



Ders-3 Yazılım ve Programlama Dilleri / Veri ve Depolama

Yazılım:

Bilgisayarlar insan dilinden anlamaz sadece 1 ve 0 lardan yani bit ve byte lardan anlarlar. Yazılım bu boşluğu doldurur. Yazılım bilgisayarın sırayla uygulaması gereken bir dizi koddur.

Yazılım bilgisayarın spesifik görevleri yerine getirmesi için oluşturuluan komut setidir. Bilgisayarın fiziksel parçalarını kasteden donanımdan farklı olarak uygulamaların ve programların çalıştırılması için gerekli olan kavramdır.

Yazılım bir bilgisayarın değişken parçası ve donanım değişmez parçası olarak düşünülebilir.

Makina Dili:

Bilgisayarların anladığı 1 ve 0 lardan oluşan dilin adı makina dilidir. Farklı türde bilgisayarların farklı dilleri vardır Çünkü bu dilin insanlar tarafında okunup anlaşılması zordur.

İnsanlar için anlaşılması ve kullanılması zor olduğundan, başka dil türleri de geliştirilmiştir. Bunlar Assembly Dili (düşük seviye) ve üst seviye programlama dilleridir.

Assembly Language:

Assembly Dili, makine dili komutlarının her birini temsil etmek için mnemonic olarak bilinen kısa açıklayıcı sözcükler kullanır. Assembly Dili bilgisayarlar için kullanışlı değildir. Bu nedenle makine diline çevrilmesi gerekir. Çevirmene "assembler" denir.

Assembly Dili, makine dilinden çok farklı değildir. Ve komutları bilmek yeterli değildir. CPU ve belleğin nasıl çalıştığını bilmelisiniz. Bu yüzden insanlar için de pek kullanışlı değil. Neyse ki, bu tür diller artık kullanılmamaktadır. Bu nedenle, bu düşük seviyeli dilin yerine yüksek seviyeli diller geliştirilmiştir.

High-Level Languages:

1950'lerde yüksek seviyeli diller ortaya çıktı. Bu diller insan seviyesindeki dillere benzerdir. Düşük seviye diller ve makine dilinden çok daha kolay anlaşılır.

Üst düzey programlama dillerindeki komutlara statement denir. Her birinin iyi ve kötü yanları olan birçok programlama dili vardır.

Üst düzey bir dildeki bir program, source program veya source kod olarak adlandırılır. Assembly dilinde olduğu gibi, kaynak kodu da çalıştırılmadan önce makine koduna çevrilmelidir. Üst düzey diller bu çeviriyi yapmak için compiler veya interpreter adı verilen araçları kullanmalıdır. Bir interpreter veya bir compiler, kaynak koddan bir ifadeyi okur ve bunu makine koduna dönüştürür. Sonra komut hemen gerçekleşir.

Libraries(Kütüphaneler):

Sizdenönceki kullanıcıların bazı işlemleri otomatikleştirmek ve hızlandırmak için program kitaplığına manuel olarak ekenebilir. Burada amaç yazılımın karmaşıklığını ortadan kaldırmaktır. Bir matematik programı yazarken bazı şeyleri sıfırdan yazmak yerine sizden önce böyle bir uygulama geliştirmiş olan diğer geliştiriciler kolaylık olamsı için yazdıkları bazı kodları ve komutları yazılım kütüphanesine kaydederler.

Paketler ve Paket Yöneticileri:

Yazılım Paketi, örneğin masaüstüne kurulum gibi belirli bir işlevi yerine getiren bir yazılım setidir. Paketler, sisteminizdeki yazılımları ve kitaplıkları otomatik olarak yönetir. Paketleri kaynaktan (web sitesinden / DVD'den veya USB'den) veya paket yöneticilerinden kurabilirsiniz. Paket Yöneticileri, işletim sisteminizdeki Yazılımları ve Kitaplıkları (birlikte Paketler olarak adlandırılır) yönetmek için kullanılan araçtır. Paketleri kolayca kurar, yükseltir ve kaldırır. Birkaç tür paket yöneticisi vardır. En popüler olanlar Linux Sistem paket yöneticileri ve programlama dili paket yöneticileridir.

Frameworks:

Frameworks, bir görevi anlamlı ve nispeten basit bir şekilde gerçekleştirmek için araçlar ve tasarım kalıpları sağlayarak bir geliştiricinin hayatını kolaylaştırmayı amaçlamaktadır. Frameworks, yazılım kitaplıklarının babası gibidir. Frameworks, bir görevi daha kolay tamamlamak için tasarlanmış kitaplıklar, sınıflar, işlevler ve sabitler koleksiyonlarıdır. Size farklı görev kodları koleksiyonu sunarlar ve tekerleği yeniden icat etmenizi engellerler.

Çerçeveler, bir uygulamaya belirli bir görünüm dikte eder ve bir geliştiricinin kodu okumayı ve yazmayı kolaylaştırmak için yapabileceği tasarım seçimlerini sınırlar.

Model-View-Controller (MVC):

Model-View-Controller (MVC) , bir uygulamayı üç ana mantıksal bileşene ayıran mimari bir modeldir: model, görünüm ve kontrolör. Bu bileşenlerin her biri, bir uygulamanın belirli geliştirme yönlerini ele almak için oluşturulmuştur. MVC, ölçeklenebilir ve genişletilebilir projeler oluşturmak için en sık kullanılan endüstri standardı web geliştirme çerçevelerinden biridir.

Model , veriler bir şekilde temsil edilir, genellikle bir veritabanından.

Görünüm , kullanıcının gördüğü. Örneğin baktığınız web sayfası.

Denetleyici , veritabanındaki verileri işleyen kod. Örneğin, web sitenize kullanıcı bilgilerini ekleyin / silin / güncelleyin.

Frontend vs Backend:

Frontend ve Backend geliştirme terimleri, **web ve uygulama geliştirmeye ilgilidir** . Web sitenizi oluşturmadan önce, web sitesinin yapması gereken işlevleri geliştirmeli ve ikinci olarak onu çok iyi görselleştirmelisiniz.

Frontend ve Backend, web geliştirmede kullanılan en popüler iki terimdir. Bu terimler web geliştirme için çok önemlidir ancak birbirinden oldukça farklıdır. Her iki tarafın, web sitesinin işlevselliğini iyileştirmek için tek bir birim olarak diğeriyle etkili bir şekilde iletişim kurması ve çalışması gerekir.

Backend:

Backend , öncelikle sitenin nasıl çalıştığına odaklandığınız geliştirmenin sunucu tarafını (Sunucular bölümünde daha sonra açıklayacağız) ifade eder. Sitenin işlevselliğini izlemenin yanı sıra güncelleme ve değişiklik yapmak birincil sorumluluklarınız olacaktır.

Bu tür web geliştirme genellikle üç bölümden oluşur: bir **sunucu** , bir **uygulama** ve bir **veritabanı** . Backend geliştiricileri tarafından yazılan kod, veritabanı bilgilerini web tarayıcısına ileten şeydir. Web tarayıcısı, Google chrome, Opera, Safari vb. Gibi web sayfalarının "pencere uygulaması" dır. Sadece web sitesini yazdırır. Veritabanları ve sunucular gibi gözle kolayca göremeyeceğiniz her şey , bir arka uç geliştiricisinin işidir. Arka uç geliştirici pozisyonlarına genellikle programcılar veya web geliştiricileri denir.

Frontend:

Frontend geliştirme, kullanıcıların web sayfalarında veya uygulamalarında görsel olarak ilk gördükleri her şeyi yönetir. Frontend geliştiricileri, bir sitenin görünümünden ve hissinden sorumludur. Frontend geliştiricileri öncelikle üç dil kullanır: HTML, CSS, Javascript.

Frontend geliştirici iş unvanlarına birkaç örnek:

- **Web tasarımcısı**, photoshop, fireworks veya wordpress gibi farklı araçlarla web siteleri tasarlar.
- **Kullanıcı Arayüzü (UI) Tasarımcısı**temelde görsel bir tasarımcıdır ve genellikle tasarıma odaklanır.
- **Kullanıcı Deneyimi (UX) tasarımcıları**, insanların siteleri nasıl kullandığını inceleyerek ve araştırarak ön uçta çalışır. Daha sonra birçok testle değişiklik yaparlar.

Bit ve Byte

Bilgisayar 0 ve 1 lerle çalışır. Her bit 0 veya 1 den sadece bir tanesini depolayabilir.

8 bit bir araya gelerek 1 byte oluşturur her byte bir karakter taşır ör:'A' 't' '>'

Her bit 2 seçenek olarak depolanabilir. Yani bir bit 1 veya 0 olacaksa 2 seçenek ortaya çıkar. Matematiksel olarak n tane bit 2^n farklı durum oluşturur.

ASCII Kod:

ASCII (American Standard Code for Information Interchange)

her yazılan karakteri bir sayı ile temsil eden bir kodlamadır ve her numara bir baytta saklanır.

Yazım ve Bytlar:

Her harf bir baytta saklanır. 1000 harf 1000 bayt yer kaplar.

Unicode:

Unicode, evrensel bir karakter kodlama standardıdır. Metin dosyalarında, web sayfalarında ve diğer belge türlerinde tek tek karakterlerin temsil edilme şeklini tanımlar. ASCII'den farklıdır, Unicode dünyadaki tüm dillerden karakterleri, matematiksel karakter emojilerini vb. Desteklemek için tasarlanmıştır. Standart ASCII karakter seti yalnızca 128 karakteri desteklerken, Unicode yaklaşık 1.000.000 karakteri destekleyebilir. ASCII her karakteri temsil etmek için yalnızca bir byte kullanırken, Unicode her karakter için 4 byte'a kadar destekler.

Bilgisayarlarda Sayılar:

Bilgisayarda her karakter 1 byte lık yer tutar lakin bilgisayarlar sayılarla hesaplama yapmakta da iyilerdir. Sayı barındıran karakterlere integer(int) diyoruz. Integerlar genelde 4 veya 8 bytelık yer kaplarlar

4 bayt, -2147483648 ile 2147483647 arasındaki sayıları depolayabilir; 8 bayt, -9223372036854775808 ve 9223372036854775807 arasındaki sayıları depolayabilir

Kilo/Mega/Gigabytes:

Kilobyte/Megabyte:

Kilobayt, yaklaşık bin bayta (1024 bayt) eşittir. Sıradan bir Latin alfabetik metninin saklanması yaklaşık 2 kilobayt alır (harf başına yaklaşık bir bayt). Tipik bir kısa e-posta da sadece 1 veya 2 kilobayt yer kaplar. İnternette mesaj gönderirken metin mesajları ses veya görüntüden daha hızlı gönderilir. Dolayısıyla metin kompakttır ve görüntülere, sese veya videoya kıyasla birkaç bayt gerektirir.

Megabayt yaklaşık bir milyon bayta (yaklaşık 1000 KB) eşittir. Yüksek kaliteli bir dijital resim yaklaşık 2-5 megabayttır ve mp3 sesinin dakikası yaklaşık 1 megabayttır.

Gigabyte/Terabyte:

Gigabyte yaklaşık bir milyar bayttır (yaklaşık 1000 MB). Gigabyte, donanım bileşenleri için ortak bir ölçümdür. Örneğin, bilgisayar arıyorsanız 4 GB RAM ve 256 GB kalıcı depolama sıradan bilgisayar özellikleridir.

Terabayt yaklaşık 1000 gigabayt veya kabaca 1 trilyon bayttır. Bugün 4 TB sabit disk satın alabilirsiniz, bu nedenle bu terimin yaygın kullanıma girdiği zamanlardayız.

Çok duyulan bir terim var. Gigahertz. Gigahertz baytlarla ilgili değildir. Hızla ilgili. Bir gigahertz saniyede 1 milyar darbedir (megahertz, saniyede bir milyon darbedir). Gigahertz bir hız ölçüsüdür, kabaca bir CPU'nun saniyede en basit işlemini yapabileceği orandır. Gigahertz size bir CPU'nun işini ne kadar çabuk tamamladığını tam olarak söylemez, ancak kabaca ilişkilidir.