdır.

C kullandığınızda programcılarınızı işletebilmek ve minimum sistem kaynağı kullanabilmek için bir hazırlık zamanı gerekir. C 20 yıl önce geliştirildi ve o zamandan beri her an popülerliği artıyor. CGI programcıları için C'yi tercih etmelerinin nedeni,



derlenmiş C programcıklarının benzer işlemlere sahip diğer dillere göre çok küçük oluşudur. Küçük programlar minimum sistem kaynağı kullanırlar ve hızlı çalışırlar. Ancak C'deki metin işleme işlevlerinin zor olması nedeniyle C karmaşık bir dildir. Bu nedenle C'de uzman değilseniz, C'nin ileri metin işleme fonksiyonlarını kullanırken dikkatli olmalısınız.

Uzun vadede geliştirme maliyeti ana konu olduğunda yazılmış olan programcıkların yeniden kullanımı da söz konusudur. C++ nesne tabanlı bir dil olarak birçok fonksiyonu kullanmanızı sağlayan bir kütüphaneye sahiptir. Bu fonksiyonlar sizin programcığınızın çekirdeğini oluşturur ve diğer CGI programcıklarında yeniden kullanılabilirler. Örneğin, kullanıcı girişlerini sıralayan bir fonksiyon, bir girişi kullanarak veri tabanını arayan bir fonksiyon ya da bir çıkışı bir HTML dökümanı olarak gösterecek bir fonksiyon kullanabilirsiniz. Nesne tabanlı C++ diğer dillerden oldukça farklıdır. Daha önce nesne tabanlı bir dil kullanmadıysanız, C'yi de tanımıyor ve C++ kullanmayı planlıyorsanız bir öğrenme sürecine girmeye hazırlıklı olun.

İşletim Sistemi Desteği	:	UNIX, DOS, Windows, MAC ve diğerleri
Programlama Düzeyi	:	Gelişmiş
İşleme Karmaşıklığı	:	İleri
Metin İşleme Yeteneği	:	Kullanımı zor

### 1.4.3 Perl Kullanma

CGI programlama işine girmişseniz Perl'i öğrenmeli ve kullanmalısınız. Perl (Practical Extraction and Report Language - Pratik Çıkarım ve Raporlama Dili); C elemanlarını ve awk, sed ve grep gibi UNIX shell niteliklerini metin katarlarını işleme ve rapor yaratma için güçlü bir dil yaratmak üzere birleştirmektedir. CGI programcıkları tarafından gerçekleştirilen metin işleme yöntemlerine yenilikler getirdiği için geniş çapta kullanılmaktadır. C ve C++ gibi Perl de her ortamda kullanılabildiği için tercih edilmektedir. Perl sürümleri düşünebileceğiniz her işletim sisteminde kullanılabilmektedir. Perl'i aşağıdaki amaçlar için kullanabilirsiniz:

- Dosyaları, metinleri ve işlemleri kolayca işlemek
- Metinlerden katan çıkarmak ve onları karmaşık yollarla işlemek
- Kolay ve hızlı bir şekilde dosya, veritabanı ve indeks aramak

- Veri sorgulama sonuçları ile ilgili ileri düzeyde rapor basmak

Perl, Bourne ve C Shell gibi yorumlanan bir dildir. Ancak Perl birçok yorumlanan dil gibi sınırlamalar içermez. Çok büyük miktarlarda veriyi işlemek, çok karmaşık desen uydurma tekniklerini kullanmak için Perl kullanabilirsiniz, dosyaları çok hızlı bir şekilde tarayabilirsiniz. Perl katarlarının uzunluk sınırlaması yoktur, bütün bir dosya bir tek katan olarak kullanılabilir. Yazım kuralları C'ye çok benzemektedir. i f, for ve whi le gibi komutları aynen C'de kullandığınız gibi kullanabilirsiniz.Şaşırtıcıdır ki Perl, özellikle C veya UNIX shell hakkında temel bilgilere sahip olduğunuz takdirde öğrenilmesi ve kullanılması kolay bir dildir. Perl programcıları genellikle UNIX shell programcılarından daha hızlı ve derlenmiş C/C++ programcılarından biraz daha yavaşlardır. Büyük miktarlarda metin işlemek için Perl'i kullanabilirsiniz.

İşletim Sistemi Desteği : UNIX, DOS, Windows, MAC ve diğerleri  
Programlama Düzeyi : Gelişmiş  
İşleme Karmaşıklığı : İleri  
Metin İşleme Yeteneği : Kullanımı kolay

#### 1.4.4 JavaScript Kullanma

JavaScript yeni bir programcık yazma dilidir ve Sun System tarafından geliştirilmiş olan Java programlama diline dayanmaktadır. Netscape Communication tarafından geliştirilmiştir ve tahmin edebileceğiniz gibi Netscape Navigator 2.0, JavaScript'in tüm niteliklerini desteklemektedir.

Netscape Navigator 2.0 HTML sayfasına gömülü JavaScript programcılarını doğrudan yorumlamaktadır ve Java programcıları gibi bu programlar tam anlamıyla etkileşimlidir. JavaScript fare tıklamalarını, giriş formlarını ve sayfa yönetimini tanımakta ve desteklemektedir. Bu, sayfanız kullanıcı girişlerine zekice yanıt verebilir anlamına gelir. JavaScripts, bir kaç önemli farkla Java programlama diline benzemektedir. Aşağıda karşılaştırma yaparken kullanılacak bir liste verilmiştir:

JavaScript;

- Yorumlayıcı bir dildir.
- Nesneye dayalı bir dildir: sınıflar ve kalıtım özelliği yoktur.

- Kodlar HTML'e gömülüdür.
- Önceden tanımlanmamış değişken veri tipleri ile gevşek yazım (Loose typing).
- Çalışma zamanı sırasında nesne referansları kontrol edilir.
- Güvenlik: Harddisk'e yazılamaz.

## Java

- İstemci sistemde çalışma öncesi sekizli kodda derlenir.
- Nesne tabanlıdır: Sınıflar kalıtım ile birlikte kullanılırlar.
- Programcık kodu HTML'den ayrıdır.
- Önceden tanımlanmış değişken veri tipleri ile sıkı yazım (strong typing).
- Nesne referansları derleme sırasında yapılmalıdır.
- Güvenlik: Harddisk'e yazılamaz.

JavaScript, Java dilinin tamamlayıcısıdır ve Web yayımcıları için müthiş nitelikler sunmaktadır. Parametreni bir Java programcığına geçiren bir JavaScript programı yaratabilirsiniz. Bu, sizin JavaScript programını Java programcıkları için kullanımı kolay bir ön uç yaratmanızı sağlar.

Bunların yanısıra Web yayımcılarının JavaScript kullanmak ve Java programcığına parametre aktarmak için sınıflar hakkında bilgi sahibi olmaları gerekmez. JavaScript Java'nın niteliklerini öğrenmek isteyen yayımcılar için basit bir çözüm içermektedir ancak genellikle yayımcılar bunu istememektedir.

### 1.4.5 VBScript Kullanma

Microsoft, VBScript ile araç geliştirenlerin birincil ihtiyacı olduğunu ispat etti. Visual Basic Script, Visual Basic'in bir alt kümesidir ve Web dökümanı üzerinde oldukça etkileşimli bir arabirim yaratmak için kullanılır. JavaScript'e benzer şekilde VBScript'te yazılan programlar HTML dökümanınızın gövdesine gömülüdür.

Visual Basic Script aynı zamanda OLE programcık yazma yönetiminin dinamik kullanımını da mümkün kılar. Web yayımcılarının, OLE (Object Linking and Embedding-Nesne Bağlama ve Gömme) programcıklarının VBScript çalışma ortamına dinamik olarak gömülebilmesini sağlar. Kısaca, bu VBScript'leri tak-çalıştır modüller gibi kullanılmasına izin verir. Örneğin bir VBScript programını bir Web dökümanınızın içine gömebilirsiniz. Gerçek bir tak-çalıştır modülü kullanıcı girişlerine dayandırılarak dinamik olarak yerleştirilebilir. Microsoft bir OLE programcığının standardını belirlemek için Internet standartlar grubu ile çalıştığından çok yakında bu gelişmeyi görebileceğiz.

VBScript'in ilk yayınlanan sürümleri çok yakında Windows (Win32 ve Win 16) ve PowerMacintos sistemlerinde kullanılacaktır. Microsoft VBScript'in Sun, HP, Digital ve IBM ortamlarındaki UNIX sürümlerini sağlamak için diğer firmalarla birlikte çalışmaktadır. Visual Basic Script bir derleyici ve ilişkilendirilmiş çalışma zamanı kütüphaneleri ile birlikte satılmaktadır.

Şaşırtıcıdır ki Microsoft, VBScript'i endüstri şirketlerine ve ortaklıklara ücretsiz lisanslamayı planlamaktadır. Fakat Microsoft VBScript'in geliştirme amaçlarını ticari olarak pazarlamayı da planlamaktadır. Şu anda Visual Basic Script'in ilk yayınlanan sürümleri geliştirmeciler tarafından değerlendirilmeye hazırdır.

### **1.5 Neden CGI Programcıkları?**

Neden ağgeçit programcıkları kullanmayı istemeniz gerektiğini konusunu merak ediyor olabilirsiniz. Bunlar gerçekten önemli konulardır. Bir programlama dilini öğrenmek kolay değildir ancak birazdan da görebileceğiniz gibi hiçbir zaman bir programın tamamını hazırlamak zorunda kalmayabilirsiniz. Bir düzine kullanıma hazır CGI programcığı Web üzerinde kullanımı serbest şekilde bulunmaktadır. İhtiyaçlarınızı karşılamak için bu mevcut programlardan faydalanabilirsiniz.

CGI programcıklarını hazır olarak kullanamamanın ilk nedeni aksi halde elle yapmanın çok zaman alıcı bir iş olmasıdır. CGI programcığı kullanmak okuyucu ve sizin için oldukça faydalıdır. Okuyucu kolaylık, cevaplarına otomatik cevaplar, geri postalamanın kolay yolları ve hızlı aramalar yapma imkanı gibi avantajlar kazanır. Ağgeçit programcıkları iş emirlerinizi, sorgulamalarınızı ve başka birçok şeyi otomatik olarak yapmanızı sağlar. CGI programcıklarının genel kullanım alanları şunlardır:

- Web sayfalarının ziyaretçilerini izlemek ve erişildikçe güncelleme numaraları sürekli olarak postalamak
- Girişleri işlemek, örneğin bir katar aramak ve aramanın sonuçlarını taşıyan dökümanı çıkış olarak göndermek gibi
- Kullanıcı tanımlamalarını ve şifre bilgilerini geçerli kılmak ve Web üzerinde girilmesi sınırlı alanlara yetkili okuyucuların erişimini sağlamak
- Görüntü haritaları girişlerini işlemek ve okuyucuyu ilişkili dökümanlara yönlendirmek
- Okuyucunun kullandığı inceleyicinin tipine dayalı dökümanlar yaratmak
- Bir veritabanına ya da indekse gönderilen cevapları incelemek ya da okuyucudan gelen geribeslemeye eklemek

## 1.6 CGI Programcıkları Nasıl Çalışır?

Ağgeçit programcıkları, Web yayınlarınızın okuyucuları tarafından geri yollanan girişlerin işlenmesi için kullanılır. Giriş genellikle Web sunucusu tarafından ağgeçit programcığına aktarılan ortam değişkenleri biçimindedir. Ortam değişkenleri sunucunun kullandığı CGI programcıklarının sürüm bilgisi, veri tipi, veri boyutu gibi aktarılmakta olan bilgi hakkında bazı değişkenleri tanımlar. Ağgeçit programcığı girişleri bir komut satırı ifadesinden ve standart bir girişten de alabilir. Bir CGI programcığını çalıştırmak için sunucu üzerinde kaynak olarak alabileceğiniz bir programcık mevcut olmalıdır. Sunucunuz, bir ağgeçit programcığını çalıştıracak ve onu kullanmayı planladığınız amaca uygun olarak konfigüre etmenizi sağlayacak yetenekte olmalıdır.

Okuyucu, programcığa bir referans içeren bağlantıyı etkinleştirerek CGI programcığına bilgi aktarır. Ağgeçit programcığı girişi işler ve sonucu Web sunucusunun kullanabileceği şekilde biçimlendirir. Web sunucusu sonucu alır ve okuyucunun inceleyicisine geri gönderir. İnceleyici çıkışı ekranda görüntüler.

Bir ağgeçit programcığı çıkışı sunucuya bir yönlendirme taşıyan bir başlıkla başlar.

Şu anda geçerli olan üç sunucu yönlendirmesi: Contenttypes (İçerik-tipi), Location (Konum) ve Status (Durum)'dur. Başlık, HTML başlığı gibi boş bir satırla izlenecek şekilde bir yönlendirme taşıyabilir. Boş satırlar başlığı, okuyucuya aktardığınız veriden ayırır. Location ve Status yönlendirmesi içeren çıkışlar genellikle tek satırdır. Bu nedenle sunucunun ihtiyacı olan Location ve Status bilgilerinin hepsi bu satırdadır ve veri içeren satırlar bulunmadığından boş satıra da ihtiyaç yoktur. Sunucu çıkışı yorumlar, ortam değişkenlerini belirler ve çıkışı istemciye iletir.

Bu nedenle sunucu ve istemci arasında gerçekleşen olaylar bir çok aşama içerir. Bu aşamalar şu şekilde özetlenebilir:

1. İstemci, girişi sunucuya geçirir.
2. Sunucu, girişe uygun olacak şekilde ortam değişkenlerini ayarlar.
3. Sunucu, girişi CGI programcığının adı ile bir değişken gibi geçirir.
4. Sunucu, eğer varsa, komut satırı girişini ya da standart giriş katarını CGI programcığına geçirir.
5. Programcık, girişi işler.
6. Programcık, çıkışı sunucuya döndürür. Bu çıkış daima belirli bir başlığı taşır ve eğer ek veri varsa, bir de gövdesi bulunur.
7. Sunucu, ortam değişkenlerini çıkışa uygun olarak ayarlar.
8. Sunucu, çıkışı istemciye geçirir. <sup>2</sup>

### **1.7 CGI Programcıklarına Girdi**

Bir kullanıcı bir ağgeçit programcığına ilişkin bir bağlantıyı etkinleştğinde giriş sunucuya gönderilir. Sunucu bu verilen ortam değişkenleri olarak biçimlendirir ve ek bir yerinin komut satırı ya da standart bir giriş katan yoluyla gönderilip gönderilmediğini kontrol eder.

### **1.8 Ortam Değişkenleri**

CGI programcıklarına giriş genellikle ortam değişkenleri biçimindedir. Ağgeçit

programcıklarına geçirilen ortam deęişkenleri inceleyicinin sunucudan istedięi bilgi ile ilişkilidir. Sunucu isteęi işler ve veri isteęin içinde geçer. Ortam deęişkenleri büyük-küçük harfe duyarlıdır ve genellikle bu bölümde tarif edildięi gibi kullanılır. Bazı ortam deęişkenleri sisteme özel olmakla birlikte, çoęu standarttır. Standart ortam deęişkenleri ařaęıdaki Tabloda verilmiştir.

Daha sonraki örnek okuyucu giriři sunucuya geçer geçmez ortam deęişkenlerinin otomatik olarak ayarlanmasını göstermektedir. Bu deęişkenler hakkında bilgi edinecek girişlerin CGI programcıklarına nasıl geçtiğini daha iyi anlayabilirsiniz. Ayrıca gerektiğinde bu deęişkenleri istediğiniz şekilde ayarlamamanın avantajlarını kullanmayı da öğrenebilirsiniz.

### **1.9 Standart Ortam Deęişkenleri**

**AUTH\_TYPE:**Doęrulama yöntemini belirler ve kullanıcı erişimlerini geçerli kılmak için kullanılır.

**CONTENT\_LENGTH :**Veri uzunluęunu sayısal bir deęer olarak izlemek için kullanılır.

**CONTENT\_TYPE:** MIME tipi verilere işaret eder.

**GATEWAY\_INTERFACE :**Sunucunun kullandığı CGI standardının sünümüne işaret eder.

**HTTP\_ACCEPT:**Sunucu tarafından aęgeçit programcıęına geçinilirken inceleyicinin kabul edebileceęi MIME içerik tiplerine (Content..Types) işaret eder.

**HTTP\_USER\_AGENT:**Sunucu tarafından aęgeçit programcıęına geçiriirken isteklerin gönderilmesi için kullanılan inceleyicinin tipine işaret eden

**PATH\_INFO:**CGI programcıęının tanınmasından sonra URL içinde bulunan ek bilgiye işaret eder.

**PATH\_TRANSLATED:**PATH\_INFO deęişkenine dayanan sunucu tarafından ayarlanan bir deęişkendir. Sunucu PATH\_INFO deęişkenine göre bu deęişkeni ayarlar.

**QUERY\_STRING:**Sorgulama katarını ayanlar (URL bir sorgulama taşıyorsa).

---

<sup>2</sup> STANEK,William Robert,HTML JAVA CGI VRML SGML - UNLEASHED,s. 344-352,1997.

REMOTE\_ADDR:İstekte bulunan uzaktaki bilgisayarın Internet Protokolu adresini tanımlar.

REMOTE\_HOST:İstekte bulunan makinanın adını belirler.

REMOTE\_IDENT:İstekte bulunan makinayı belirler.

REMOTE\_USER:Kullanıcı tarafından doğrulanmak üzere kullanıcı adını belirler.

REQUEST\_METHOD: İsteğin yapılma yöntemini belirler.

SCRIPTNAME:Çalıştırılacak programcığın görüntü adresini tanımlar.

SERVER\_NAME:Sunucunun evsahibi adını, takma adını ya da İP adresini tanımlar.

SERVER\_PORT:İsteklerin kabul edildiği sunucunun port numarasını tanımlar.

SERVER\_PROTOCOL: Sunucuya gönderilen isteklerin protokolünü tanımlar.

SERVER\_SOFTWARE :Web sunucu yazılımını tanımlar.

### **1.9.1 AUTH\_TYPE**

AUTH\_TYPE değişkeni Web sunucusuna ayrılmış alanda erişim kontrolü sağlamak için kullanılır. Yalnızca kullanıcı doğrulama desteği için kullanılır. Web merkezinde erişim kontrolü bulunmayan bir alan varsa AUTH\_TYPE değişkeni kullanılmakta olan doğrulama şemasını tanımlayacak özel bir değere ayarlanır. Aksi halde bu değişkene herhangi bir değer ilişkilendirilmez. HTTP'nin şimdiki sürümleni içinde basit bir sorgulama-cevaplama doğrulama yöntemi geliştirilmektedir.Sunucu bu mekanizmayı kullanarak istemcinin bir isteğini sorgulayabilir ve istemci de bu sorguyu cevaplayabilir. Bu, sunucu tarafından AUTHTYPE değişkenine bir değer atanarak sağlanır. İstemci de uygun değeri sağlar. Sonraki adım kullanıcıyı doğrulamaktır. Kullanıcının inceleyicisi, basit bir doğrulama şeması kullanarak tek bir kullanıcı tanımlayan doğrulama bilgisini sağlamalıdır. Bu bilgi bir kullanıcı ID'si ve şifre içerir.

HTTP'nin şu andaki sürümleninden HTTP 1.0'ın basit doğrulama şeması en sık kullanılan doğrulama yöntemidir. Bu yöntemi AUTH\_TYPE değişkenini ayarlayarak belirlemek için şu atama yapılmalıdır:



AUTH\_TYPE= Basic

### 1.9.2 CONTENT\_LENGTH

Veri katarının uzunluğunu izlemek için kullanılır. İstemci ve sunucunun standart giriş katarı olarak ne kadar veri okuyacağını belirtir. Değeri istek ise aktarılan venideki karakter sayısı ile ilişkilidir. Eğer veri aktarılmazsa değer almaz.

Karakterler sekizlilerle ifade edildiği sürece CONTENT\_LENGTH değişkeninin değeri standart giriş ya da çıkıştan geçen karakter sayısına eşit olacaktır. 0 halde eğer 25 karakter aktarıldıysa ve karakterler sekizlilerle ifade ediliyorsa CONTENT\_LENGTH değişkeninin değeri aşağıdaki gibi ayarlanmalıdır.

CONTENT\_LENGTH = 25

### 1.9.3 CONTENT\_TYPE

CONTENT\_TYPE değişkeninin değeri yerinin MIME tipine işaret eder. MIME tipi bir HTTP 1.0 niteliğidir ve HTTP 0.9 kullanan sunucularda kullanılamaz. Bu değişken, aktarılan veri standart giriş ya da çıkış katarını kullanıyorsa anlamlıdır. Bu değişkene iliştilen değer, MIME tipini ve alttipini aşağıdaki gibi belirler:

CONTENT\_TYPE = type/subtype

MIME tipleri basit gruplara ayrılabilir. Her grup bir alttip ile ilişkilendirilmiştir. Temel MIME tipleri ve onların tanımları aşağıdaki tabloda verilmiştir.

### 1.9.4 Basit MIME Tipleri

Tip	Tanım
application	Bir başka uygulama ile kullanılabilen ya da çalıştırılabilen ikili veri audi 0 Önceden genel görünüm hakkındabir fikir edinmek için çıkış cihazının ihtiyaç duyduğu ses dosyası
image	Önceden genel görünüm hakkındabir fikir edinmek için çıkış cihazının ihtiyaç duyduğu görüntü dosyası
message	paketlenmiş bir posta mesajı
multipart	Birçok veri tipini mümkün kılan çoklu parçalardan oluşan veriler text

Herhangi bir karakter seti ya da dil formatı ile gösterilebilen metinsel veri

video Önceden genel görünüm hakkındabir fikir edinmek için çıkış cihazının ihtiyaç duyduğu video dosyası

x-world Dünya dosyaları için deneysel veri tipi

MIME alttipleri asıl veri tipleri olarak ve veri tiplenine ile genişletilmiş veri tipleri-ne ek olarak tanımlanır. Asıl alttip, MIME Content-Type (MIME İçerik-Tipi) olarak kullanılmak üzere asıl veri tipi olarak uyarlanır. Ayrıca tanımlanmış veri tipleri ek alttiplendir ve resmen MIME Content-Type (MIME İçerik-Tipi) olarak kullanılırlar. Genişletilmiş alttipleri de kolaylıkla kullanabilirsiniz, çünkü bunlar kısa çizgi ile takip edilen bir x harfi ile başlarlar. Tablo 11.3 sık kullanılan MIME tiplerini ve onların tanımlanını listelemektedir.

### 1.9.5 Sık Kullanılan MIME Tipleri

application/mac-binhex40 Macintosh ikili formatta veri

application/msword Microsoft Word dökümanı

application/octet-stream Çalıştırılabilen ya da başka herhangi bir uygulama ile birlikte kullanılabilen ikili veri

application/pdf ACROBAT PDF dökümanı

application/postscript Postscript formatında veri

application/rft Rich Text Format (Zengin Metin Formatı RET) dökümanı

application x-compressUNIX compress kullanarak sıkıştırılmış döküman

application/x-dvi Cihaz bağımsız dosya

application/x-gzip UNIX gzip kullanılarak sıkıştırılmış döküman

application/x-latex LaTeX dökümanı

application//x-tar	UNIX tar kullanılarak arşivlenmiş veriler
application/x- zip-compres	PKZİP kullanılarak sıkıştırılmış veriler
audio/basic	Tanımsız formatta ses venisi
audio/x-aiff	Apple AJFF formatında ses venisi
audio/x-wav	Microsoft WAV formatında ses venisi
image/gif	GIF formatında bir görüntü venisi
image/jpeg	JPEQ fonmatında bin görüntü venisi
image/tiff	TIFF fonmatında bir görüntü
image/x-portable-bitmap	Portable bitmap
image/x-portable-graymap	Portable graymap
image/x-portable-pixmap	Portable pixmap
image/x-xbitmap	X-bitmap
image/x-xpixmap	X-pixmap
message/external body	Harici veri kaynaklı mesaj
message/partial	Bölmelere ayrılmış ve kısmi mesaj
message/rfc822	RFC-822 uyumlu mesaj
multipart/alternative	Alternatif formatlarda veri
multipart/digest	Çok parçalı mesaj yazısı
multipart/mixed	Çeşitli formatlardaki veriyle birlikte çok parçalı mesaj
multipart/parallel	Parçaları eş zamanlı görüntülenmesi gereken çok parçalı veri
text/html	HTML formatlı metin

text/plain	HTML formatı içermeyen düz metin
video/mpeg	MPEG formatında video
video/quicktime	Apple QuickTime formatında video
video/x-msvideo	Microsoft AVI formatında video
x-world/x-vrml	VRML dünya dosyası

Bazı MIME Content-Type (MIME İçerik-Tipi)'lan ek parametrelerle kullanılabilir. Bunlar arasında text/plain, text/html ve tüm çok parçalı mesaj yenileri vardır. CharSet parametresi, text/plain tipiyle birlikte, veri için kullanılacak karakter setini belirlemek için kullanılır. Version panametnesi, text/html tipiyle HTML'nin hangi sürümünün kullanıldığının belirtilmesi için kullanılır. Boundary parametresi ise çok parçalı yerlerle mesajın parçalarını ayıran sınır ayarlarını belirlemek için kullanılır.

Text/plain tipi için kullanılan charset parametresi isteğe bağlıdır. Eğer belirtilmezse, varsayılan değer us-ascii olur. CharSet için geçerli diğer değerler, Uluslararası Standartlar Kurumu tarafından onaylanmış tüm karakter setleridir. Bu karakter setleri ISO-8859-1 ila ISO-8859-9 tarafından tanımlanırlar ve aşağıdaki gibi belirtilirler:

```
CONTENT_TYPE = text/plain; charset= iso-8859-1
```

text/html tipi için version parametresi isteğe bağlıdır. Eğer parametreye değer atanırsa, veniyi okuyan tarayıcı eğer HTML'in o sürümünü destekliyorsa veriyi yorumlayabilir. Aşağıdaki metin HTML 3.0 özelliklerini belirler:

```
CONTENT_TYPE = text/html; version= 3.0
```

Çok parçalı mesaj tipleri için boundany parametresi gereklidir. Bu parametre 1 ile 70 karakter arasında uzunluğu olan bir katan şeklinde ayarlanmıştır. Katan, boşluk karakteri ile sonlanamaz ama her türlü sayıyı, sınırlı bir grup özel karakteri ve boşluk karakterini içerebilir. Boundary parametresi tek olacak şekilde aşağıdaki biçimde tanımlanır:

```
CONTENT_TYPE = multipart/mixed; boundary= boundary_katarı
```

### 1.9.6 GATEWAY\_INTERFACE

GATEWAY\_INTERFACE deęişkeni sunucunun kullandığı CGI yazılımının sürümüne işaret eder. Bu deęişkenin değeri aşağıdaki gibi sürümün adından ve sürüm numarasından oluşur:

GATEWAY\_INTERFACE = name/version

CGI yazılımının son sürüm numarası 1.1'dir. Bu sürümü sağlayan sunucunun GATEWAY\_INTERFACE deęişkeni şöyle olmalıdır:

GATEWAY\_INTERFACE = CGI/1.1

### 1.9.7 HTTP\_ACCEPT

HTTP\_ACCEPT deęişkeni istemcinin kabul edeceği yerinin tipini tanımlar. Kabul edilebilir değerler bin tip/alttip çifti ile belirlenir. Her bir tip/alttip çifti virgülle ayrılır. Çoğu istemci düzinelerce MIME tipini desteklemektedir.

### 1.9.8 HTTP\_USER\_AGENT

HTTP\_USER\_AGENT deęişkeni istek yollamak için kullanılan inceleyicinin tipini belirler. Kabul edilebilir değerler yazılım tipi/sürüm ya da kütüphane/sürüm şeklinde verilir. Aşağıdaki örnekte HTTP\_USER\_AGENT deęişkeni Netscape Navigator sürüm 2.0'ı ifade etmektedir:

HTTP\_USER\_AGENT = Mozilla/2.0

Görüldüğü gibi Netscape kendini ifade etmek için Mozilla takma adını(alias) kullanmaktadır. Bu deęişkenin tipini belirlediği ilk istemci tipleri inceleyiciler, Web spiders ve robots'tur. Bin programcığa erişirken kullanılan istemcinin tipini belirlemek için kullanılması gereken bin parametredir ve her istemcinin bu parametreyi ayarlaması gerekmediği unutulmamalıdır.

Burada popüler inceleyiciler tarafından kullanılan yazılım tipleri listelenmiştir:

- Arena
- Enhanced NSCA Mosaic

- Lynx
- Mac Web
- Mozilla
- NSCA Mosaic
- NetCruiser
- WebExplorer
- WinMosaic

Bazı Web Spider'ları da aşağıda listelenmiştir:

- Lycos
- MOMSpider
- WebCrawler

### **1.9.9 PATH\_INFO**

PATHINFO değişkeni fazladan verilen bin yol bilgisidir, ağgeçit programcısına ek bilgi göndermek için kullanılabilir. Bu ekstra yol bilgisi kaynak ağgeçit programcısının URL'sini takip eder. Genellikle bu bilgi, sunucunun yorumlaması gereken kaynağın görüntüsel ya da bağlı bin yol tanımıdır. CGI programcığınızın URL'si aşağıdaki gibidir:

```
/usr/cgi -bin/formparse.pl/home.html
```

PATHINFO değişkeni aşağıdaki gibi ayarlanabilir:

```
PATHINFO = /home.html
```

### **1.9.10 PATH\_TRANSLATED**

Sunucular PATH\_INFO değişkenini PATH\_TRANSLATED değişkenine çevirirler. Bu işlem öngörülen Web dökümanının yol tanımını fazlalık yol tanımının önüne eklemek sureti ile yapılır. Örneğin PATH\_INFO değişkeni home.html ve önceden belirlenmiş

yol tanımı /usr/documents/pubs ise PATH\_TRANSLATED değişkeni şu biçimdedir:

```
PATH_TRANSLATED = /usr/documents/pubs/home.html
```

### 1.9.11 QUERY\_STRING

QUERY\_STRING değişkeni URL-kodlu arama katarlarını belirtir. Bu değişken, doldurulan bir formun işlerken GET yöntemi kullanıldığında ya da bir dökümanı aramak için ISINDEX sorgulaması kullanıldığında ayarlanır. Bu sorgulama katanı URL'den bir soru işareti ile ayrılır. Kullanıcı tarafından işleme sokulan sorgulama katanı URL'den soru işareti ile ayrılan bilginin yapısı aşağıdaki biçimdedir:

```
/usr/cgi-bin/formparse.pl ?katar
```

Sorgulama katanı URL kodlu ise inceleyici katanın anahtar parçalarının kodunu çözen. Artı işareti sözcükler arasında yer tutucu olarak kullanılır ve boşluk karakteri yerine kullanılır. Örneğin;

```
/usr/cgi-bin/formparse.pl?sozcuk1+sozcuk2+sozcuk3
```

Eşittir işareti yayımcının belirlediği anahtarlar ile kullanıcının girdiği cevapları birbirinden ayırır. Aşağıdaki örnekte cevap anahtar sözcüğü asla kullanıcı cevabından eşittir işareti ile ayrılmıştır:

```
/usr/cgi-bin/formparse.pl ?cevap=asl a
```

Ampersand (ve) işaretleri anahtar ve değer kümelerini birbirinden ayırın. Aşağıdaki örnekte yayımcının belirlediği birinci anahtar sözcük "cevap". kullanıcının girdiği cevap ise "bazen". İkinci anahtar sözcük "neden", kullanıcı cevabı ise "ı am not really sure"dur.

```
/usr/cgi-bin/formparse.pl?cevap=bazen~~neden=ı+am+not+really+sure
```

Son olarak yüzde işareti, escape karakterlerini belirlemek için kullanılır. Aşağıdaki örnekte yüzde işareti onaltı tabanında bir veriye işaret etmek üzere belirlenmiş bin karakter olarak kullanılmaktadır. Yukarıdaki örnekteki ikinci cevapta kesme işaretinin yerine kodunu yazmak için yüzde işareti kullanılabilir.

```
/usr/cgi-bin/formparse.pl?cevap=bazen&neden=ı+%27m+not+really+sure
```

### **1.9.12 REMOTE\_ADDR**

REMOTE\_ADDR deęişkeni istekte bulunan uzaktaki bilgisayarın Internet Protocol adresini belirler. İP adresi aę üzerinde alıřan bilgisayarlar iin sayısal bir belirleyicidir. Bu deęişken ařaęıdaki gibi kullanılabilir:

```
REMOTE_ADDR= 205.1.20.11
```

### **1.9.13 REMOTE\_HOST**

REMOTE\_HOST deęişkeni istekte bulunan ev sahibi bilgisayarın adını belirler. Yalnızca sunucu ters adres bulma işlemini gerçekleştirebiliyorsa bu deęişken ayarlanabilir. Bu deęişken ayarlanmışsa ařaęıda görüldüęü gibi tam alan adı ve ev sahibi bilgisayar adı kullanılır:

```
REMOTE_HOST = www.tvp.com
```

### **1.9.14 REMOTE\_IDENT**

REMOTE\_IDENT deęişkeni istekte bulunan uzaktaki kullanıcıyı belirler. Bu deęişken yalnızca sunucu ve istekte bulunan uzaktaki bilgisayar tanımlama protokolunu destekliyorlarsa ayarlanır. Ayrıca uzaktaki kullanıcıya ait bilgiler her zaman kullanılabilir durumda değildir ve kullanılabilir olsa da güvenilir değildir. Ařaęıdaki örnekte görüldüęü gibi, deęişken ayarlandığında ilişkili deęer, alan bilgilerini de ieren tam tanımlanmış isimdir.

```
REMOTE_IDENT = william.www.tvp.com
```

### **1.9.15 REMOTE\_USER**

REMOTE\_USER deęişkeni kullanıcıya baęlı bir kullanıcı adıdır ve kullanıcının kim olduęunu belirlerken güvenebileceğiniz tek deęişkendir. Bu deęişken de dięer kullanıcı kaynaklı deęişkenler gibi, sunucu kullanıcıya bu tür bir destek sağladığı takdirde deęer alabilir. Bu durumda aęgeit programcığı korunur. Eęer deęişken ayarlanmışsa istemci tarafından sunucuya ařaęıdaki şekilde gönderilir:

```
REMOTE_USER = william
```

### **1.9.16 REQUEST\_METHOD**

REQUEST\_METHOD isteęin hangi yöntemle yapılacaęını belirtir. HTTP 1.0'da



aşağıdaki yöntemlerden herhangi biri seçilebilir.

GET

HEAD

POST

PUT

DELETE

LINK

UNLINK

GET, HEAD ve POST yöntemleri en sık kullanılan istek yöntemleridir. GET ve POST birbirlerinin yerine kullanılabilirler. HEAD yöntemi şöyle kullanılır:

```
REQUEST_METHOD = HEAD
```

#### **1.9.17 SCRIPT\_NAME**

SCRIPT\_NAME değişkeni çalıştırılan programcığın görüntü adresini belirler. Bu bilgi programcık bin HTML dökümünü yarattığında kullanışlıdır. HTML dökümanınızdaki URL aşağıda gösterildiği gibi ise

```
http://tvp.com/cgi-bin/formparse.pl
```

SCRIPT\_NAME değişkeni de aşağıdaki gibidir:

```
SCRIPT_NAME = /cgi-bin/formparse.pl
```

#### **1.9.18 SERVER\_NAME**

SERVER\_NAME değişkeni sunucunun adını ev sahibi adı, takma adını ya da İP adresini belirler. Her zaman tek bir değer alır.

```
Server_NAME = tvp.com
```

#### **1.9.19 SERVER\_PORT**

SERVER\_PORT değişkeni sunucunun istekleri kabul ettiği portun numarasını

belinler. Bu bilgi gerekirse URL'den programcıya aktarılabilir. Fakat sunucu önceden belirlenmiş değer olarak HTTP istekleri için 80 numaralı portu kullanır. HTML dökümanları için belirlenmiş URL aşağıdaki gibi ise:

```
http://www.nasca.edu:8080/cgi-bin/formparse.pl
```

SERVERPORT değişkeni aşağıdaki gibi ayarlanır:

```
SERVER_PORT = 8080
```

### **1.9.20 SERVER\_PROTOCOL**

SERVER\_PROTOCOL değişkeni istek gönderirken kullanılacak protokolü belirler. Bu değişkene iliştilenmiş değer, kullanılan protokolün adını ve sürümünü ifade eden. sunucu/sürüm biçimindedir örneğin; HTTP/1.0. Değişkene aşağıdaki gibi değer atanır.

```
SERVER_PROTOCOL = HTTP/0.9
```

### **1.9.21 SERVER\_SOFTWARE**

SERVER\_SOFTWARE değişkeni sunucu yazılımının adını ve sürümünü belirler. Değeri belirleyen ifade ad/sürüm şeklinde olmalıdır örneğin; CERN/2.17. Değişken aşağıdaki gibi değer alır:

```
SERVER_SOFTWARE = NSCA/1.4
```

## **1.10 CGI Komut Satırı**

UNIX ortamlarında kullanıcılar komut satırı olarak bilinen alandan bilgi aktarabilirler ve programcılarını çalıştırabilirler. Komut satırında yazılan bilgi genellikle bir yorumlayıcıya bir işlemi tamamlaması için kullanacağı bir giriş olarak aktarılır. Web yayınına bağlı olarak, komut satırı genellikle İSINDEX sorgulamasını gerçekleştirmek için kullanılır. İSINDEX sorgulaması dökümanınıza etkileşimli aramalar eklemenin en kolay yoludur.

İSINDEX sorgulamaları çok kullanışlıdır. Ancak sunucunuzun bu sorgulamayı desteklemesi gerekir. HTML 2.0 altında bir İSINDEX sorgulamasını gerçekleştirmenin yedi adımı bulunmaktadır. Bu adımlar aşağıda verilmiştir.

1.Kullanıcı ağgeçit programcıđına ait bir bađlantıyı tıklar ve istemci bunu sunucuya iletir. Arama için bir ağgeçit programcıđına yapılacak bađlantı ađađıdaki gibi belirtilebilir:

```
<A HREF" /cgi-bin/search.pl">Dokuman Arama</A>
```

2.Sunucu giriđi ağgeçit programcıđına aktarın.

3.Programcıđa hiç deđiđken aktarılmamıđsa programcık <İSİNDEX> etiketi tađıyan bir HTML dökümanı yaratır. HTML 2.O'da <İSİNDEX> etiketinin herhangi bir niteliđi yoktur ve ađađıda görüldüđü gibi HEAD elemanlarının içinde kullanılır.

```
<HTML>
```

```
<HEAD>
```

```
<TİTLE> Aranabilen Döküman<\TİTLE>
```

```
<ISINDEX>
```

```
<\HEAD>
```

4.Kullanıcı bir arama katarı girmesi için uyarılır ve bilgiyi uygun řekilde girer.

5.Arama katarı URL kodlu bir arama katan olarak asıl programcıđa geri gönderilir.

6.Programcık arama katarını bir aramanın sonuçlarını saklayan bir dökümanı yaratmak gibi bir iđlemi gerçekteřtirmek için kullanır.

7.Sunucu sonuçları istemciye yollar.

HTML 3.0, sorgulama iđlemlerini gerçekteřtirir ve sorgulamaların indeksler üzerinden yapılmasını sađlayarak iđlemleri kolaylařtırır. İSİNDEX sorgulaması ađađıda anlatıldıđı gibi HTML 3.0 altında beř adımda gerçekteřtirilir.

1.Kullanıcı ISINDEX sorgulaması tađıyan dökümana eriřir. HTML altında ISINDEX etiketinin HREF ve PROMT olmak üzere iki niteliđi bulunmaktadır~ HREF niteliđi, sunucunun öngörülen URL'si geçersiz aramalarda bir ağgeçit programcıđı belirlemek için kullanılır. PROMPT niteliđi, küllanıcının göreçeđi önceden belirlenmiř komut istemi (prompt) tipini geçersiz kılmak için kullanılır. Bu nitelikler ađađıdaki gibi

kullanılır:

<HTML>

<HEAD>

<TITLE> Aranabilen Dokuman<\TITLE>

<ISINDEX>

<\HEAD>

2.Kullanıcı komut isteminin (prompt) arkasından bir arama katarı girer ve istemci bunu sunucuya iletir.

3.Sunucu arama katarını URL kodlu arama katarı olarak programcığa yollar.

4.Programcık arama katarını bir aramanın sonuçlarını saklayan bir döküman yaratmak gibi bir işlemi gerçekleştirmek üzere kullanır.

5.Sunucu sonuçları istemciye iletir.

ISI NDEX sorgulamaları döküman arama için en kolay çözümdür ancak en iyi çözüm değildir. Çoğu Web yayımcısı okuyucu için tam indekslenmiş dökümanlarda ya da form-tabanlı dökümanlarda arama yapma imkanı sağlamayı tercih etmektedir.

Bazı sunucu yazılımları exec komutunu kullanarak komut satırından parametre girmenize imkan sağlar. exec komutu ile dökümanınıza bir bilgi yerleştirmek için bir değeri geri göndermek gibi özel bir işlemi gerçekleştiren bir ağgeçidi doğrudan çalıştırabilirsiniz. exec komutu hakkında ilginç bir nokta da okuyucuların kaynak kodlara erişebilseler de sadece çıkış değerlerini görebilmeleridir. Bu durum kaynak kodu bilgisayarınıza yüklediklerinde bile değişmez. Bu nedenle komut açıkça her karşılaşıldığında çalıştırılır.

exec komutunu sunucuya bir komutu ya da programcıği çalıştırmasını söyler. Çalıştırılacak parametreler çift tırnak işareti içinde bulunur. Döküman istemciye aktarılmadan önce programcığın çıkışı o andaki pozisyonla yer değiştirir. Aşağıdaki HTML kodu günlük\_erisim adlı bir programcıği çalıştıran bir komut satırı parametresi içerir. Günlük..erisim programcıği belirli bir tarihte bir sayfaya yapılan erişim sayısını

tutmak için kullanılır. Bu nedenle o gün 25 ziyaret olmuşsa;

Bu sayfaya bugün 25 kez erişildi.

çıktısı alınır.

### **1.11 Standart CGI Girdisi**

Web sunucusuna gönderilen çoğu girdi ortam değişkenlerini ayarlamak için kullanılır. Bir kullanıcı bir ağgeçit programcığı tarafından işlenmek üzere gerçek veriyi gönderdiğinde, bu veri bir URL kodlu arama katarı olarak ya da standart bir giriş katarı yoluyla alınır. Sunucu gerçek yerinin veriyi gönderme yöntemi aracılığı ile nasıl işleneceğini bilir.

Standart bir giriş olarak veri göndermek veri göndermenin en doğrudan yoludur. Kısaca; sunucu, ağgeçide standart girişten kaç sekizli okuyacağını söyler. Programcık standart giriş akışını başlatır ve belirtilmiş miktarda veri okur. Uzun URL kodlu arama katarları kesilip kısaltılmasına rağmen standart giriş akımı üzerinden gönderilen veriler kesilmez. Sonuç olarak standart giriş akımı veri aktarmak için tercih edilir.

### **1.12 CGI Girdisini Temizleme**

Bir doldurma formunu yaratacağınızda bir başvuru yöntemi belirleyebilirsiniz. HTTP 1.0 altında iki başvuru yöntemi bulunmaktadır.

1.HTTP GET yöntemi URL kodlu arama katarlarını kullanır. Bir sunucu URL kodlu bir arama katarını alırken, sunucu, arama katarının değerini QUERY\_STRING değişkeni ile ilişkilendirir.

2.HTTP POST yöntemi standart giriş akımlarını kullanır. Bir sunucu standart giriş akımı aracılığı ile veri alırken, sunucu, arama katarının uzunluğunu CONTENT\_LENGTH değişkeni ile ilişkilendirir.

AD. ADRES, TELEFONNUMARASI adlı üç anahtar alanlı bir form içeren örnek bir döküman yaratalım. Programcığın URL'sinin <http://www.tvp.com/cgi-bin/survey.pl> ve kullanıcı cevabının aşağıdaki gibi olduğunu varsayın:

Sandy Brown

12 Sunny Lane WhoVille. USA

987-654-3210

GET ve POST yöntemlerini kullanarak gönderilen belirleyici bilgiler sunucu tarafından farklı biçimlerde işlenir. GET metodu kullanıldığında sunucu aşağıdaki ortam değişkenlerini ayarlar ve sonra girdiyi survey.pl programcığına aktarır.

PATH=/bin: /usr/bin /usr/etc /usr/ucb

SERVER\_SOFTWARE=CERN/3 .0

SERVER\_NAME=www.tvp.com

GATEWAY\_INTERFACE=GCI /1.1

SERVER\_PROTOCOL=HTTP/1 .0

SERVER\_PORT=80

REQUEST\_METHOD=GET

HTTP.ACCEPT=text/plain, text/html, application/rtf. application/postscript.

audio/basic. audio/x-aiff. image/gif. image/jpeg. image/tiff. video/mpeg

PATH\_INFO=

PATH\_TRANSLATED=

SCRIPPT\_NAME=/cgi -bin/survey.pl

QUERY\_STRING=AD=Sandy+Brown&ADRES=12+Sunny+Lane+WhoVi1le.+USA

&TELEFON\_NUMARASI=987-654-3210

REMOTE\_HOST=

REMOTE\_ADDR=

REMOTE\_USER=

AUTH\_TYPE=

CONTENT\_TYPE=

CONTENT\_LENGTH=

POST metodu kullanıldığında sunucu aşağıdaki ortam değişkenlerini ayarlar ve sonra girdiyi survey.pl programcısına aktarır.

PATH=/bin:/usr/bin:/usr/etc:/usr/ucb

SERVER\_SOFTWARE=CERN/3.0

SERVER\_NAME=www.tvp .com

GATEWAY\_INTERFACE=GCI /1.1

SERVER\_PROTOCOL=HTTP/1 .0

SERVER\_PORT=80

REQUEST\_METHOD=POST

HTTP.ACCEPT=text/plain, text/html. Application/rf. application/postscript,

audio/basic, audio/x-aiff. Image/gif. image/jpeg, image/tiff. video/mpeg

PATH\_INFO=

PATH\_TRANSLATED=

SCRIPT\_NAME=/cgi -bin/survey.pl

QUERY\_STRING=

REMOTE\_HOST=

REMOTE\_ADDR=

REMOTE\_USER=

AUTH\_TYPE=

CONTENT\_TYPE=application/x-www-form-urlencoded

CONTENT\_LENGTH=81

POST başvuru yöntemi kullanılarak gönderilmiş veri standart giriş akımı yoluyla ağgeçit programcığına geçirilir.

AD=Sandy+Brown&ADRES=12+Sunny+Lane+WhoVille,+USA&TELEFON\_NUMARASI=987-654-3210

### 1.13 CGI Programcıklarından Çıktı

Programcık girişi işlemeyi tamamladıktan sonra çıktıyı sunucuya döndürmelidir. Daha sonra sunucu çıktıyı istemciye geri gönderecektir. Bu çıktı, boş bir satır ve mesaj gövdesinin takip ettiği başlığı içeren bir HTTP cevabı şeklinde olacaktır. CGI başlık çıktısının formatı tam olarak belli olmakla birlikte mesaj gövdesinin formatı başlıkta belirlediğiniz durumlara uygun olacak şekilde yapılır. Örneğin mesaj gövdesi istemciye gösterilecek bir HTML dökümanını taşıyabilir.

### 1.14 CGI Başlıkları

CGI başlıkları sunucuya ilişkin bazı talimatlar içerir. Üç geçerli sunucu talimatı vardır:

Content-Type (içerik tipi)

Location (konum)

Status (durum)

Tek bir başlık sunucu talimatlarının birini ya da tümünü taşıyabilir. Bu talimatları sizin CGI programcığınız sunucuya aktarır. Normalde başlığı mesaj gövdesinden ayıran boş bir satır takip eder ancak buradaki çıkışın mesaj gövdesi yoktur.

#### 1.14.1 CGI Başlıklarında Kullanılan Content-Type'ler

Bir CGI başlığındaki Content-Types (içerik-tipi) alanları istemciye geri gönderdiğiniz yerinin MIME tipini tanımlar. Genellikle programcığın yeri çıktısı bir HTML dökümanı gibi tam formatlı bir dökümandır. Bu formatı başlık içinde aşağıdaki gibi belirleyebilirsiniz:

Content-Type: text/html



### 1.14.2 CGI Başlıklarında kullanılan Location'lar

Programcığınızın çıktısı programcığınız içinde yaratılan bir döküman olmak zorunda değildir. Location (konum) alanını kullanılarak Web üzerindeki herhangi bir dökümana erişebilirsiniz. Location alanı, URL aracılığı ile bir dosyaya erişmek için kullanılır. Sunucu, dosyanın konumuna göre doğrudan ya da dolaylı olarak konum referanslarını işler. Sunucu dosyayı yerel olarak bulabilirse istemciye aktarır. Aksi halde sunucu istemcinin URL'sini yeniden yönlendirir ve istemci dosyayı alır. Location'ı bir programcık içinde aşağıdaki gibi belirleyebilirsiniz:

Location: <http://www.tvpress.com/>

### 1.14.3 CGI Başlıklarında kullanılan Status'lar

Status (durum) alanı istemciyi izlemek için sunucuya bir durum satırı aktarmak amacıyla kullanılır. Durum kodları genellikle durumun ne olduğunu açıklayan bir katarla onu takip eden üç rakamlı bir kod olarak ifade edilir. Bu Status (durum) kodunun ilk rakamına karşılık gelen durumlar aşağıda listelenmiştir:

1xx Henüz yerleştirilmedi

2xx Başarılı

3xx Yeniden yönlendirme

4xx İstemci hatası

5xx Sunucu hatası

Sunucu tarafından birçok durum kodu kullanılmasına rağmen CGI programcığınız aracılığı ile istemciye aktardığınız durum kodları genelde istemci hata kodlarıdır. Örneğin, programcığınız bir dosyayı bulamazsa bir hata kodu döndürür. Kullanmayı isteyebileceğiniz istemci hata kodları aşağıda listelenmiştir:

Status:401 Unauthorized Authentication has failed. Kullanıcının dosyaya erişimine izin verilmedi ve tekrar denenmeli.

Status:403 Forbidden. İstek kabul edilemez. Kullanıcının dosyaya erişimine izin verilmedi.

Status:404 Not found. Belirtilen kaynak bulunamadı.

Status:405 Method not allowed. Başvuru yöntemine izin verilmedi.

#### 1.14.4 CGI Başlıklarını Temizleme

Bir CGI programcısından bir çıktı yaratmak görüldüğünden daha kolaydır. Tüm yapacağınız çıkışı, en iyi bildiğiniz programlama dilini kullanarak bir başlık ve mesaj gövdesi biçimine getirmektir. Bu bölümde iki tane örnek verilmiştir. İlk örnek Perl programlama dilinde ve ikincisi de UNIX Bourne Shell'de yazılmıştır.

Perl kullanarak basit bir HTML dökümanı çıktısı sağlayan programcık aşağıda verilmiştir:

```
#!/usr/bin/perl
```

```
#Bos bir satırla birlikte bir baslık yaratır.
```

```
print "Content-Type: text/html \n\n";
```

```
#Mesaj gövdesine HTML formatında ekleme yapar.
```

```
print <<"MAIN":
```

```
<HTML><HEAD><TITLE>Programcığın çıktısı</TITLE></HEAD> <BODY>
```

```
<H1>CGİ Kullanmanın 10 Ana Nedeni</H1>
```

```
<P>10 Müşteri geri beslemesi</P>
```

```
<P>9. Anket formları ve araştırma cevapları sağlar</P>
```

```
<P>8 Ziyaretci sayısını izleme</P>
```

```
<P>7 Otomatik arama</P>
```

```
<P>6 Kolayca veritabanı arayüzleri yaratma</P>
```

```
<P>5 Diğer protokollere geçitler oluşturma</P> <P>4 HTML 2.0 görüntü haritaları</P>
```

```
<P>3 Kullanıcı yetkilendirme</P>
```

```
<P>2 Online komut işleme</P>
```

<P>1 İstediginiz biçimde dokuman uretme

</P> </BODY>

3

---

<sup>3</sup> STANEK,William Robert,HTML JAVA CGI VRML SGML - UNLEASHED,s. 353-369,1997.