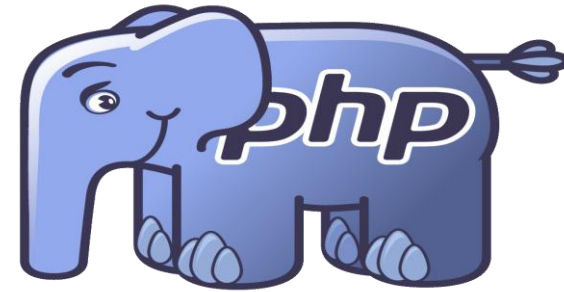


MBP192 INTERNET PROGRAMLAMA

Bahar 25-26

İEÜ MYO - Bilgisayar Programcılığı



M. Volkan GÜNGÖR



Elektronik ve Haberleşme
Mühendisi



Yazılım Proje Yöneticisi ve
Backend Developer



Yönetim Bilişim Sistemleri
Doktora öğrencisi



Yarıyıl Aktiviteleri

Ödev

7 adet

%25

Veriliş tarihinden itibaren 1 hafta sonra teslim

Github Account içinde bulunacak. - Kişisel Portfolyo

Proje

1 adet

%5

İlerleyen haftalarda Vize haftasına kadar konu belirlenmeli - karma uygulama sayfası - serbest seçim

Vize

Karma soru tipi

%25

Final

%45

Dönem içi ilerleyiş ile belirlenecek

LimitSizsiniz 😊



Final Proje Konuları önerileri

Proje

- Bilişim sistemi tasarımı ve uygulanması.
- Aklınıza gelen, gözlemlediğiniz veya çevrenize destek olacak herhangi bir sorunu ele alabilirsiniz.
- Rapor ve Github repo bulunacak.

Final

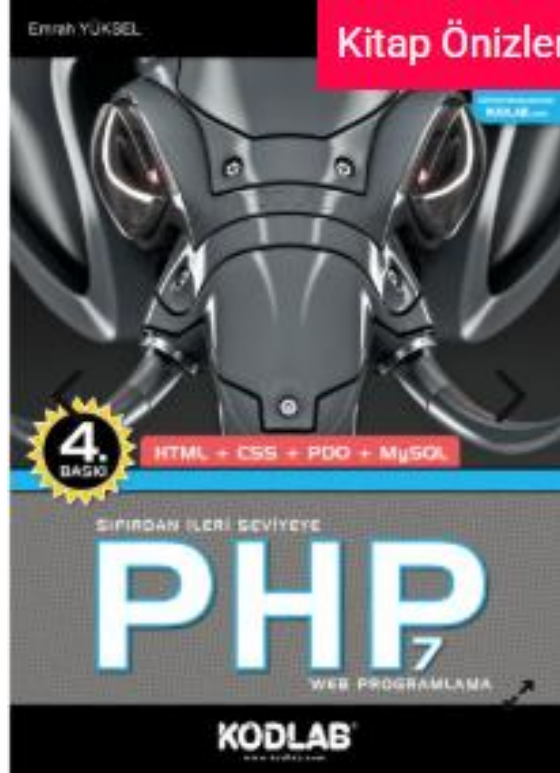
Dönem içi ilerleyiş ile belirlenecek
Yazılı veya Proje Teslimi.

Amaç, ***github repo olması ve iş başvurusunda bir sorunu anlayıp analiz edip çözümü dijital olarak bulabildiğimizi göstermek.***

Proje teslimi olursa en az 5 sayfadan oluşacak.

- Bilişim Sistemi – Talaşlı imalat sipariş takip sistemi
- Oyunlaştırma,
- Eğlenceli sayfa animasyonu.
- Yönetim Paneli
- LimitSizsiniz 😊

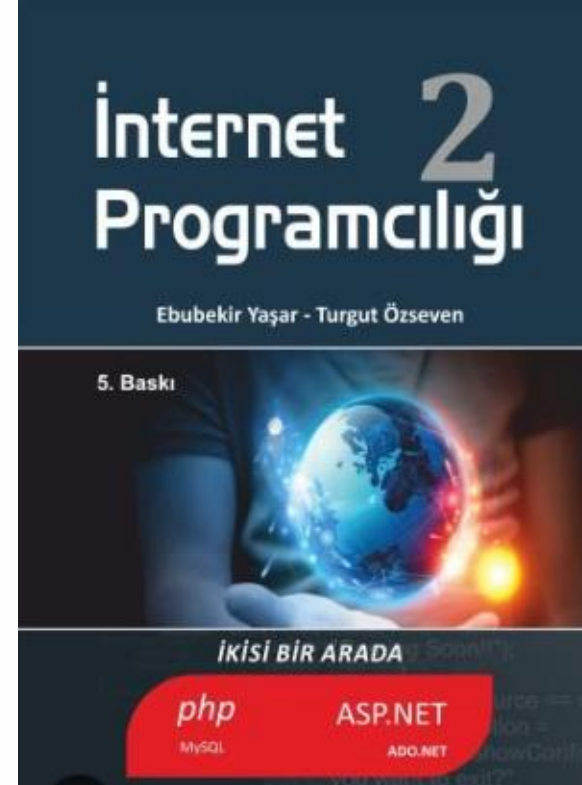
Ders Kaynakları



Emrah Yüksel

**Sıfırdan ileri
Seviyeye
PHP WEB
PROGRAMLAMA**

Kodlab Yayınları



Ebubekir Yaşar,
Turgut Özseven

**İnternet
Programcılığı 2**

**EKİN KİTABEVİ
YAYINLARI**



<https://www.php.net/manual/tr/>

www.google.com

www.w3school.com

www.youtube.com

<https://web.dev/?hl=tr>

Hafta	Konular	Ön Hazırlık
1	İnternet Kavramları	Yaşar, E., Özseven, T., "İnternet Programcılığı 2", Ekin Yayınevi (2016). Bölüm1-1
2	Php Program Yapısı ve Tanımlamalar	Yüksel, E., "Sıfırdan İleri Seviyeye PHP Web Programlama", KODLAB (2023). Bölüm 5
3	Değişkenler, Veri Tipleri	Yaşar, E., Özseven, T., "İnternet Programcılığı 2", Ekin Yayınevi (2016). Bölüm1-2 Yüksel, E., "Sıfırdan İleri Seviyeye PHP Web Programlama", KODLAB (2023). Bölüm 6
4	Operatörler	Yaşar, E., Özseven, T., "İnternet Programcılığı 2", Ekin Yayınevi (2016). Bölüm1-3 Yüksel, E., "Sıfırdan İleri Seviyeye PHP Web Programlama", KODLAB (2023). Bölüm 6
5	Diziler ve Dizi Fonksiyonları	Yüksel, E., "Sıfırdan İleri Seviyeye PHP Web Programlama", KODLAB (2023). Bölüm 7
6	Diziler ve Dizi Fonksiyonları, For döngüsü	Yüksel, E., "Sıfırdan İleri Seviyeye PHP Web Programlama", KODLAB (2023). Bölüm 8
7	Karar Yapıları	Yüksel, E., "Sıfırdan İleri Seviyeye PHP Web Programlama", KODLAB (2023). Bölüm 8
8	Ara Sınav	
9	Döngüler	Yüksel, E., "Sıfırdan İleri Seviyeye PHP Web Programlama", KODLAB (2023). Bölüm 8
10	Fonksiyonlar	Yüksel, E., "Sıfırdan İleri Seviyeye PHP Web Programlama", KODLAB (2023). Bölüm 10
11	Fonksiyonlar	Yüksel, E., "Sıfırdan İleri Seviyeye PHP Web Programlama", KODLAB (2023). Bölüm 10
12	Dosya Sistemi	Yüksel, E., "Sıfırdan İleri Seviyeye PHP Web Programlama", KODLAB (2023). Bölüm 14
13	Veritabanı İşlemleri	Yüksel, E., "Sıfırdan İleri Seviyeye PHP Web Programlama", KODLAB (2023). Bölüm 15
14	Veritabanı İşlemleri 2	Yüksel, E., "Sıfırdan İleri Seviyeye PHP Web Programlama", KODLAB (2023). Bölüm 15
15	Dönemin gözden geçirilmesi	
16	Final Sınavı	

DEVAMSIZLIK: toplam **14** saat

Teorik dersi %70 devam zorunlu

Uygulama dersi %80 devam zorunlu

Toplam vize/final hariç 56 saat (28 + 28)

Devamsızlık kayıtları

haftalık olarak sisteme işleniyor.

Ders Sözleşmemiz



Anlamamak serbesttir

Herkesin anlamama ve anlamak için soru sorma hakkı vardır.

Burada “anlamadım” demek tamamen normaldir.



Soru sormak öğrenmektir

Ders sırasında konu ile ilgili her türlü soru serbesttir.

Özel durumlar dışında - erteleme talebi değerlendirilmez.



Anlatabiliyorsan senindir

Yazdığın kodu ve yaptığın çalışmayı anlatabiliyorsan sorulara cevap verebiliyorsan o çalışma senindir.

Değerlendirme de buna göre yapılır.



Kıyas yok, ilerleme var

Başkasının ödevi veya çalışması senin için ölçüt değildir.

Herkes kendi sürecinden sorumludur.



Hata = öğrenme

Hata yapmak öğrenmenin doğal parçasıdır.

Hatalarla dalga geçilmez, birlikte öğrenme fırsatına çevrilir.



Söz kesmek yok

Fikirler özgür, iletişim saygılıdır.



Yarış değil, ortak yol

Burası yarış alanı değil, ortak öğrenme alanıdır.

Bilgi paylaşılır; üstünlük kurmak değil birlikte ilerlemek hedeflenir.



Zamana saygı

Zaman herkes için değerlidir.

Ders saatlerine, teslim tarihlerine ve ortak kurallara özen göstermek hepimizin sorumluluğudur



Herkes öğrenir

Eğitmen de öğrenme sürecinin içindedir.

Bilinmeyen bir konu varsa açıkça söylenir, birlikte araştırılır.



HABERLEŞME

- mvolkang@gmail.com
- OBS mesajları

mvolkang@gmail.com

[MBP-192/ Şube] konu.... **Konu kısmında kesinlikle [MBP-192 / şube] bilgisi bulunmalı**

.....

--

İyi günler

İsim/Soyad

Okul numarası

Web Geliřtirmede Problem Çözme

Web geliřtirme sadece kod yazmak deęildir. **Bir ihtiyaçı anlamak ve ona çalıřan bir çözüm üretmektir.** Yapılan her proje, karşılaşılan bir problemi çözmeyi amaçlar.

- Etkileřimli, Kullanıřlı
- Kullanıcıyı memnun eden
- Fikri çalıřan bir ürüne dönüřtüren
web siteleri ve uygulamalar geliřtirmektir.

Web geliřtirme = Problem + Kod + Çözüm

Bir web geliřtirici olarak,

- her projede yeni řeyler öğrenir ve farklı problemlerle karşılaşırsınız
- sadece kod yazan deęil, **çözüm üreten bir programcı** olursunuz.

Bugün YZ hepimize yeni fırsatları ile geliyor. YZ ile meslekler arasındaki farklar hem kapanıyor hem açılıyor. Bu deęiřimin yönünü belirlemek sizin elinizde.

Bu dersin ana amacı sizin yeni bir dilde daha sorun çözebilmenizi saęlamak.



Bunlar ne ola ki?

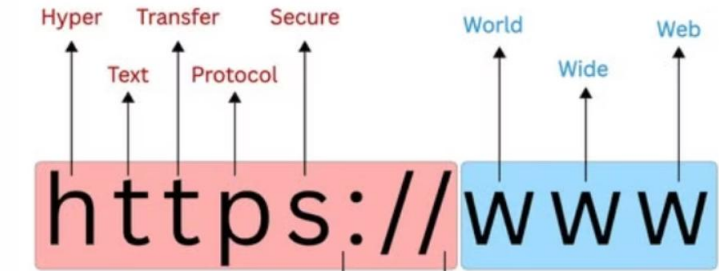


Tarayıcılar!



Internet!

```
Domain Name: ISC.COM
Updated Date: 2016-06-05T07:03:37Z
Creation Date: 1986-08-05T04:00:00Z
Admin Name: Koosis, Donald
Admin Organization: ISC Consultants Inc.
Admin Street: PO Box 1379
Admin City: Woodstock
Admin State/Province: NY
Admin Postal Code: 12498
Admin Country: US
Admin Phone: +1.2124778800
Admin Email: donald.koosis@ISC.COM
```



Web!

Temel Tanımlar



Tarayıcılar!

Tarayıcı, WWW üzerindeki içeriklere erişmemizi, bu içerikleri görüntülememizi ve onlarla etkileşime girmemizi sağlayan **bir yazılımdır**.

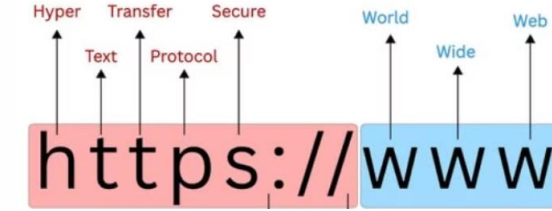
- **Çevirmen Rolü:** Sunucudan gelen karmaşık kodları (HTML, CSS, JavaScript) bizim anlayabileceğimiz görsel bir arayüze (resimler, butonlar, metinler) dönüştürür.
- **Popüler Örnekler:** Google Chrome, Safari, Microsoft Edge, Mozilla Firefox.
- **Temel İşlevi:** URL'yi IP adresine dönüştürmek için istek gönderir ve gelen yanıtı ekranımıza çizer (rendering).



İnternet!

İnternet, dünya üzerindeki bilgisayar ağlarını birbirine bağlayan **devasa bir fiziksel ağ altyapısıdır ve üstünde çalışan protokoller bütünüdür**.

- **Kapsamı:** Fiber optik kablolar, uydular, routerlar ve veri merkezlerinden oluşur.
- **Görevi:** Bilginin (verinin) bir noktadan diğerine taşınmasını sağlar.
- **Örnek:** Sadece web sitelerini değil; e-postaları, WhatsApp mesajlarını, online oyun verilerini ve dosya transferlerini (FTP) taşıyan ana "yoldur".



Web!

WWW, İnternetin altyapısı üzerinde çalışan bir servistir.

HTTP/HTTPS protokolü üzerinden erişilen dokümanlar sistemi.

Mantığı: Birbirine "hyperlink" (bağlantı) dediğimiz köprülerle bağlı olan dokümanlar (web sayfaları) bütünüdür.

Tarihçesi: Tim Berners-Lee tarafından 1989'da CERN'de geliştirilmiştir.



Web Geliştirme Nedir?

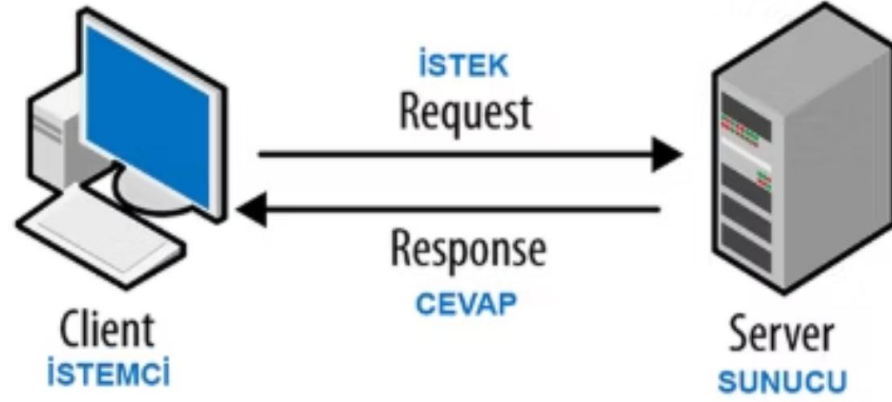
Web sitelerinin oluşturulması, geliştirilmesi ve bakımını ifade eder.

Web tasarımı, web yayıncılığı, web programlama ve veritabanı yönetimi gibi alanları kapsar.

İnternet/Intranet üzerinden çalışan uygulamaların ve web sitelerinin oluşturulması sürecidir.

Web geliştiriciler, basit statik web sitelerinden karmaşık ve dinamik web uygulamalarına kadar her şeyi oluşturmak için çeşitli programlama dilleri, araçlar ve framework'ler kullanır.

İstemci-Sunucu (Client-Server) Modeli



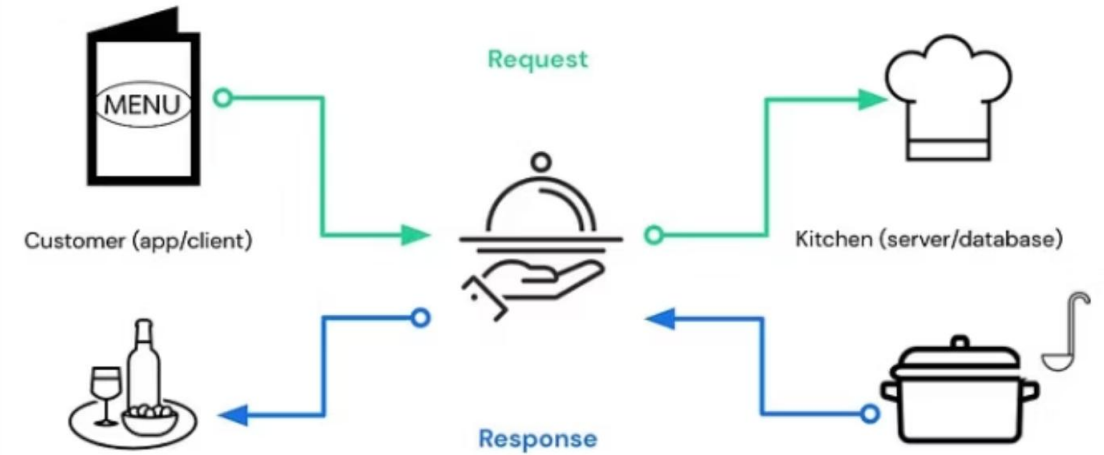
İstemci (Client)

Hizmet talep eden taraftır. Web bağlamında bu genellikle kullanıcının tarayıcısıdır (Chrome, Firefox vb.). Kullanıcı arayüzünü sunar ve kullanıcı etkileşimlerini yakalar.



Sunucu (Server)

Hizmet sunan, kaynakları (veri, dosya, veritabanı) barındıran taraftır. İstemciden gelen talepleri dinler, işler ve sonucu geri gönderir.



HTTP İstek-Yanıt (Request-Response) Döngüsü

1

İstek (The Request)

Tarayıcı, sunucuya bir paket gönderir. Bu paketin içinde şunlar yer alır:

1. Metot (Method): Ne yapmak istiyoruz?

- GET: Veri getir (Sayfayı görüntüle).
- POST: Veri gönder (Form doldur, kayıt ol).

2. URL (Path): Hangi kaynağa erişmek istiyoruz? (/anasayfa veya /api/kullanici)

3. Header (Üstbilgi): İstemci hakkında ek bilgiler (Hangi tarayıcı kullanılıyor, dil tercihi nedir?).

2

İşleme (Processing)

Sunucu isteği alır, gerekiyorsa veritabanına gider, dosyaları okur ve bir sonuç hazırlar.

3

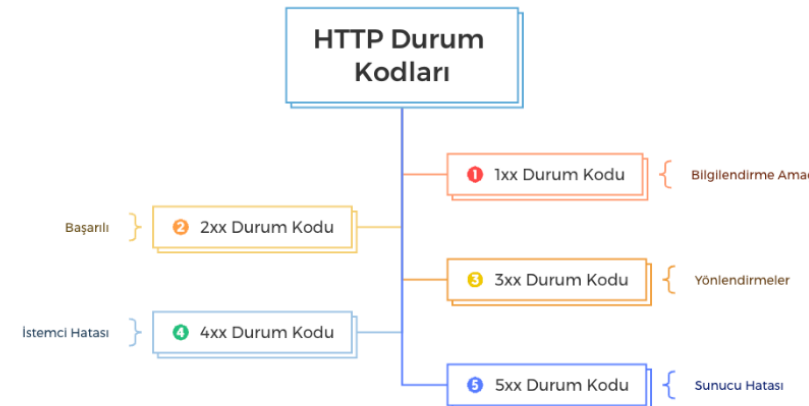
Yanıt (The Response)

Web Sunucusu hazırladığı paketi istemciye geri gönderir. Bu paketin en önemli parçaları:

1. Durum Kodu (Status Code): İşlem başarılı mı?

- 200 OK: Her şey yolunda.
- 404 Not Found: Aradığın sayfa yok.
- 500 Internal Server Error: Sunucuda işler karıştı.

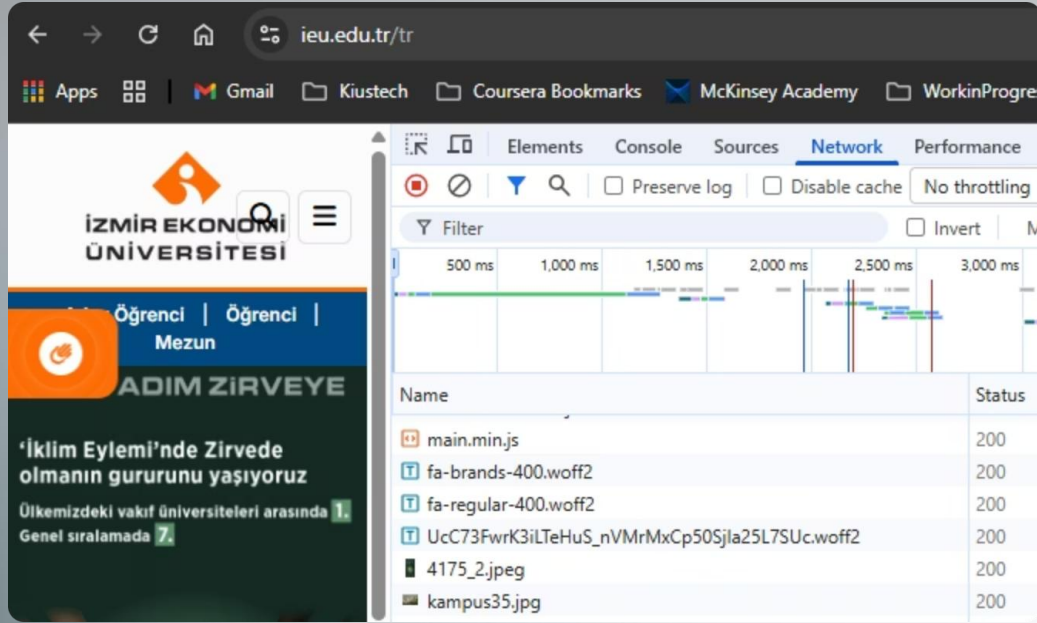
2. Body (Gövde): İstenilen asıl içerik. Genellikle HTML, JSON veya bir görsel dosyasıdır.



Adres Çubuğunun Arkasında Neler Oluyor?

Bir URL (örneğin: <https://www.ieu.edu.tr/>) yazıldığında gerçekleşen mikro saniyelik olaylar

- **DNS Çözümleme:** Tarayıcı, "www.ieu.edu.tr" isminin hangi IP adresine (örneğin: 192.0.2.1) karşılık geldiğini öğrenmek için telefon rehberine (DNS) bakar.
- **TCP/IP Bağlantısı:** Sunucu ile güvenli bir hat kurulur
- **HTTP İstekleri:** Tarayıcı "Bana ana sayfayı gönder" der.
- **HTTP Cevapları:** Sunucu "Ana sayfayı ile gönderildi" ile dosyaları gönderir.
- **Tarayıcı Render Süreci:** Sunucudan gelen HTML/CSS/JS dosyaları tarayıcı tarafından yorumlanarak görsel bir arayüze dönüştürülür.



İnternet Nasıl Çalışır?

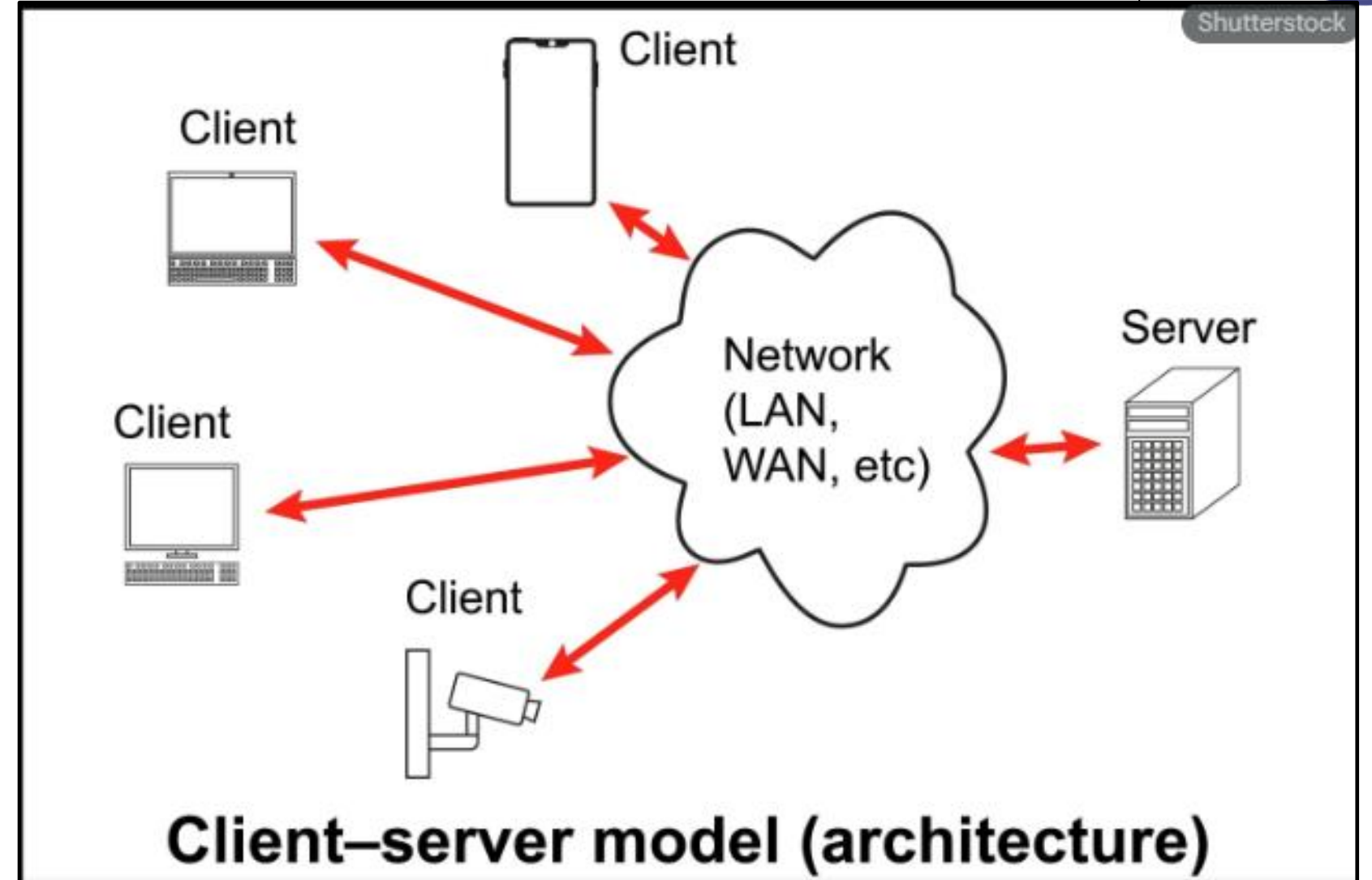
İstemci(Client): Girdiyi aldıktan sonra istemciye istek gönderen yazılımdır

Server: İstemcilerden gelen istekleri alan ve işleyen yazılımdır.

Sunucu, diğer bilgisayarların ihtiyaçlarını karşılamak için hizmetleri çalıştırmaya yönelik çalışır.

Sunucular bir dosya sunucusu, veritabanı sunucusu, web sunucusu olabilir.

Kullanılacak kuralları, dili ve diyalog modellerini resmi olarak tanımlayan ortak bir **iletişim protokollerine** (TCP ve HTTP/HTTPS) bağlı kalır.



Web Geliştirmenin Bölümleri



Front-end Geliştirme (İstemci Tarafı)

Kullanıcıların bir web sitesinde **gördüğü ve etkileşime girdiği** her şeyi kapsar.

- Görseller, yerleşim düzenleri ve kullanıcıya dönük tüm öğeler yönetilir
- Sayfanın yapısı, tasarımı ve etkileşimi için **HTML, CSS ve JavaScript** kullanılır.

UX - Kullanıcı deneyimi?



Back-end Geliştirme (Sunucu Tarafı)

Web sitelerinin arka plandaki işlemlerini yönetir, performansı sağlar ve verilerin güvenli şekilde akmasını mümkün kılar.

- Sunucu ile kullanıcı arasında doğru veriyi gönderip alır
- Verileri güvenli şekilde saklar ve sunucu işlemlerini yönetir

Web Geliřtirmenin Bölümleri



Front-end Teknolojileri

- **HTML (HyperText Markup Language):** Web sayfasının temel yapısını oluşturur. Tarayıcıya başlıklar, paragraflar ve görseller gibi öğelerin ne olduğunu söyler.
- **CSS (Cascading Style Sheets):** Sayfanın nasıl görüneceğini belirler. Renkler, yazı tipleri, düzen ve konumlandırma CSS ile yapılır.
- **JavaScript:** Web sayfasına etkileşim kazandırır. Açılır mesajlar, etkileşimli formlar, görsel kaydırıcılar ve hatta oyunlar JavaScript sayesinde çalışır. JavaScript olmadan web siteleri statik ve sıkıcı olurdu.



Back-end Teknolojileri

- **Sunucu (Server):** Web sitesinin barındığı güçlü bilgisayardır. Tüm dosyalar ve veriler burada bulunur.
- **Veritabanları:** Kullanıcı profilleri, blog yazıları veya ürün stokları gibi veriler burada saklanır.

Örnekler: **MySQL, MongoDB, PostgreSQL**

- **Programlama Dilleri:** Back-end geliştiriciler **PHP, Python, C# ve Java** gibi dilleri kullanır.

Örneğin: giriş yaptığında, back-end kodu bilgilerini kontrol eder ve eşleşiyorsa seni içeri alır.

Web Geliřtirmede Kullanılan Aralar

Kod Editörleri

- **VS Code:** Güçlü ve kullanıcı dostu
- **Sublime Text:** Hafif ve hızlı

Versiyon Kontrol

- **Git:** Kod deęiřikliklerini takip etmeyi sağlar
- **GitHub:** Kodların saklandığı ve paylaşıldığı platform

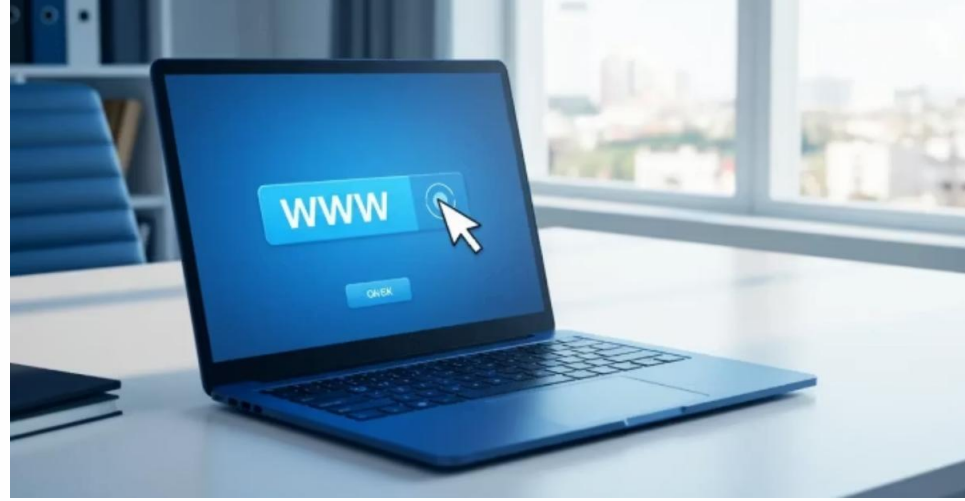
Web Tarayıcıları

Chrome, Firefox ve Safari; siteleri test etmek ve hata ayıklamak için kullanılır.

Framework ve Kütüphaneler

- **Laravel:** Modern web uygulamaları geliřtirmek için kullanılan PHP framework'ü
- **React:** Arayüz geliřtirmek için JavaScript kütüphanesi
- **Bootstrap:** Responsive tasarım için CSS framework'ü
- **Node.js:** Back-end geliřtirme için JavaScript alıřma ortamı
- **Django:** Hızlı ve güvenli web uygulamaları geliřtirmek için kullanılan Python framework'ü

Web Uygulamaları



Tarayıcıya bir web adresi yazdığında ya da bir linke tıkladığında **arka planda neler oluyor?**

Farz edelim:

Evde oturuyorsun ve canın pizza çekti. Bunun için bir uygulama ya da web sitesi açıp sipariş veriyorsun?

Web Uygulamalarında Akış

Kullanıcı Front-End ile Etkileşime Girer

Bir online mağazayı (www.onlineshop.com) ziyaret edersin ve beğendiğin bir ürünü görürsün. “Sepete Ekle” butonuna tıklarsın. Bu, senin front-end ile (gördüğün ve tıkladığın kısım) etkileşime girmendir.

Front-End Back-End'e İstek Gönderir

“Sepete Ekle” butonuna tıkladığında, front-end ürünün bilgilerini (stok durumu, fiyat vb.) kontrol etmek için back-end'e bir istek gönderir. Bu, bir garsona “Bu ürünü siparişime eklemek istiyorum” demek gibidir.

Back-End İsteği İşler

Sunucu (back-end), ürünün stokta olup olmadığını kontrol etmek için veritabanına bakar ve gerekli verileri hazırlar (ürün bilgileri, stok miktarı vb.).

Back-End Veriyi Front-End'e Gönderir

Sunucu isteği işledikten sonra, ürünün sepete eklendiği ve toplam tutarın güncellendiği bilgileri front-end'e geri gönderir.

Kullanıcı sayfayı görüntüler

Front-end sayfayı günceller ve yeni ürünün eklendiği alışveriş sepetini ve güncellenmiş toplam fiyatı ekranda gösterir. Artık yaptığın işlemin sonucunu görürsün.

Web Uygulamaları Nasıl Çalışır?

Yolculuk Başlıyor: Tarayıcı ve İstek

- **Tarayıcı (Browser):** İnternete açılan kapınız (Chrome, Safari, Firefox).
- **Rolü:** Sizin adınıza verileri gidip getiren bir "uşak" gibidir.
- **İlk Adım:** Adres çubuğuna `www.pizzayum.com` yazıldığında süreç tetiklenir.

Web Sunucusu (Web Server)

Web sitesi dosyalarını saklayan ve isteklere yanıt veren merkezi sistem.

Yazılımlar: Apache, Nginx

İşleyiş:

1. İsteği alır.
2. Gerekli dosyaları hazırlar.
3. Yanıtı (HTML) tarayıcıya gönderir.

Veritabanı (Database)

- **Rolü:** Menü fiyatları, kullanıcı yorumları ve stok bilgilerini saklar.
- **SQL vs NoSQL:**
 - **SQL:** Düzenli, ilişkisel tablolar.
 - **NoSQL (MongoDB):** Esnek ve büyük veri yapıları.
- **Süreç:** Sunucu veritabanına "Pizza fiyatlarını getir" der, veritabanı yanıtı gönderir.

1

2

3

4

5

6

Adres Bulma: DNS ve URL → IP

- **URL Ayırıştırma:** Protokol (`https`), Alan Adı (`pizzayum.com`) ve Yol (`/menu`).
- **DNS (Domain Name System):** İnsan dostu isimleri, bilgisayarların anladığı **IP adreslerine** (Örn: 192.0.2.1) çevirir.
- **Neden Gerekli?** Karmaşık sayıları ezberlemek zorunda kalmamamız için.

Protokoller ve Güvenlik

- **HTTP/HTTPS:** Tarayıcı ile sunucu arasındaki konuşma dili.
- **TCP/IP:** Verilerin parçalanmadan, doğru sırayla ulaşmasını garanti eder.
- **SSL/TLS (Handshake):** Verilerin şifrelenmesi. Kredi kartı ve şifre gibi hassas bilgileri korur.

Görselleştirme (Rendering)

- **HTML:** Sayfanın iskeleti (Başlıklar, paragraflar).
- **CSS:** Sayfanın makyajı (Renkler, düzen, yazı tipleri).
- **JavaScript:** Sayfanın ruhu (Etkileşimler, butonlar, animasyonlar).
- **DOM:** Tarayıcının bellekte oluşturduğu sayfa haritası.

Web Uygulamaları Nasıl Çalışır?

Adresi yaz
(**Tarayıcı**)

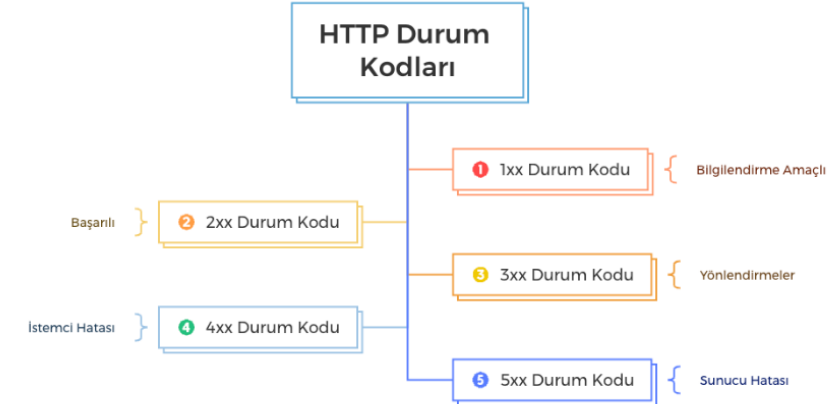
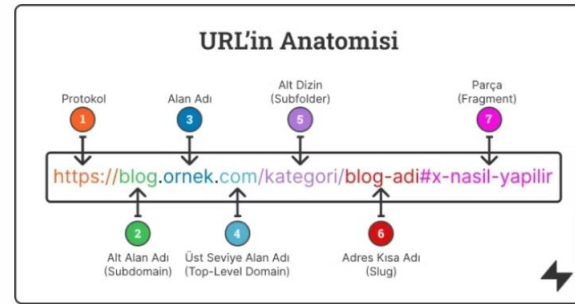
Kapıyı çal - İsteği
gönder
(**HTTP/HTTPS İsteği**)

Güvenliği sağla
(**SSL/TLS**)

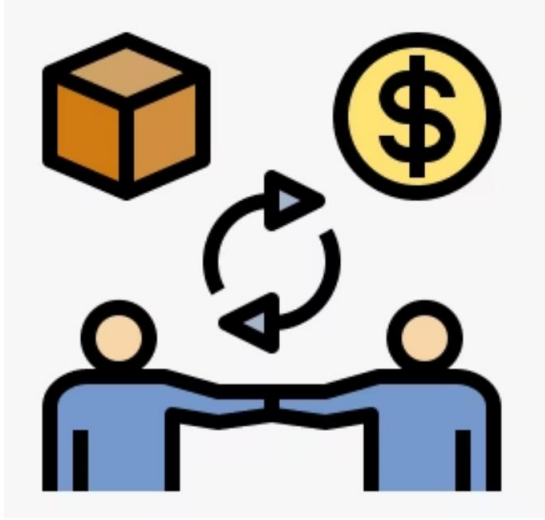
Konumu bul
(**DNS: URL → IP**)

İçeriği hazırla
(**Sunucu & Veritabanı**)

İçeriği Görüntüle
(**Rendering**)

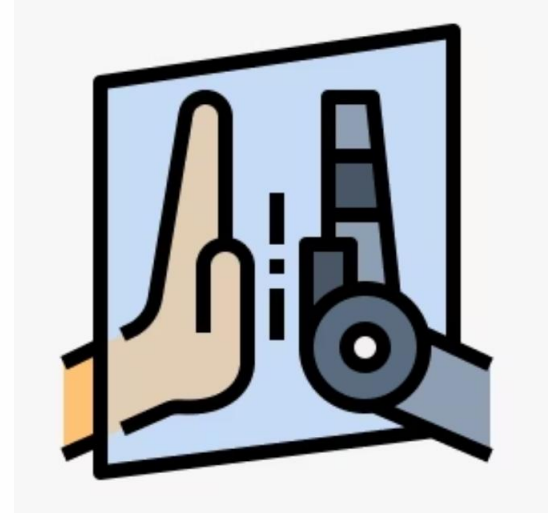


Web Geliştirme neden önemli



Artan Talep

Daha fazla işletme, süreç çevrimiçi hale geliyor



Dijital Dönüşüm

Artık dünya yeni bir modelde evrilmek zorunda.

Dijital dönüşüm; kod, veri ve otomasyonla süreçleri akıllı, hızlı ve ölçülebilir hale getirme sanatıdır.



Yaratıcılık ve Problem Çözme Süreci

Web geliştirme sadece kod yazmak değildir – kullanıcı arayüzleri tasarlamak, gerçek dünya problemlerini yaratıcı yollarla çözmek gibi işler de içerir.



Tavan da taban da sizsiniz - yapabilirsiniz
👍

Temelleri öğrendikten sonra mobil uygulama, oyun geliştirme, yapay zekâ araçları veya kendi teknolojik girişimini kurma gibi alanlara yönelebilirsin.

Web Geliştirme neden önemli -2



Kendi kendinize başlayabilirsiniz

Çevrimiçi ücretsiz veya uygun maliyetli kaynaklar ve sistemler ile öğrenmeye ve uygulamaya hemen başlayabilirsiniz.



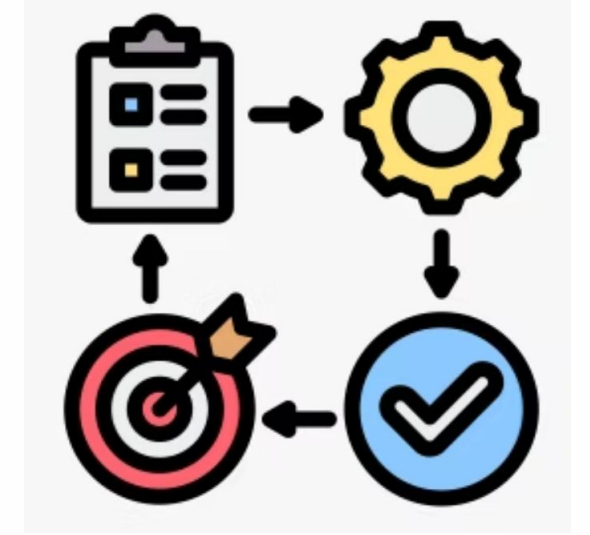
Aklınıza geleni uygulama fırsatı

Aklına gelen fikrin kaderi sensin: programcıysan hayata geçer. Programcılık, 'ya şöyle olsaydı' cümlesini 'işte oldu'ya dönüştürmektir.



Farklı sektörlerde çalışabilirsiniz

Sağlıktan oyuna, finanstan sanata: Programcılık, fikri gerçeğe çevirir.



Sürekli öğrenme

Web geliştirme sürekli evrim geçirir. Hem öğrenmeye devam edersin hem de yeteneklerin her yıl değer kazanır.

Web Geliřtirmede Problem Çözme

Web geliřtirme sadece kod yazmak deęildir. **Bir ihtiyaçı anlamak ve ona çalıřan bir çözüm üretmektir.** Yapılan her proje, karşılaşılan bir problemi çözmeyi amaçlar.

- Etkileřimli, Kullanıřlı
- Kullanıcıyı memnun eden
- Fikri çalıřan bir ürüne dönüřtüren
web siteleri ve uygulamalar geliřtirmektedir.

Web geliřtirme = Problem + Kod + Çözüm

Bir web geliřtirici olarak,

- her projede yeni řeyler öğrenir ve farklı problemlerle karşılaşırsınız
- sadece kod yazan deęil, **çözüm üreten bir programcı** olursunuz.



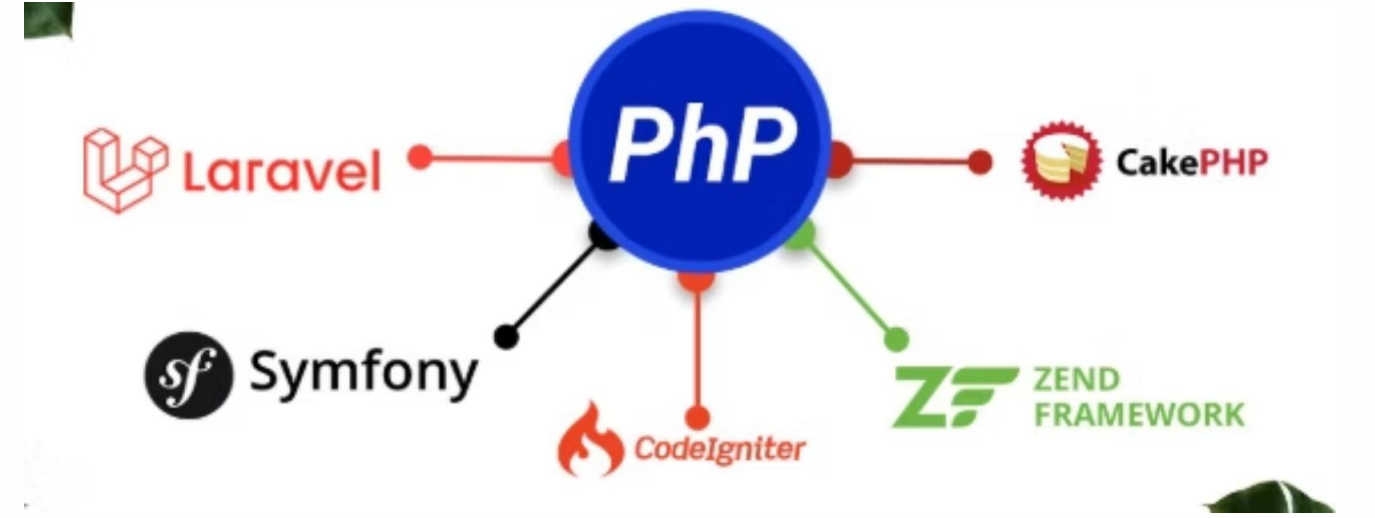
Framework nedir?

Framework = Hazır İskelet

- Web uygulaması geliřtirmek için
- **Hazır yapı + kurallar** sunar
- Tekrar eden işleri senin yerine yapar

🚀 Popüler PHP Framework'leri

- 📌 **Laravel** → En popüler, öğrenmesi kolay
- 🏢 **Symfony** → Büyük ve kurumsal projeler
- ⚡ **CodeIgniter** → Hafif ve hızlı





PHP

Web geliřtirmede yaygın olarak kullanılan, açık kaynaklı bir betik (scripting) dilidir. Sunucu tarafında çalışır ve bir web uygulamasında görüntülenen dinamik içerikleri üretir.

PHP kodu sunucuda çalıştırılır ve ortaya çıkan HTML çıktısı istemcinin (kullanıcının) tarayıcısına gönderilir.

PHP platformdan bağımsızdır; Windows, Linux ve macOS gibi farklı işletim sistemlerinde çalışabilir.

PHP, oturum (session) yönetimini destekler; bu sayede bir web sitesindeki farklı sayfalar arasında kullanıcı aktiviteleri takip edilebilir.

Dinamik tipli bir dildir; değişkenlerin türleri çalışma sırasında değişebilir, bu da kodlama sürecinde esneklik sağlar.

Neden PHP?

PHP kullanmanın en güzel yanlarından biri, yeni kullanıcılar için öğreniminin oldukça kolay olması ve aynı zamanda profesyonel kullanıcılar için ileri seviyede özellikler içermesidir.

PHP ile programlamaya başlayabilir ve birkaç saat içerisinde basit programlar yazmaya başlayabilirsiniz.

PHP büyük işletim sistemlerinde; **Linux**, birçok **Unix** türevi (HP-UX, Solaris, OpenBSD vb.), **Microsoft Windows**, **Mac OS**, çok çeşitli platformlarda çalışabilir.

Yaygın kullanılan HTTP sunucularının büyük kısmını destekler.

Apache, IIS, nginx

Neden PHP Öğreniyoruz?

Web Geliştirmede Yaygın Kullanım: PHP, web geliştirmede

- ▶ oldukça yaygındır; web sitelerinin %40'ından fazlası WordPress kullanmaktadır.

PHP ile Kod Yazmanın Kolaylığı: PHP, HTML ile uyumlu

- ▶ çalıştığı için kod yazımı oldukça basittir. Bu sayede geliştiriciler sunucu tarafı ve istemci tarafı kodlarını bir arada kullanabilir, web geliştirme süreci daha sade ve verimli hâle gelir

Yeni Başlayanlar İçin Uygun: PHP, basit sözdizimi ve kolay

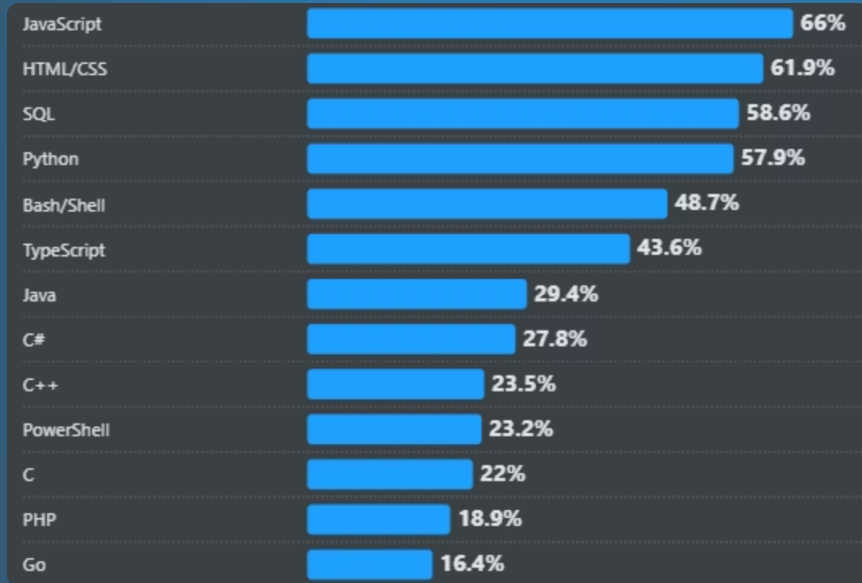
- ▶ kurulumu sayesinde başlangıç seviyesindeki geliştiriciler için idealdir

HTML ile Gömülebilme: PHP, HTML içine kolayca gömülebilir.

- ▶ Bu da dinamik içerik üretimini kolaylaştırır ve sunucu tarafı mantığının ön yüz (front-end) arayüzüyle birlikte çalışmasını sağlar.

Popüler Framework'lerle Uyum: PHP; Laravel ve Symfony

- ▶ gibi popüler framework'lerle uyumludur.



PHP ve Gerekli Uygulamaların Kurulumu

- PHP kurulumu
- Env. Variables değiştirilmesi
- METHOD 1: VS code ile PHP
 - Extensions (PHP Extension Pack)
 - LiveServer
 - Php Server
- METHOD 2: XAMPP ile PHP
 - Kurulum

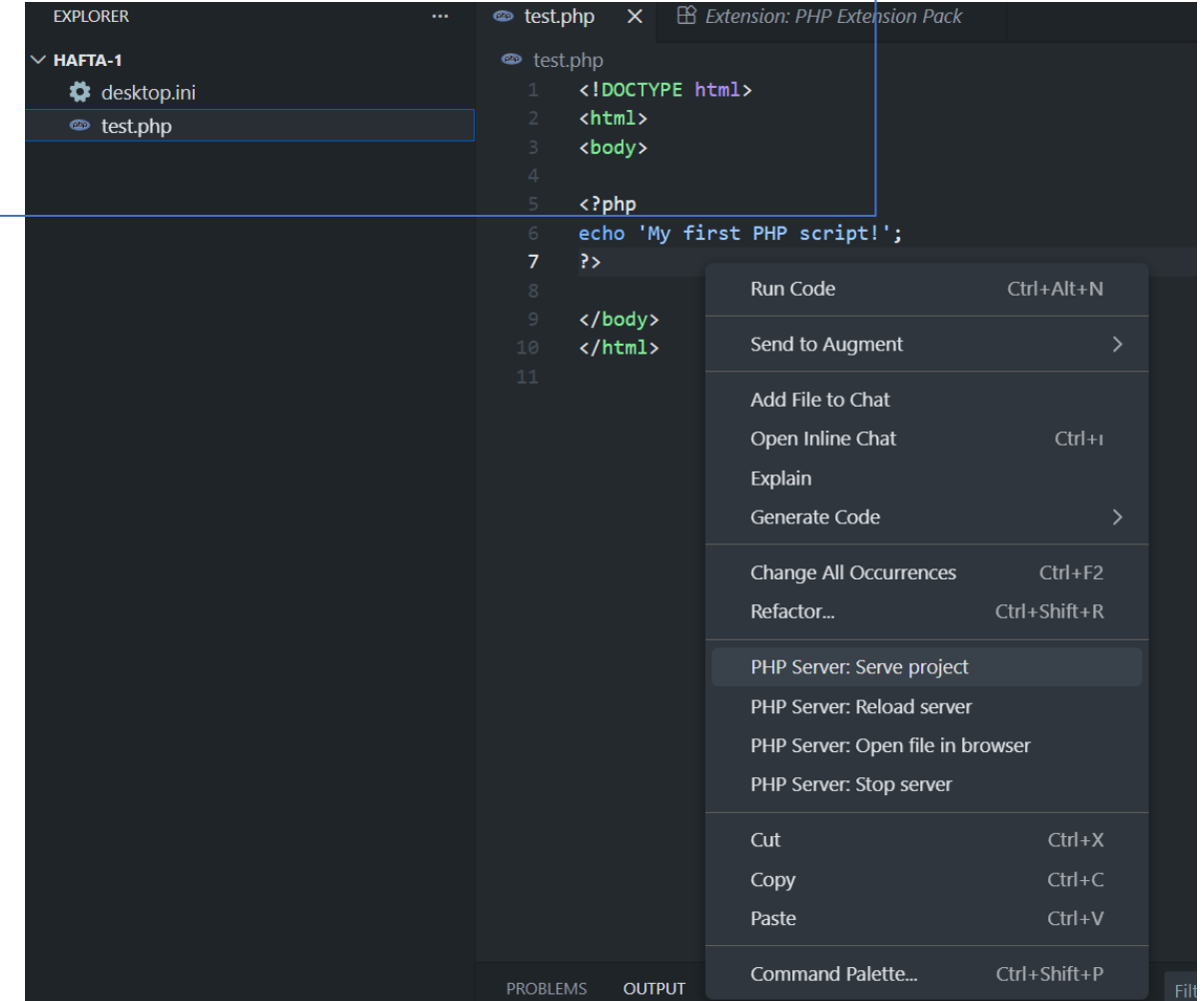
METHOD 1: Kurulum

- VS code kurulumunu gerçekleştirin.
- Sonra gerekli VS Code extensionların kurulumunu gerçekleştirin.
 - PHP Server
 - PHP Extension Pack
- Klasör yapısını oluşturun
- Klasör altına “test.php” dosyasını ekleyin.
- Dosya üstünde sağ tıklama ile menüyü açın ve “PHP server Server Project” seçerek başlatın.

```
<!DOCTYPE html>  
<html>  
<body>
```

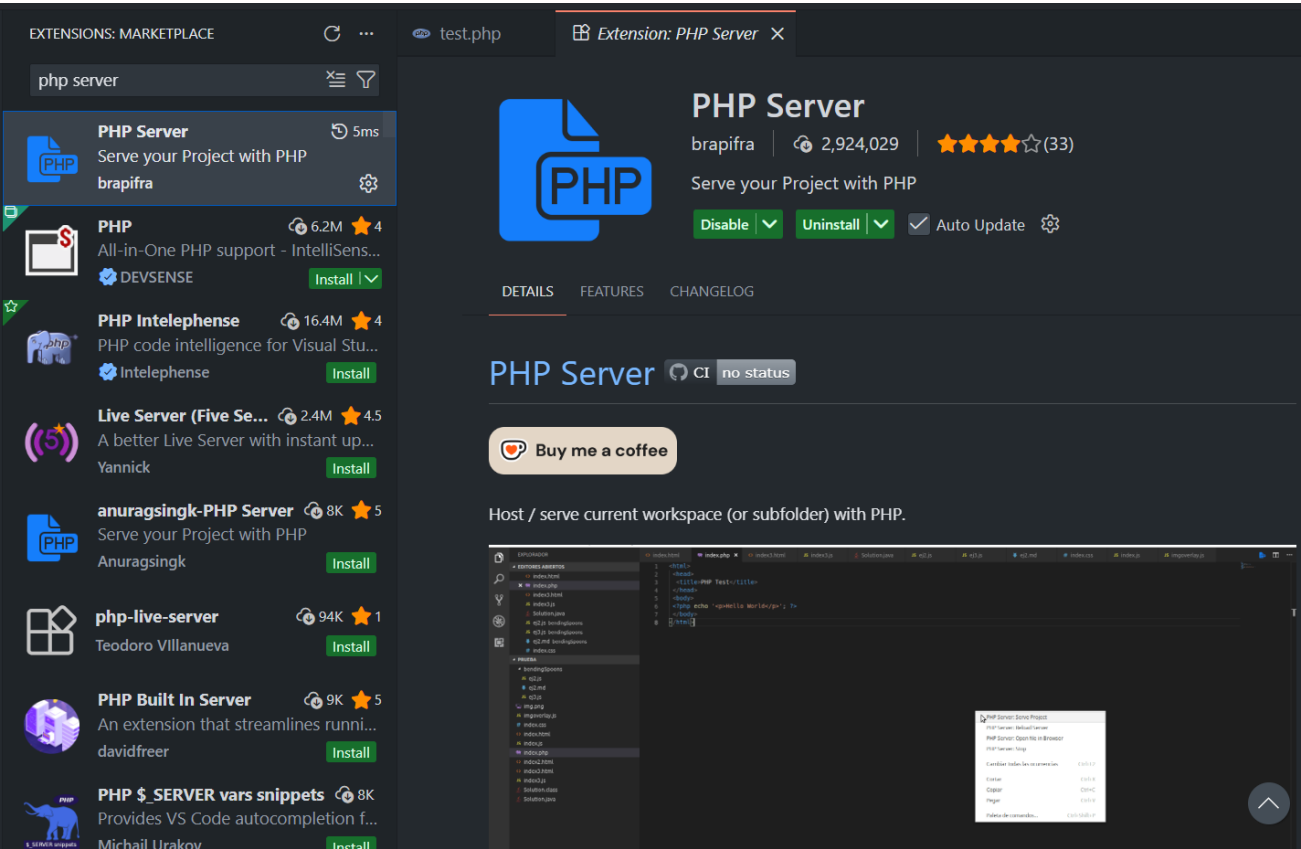
```
<?php  
echo 'My first PHP script!';  
?>
```

```
</body>  
</html>
```

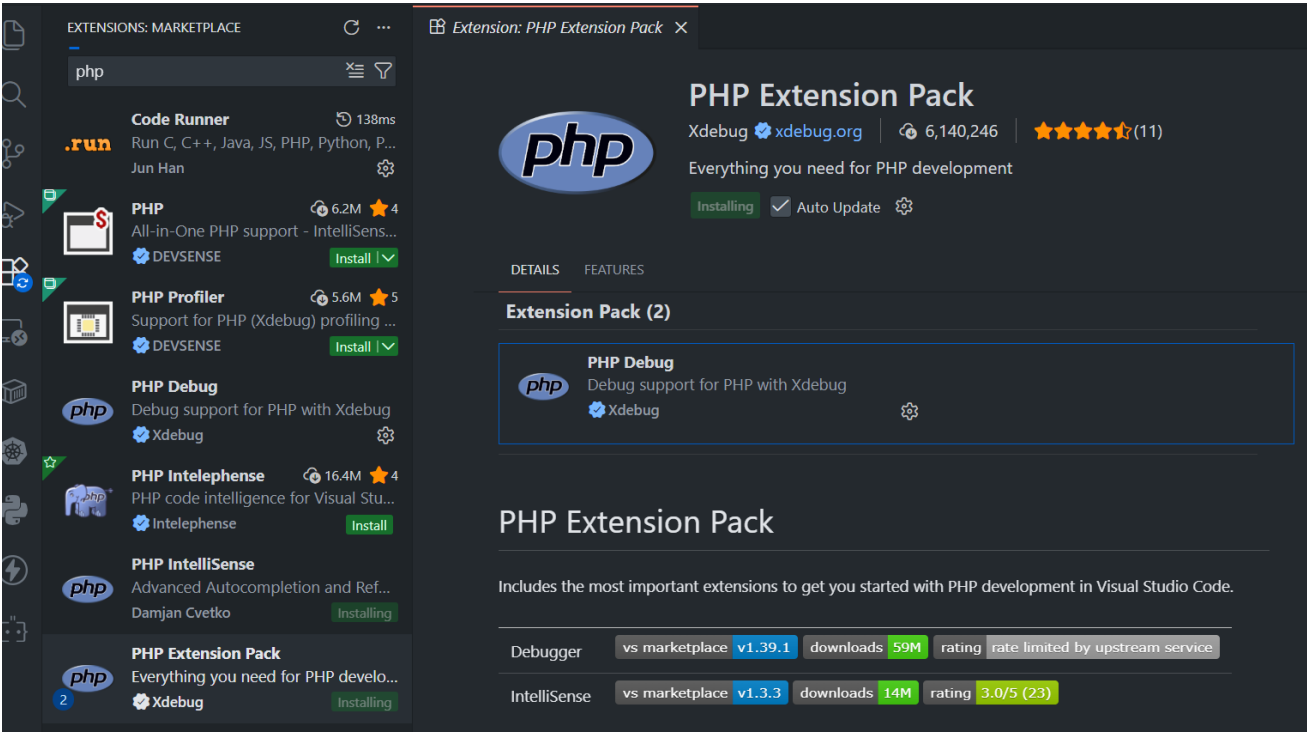


VS code ile PHP kullanımı

PHP Server

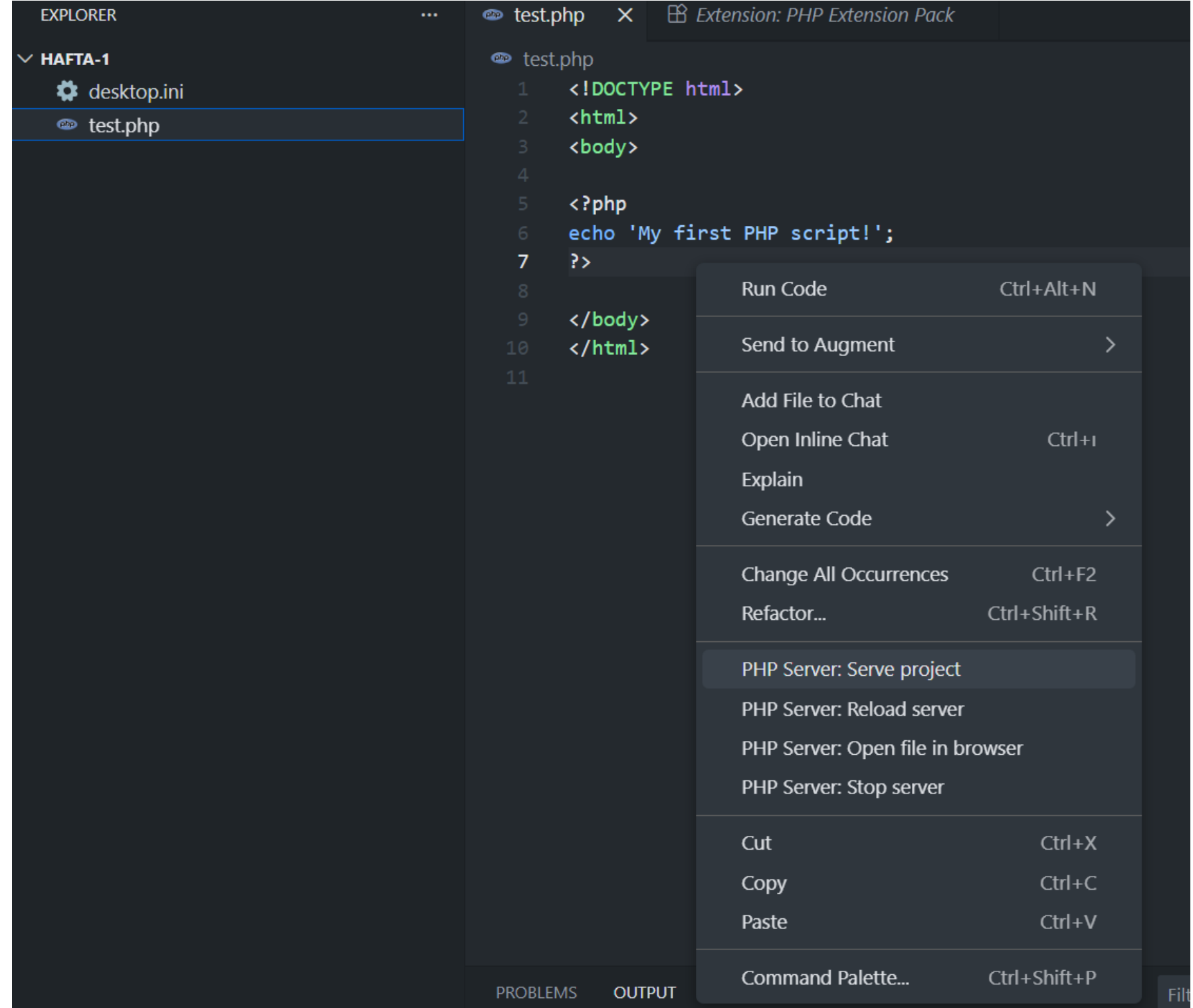


PHP Extension Pack



PHP dosyasını çalıştırmak

- Dosya üstünde sağ tıklama ile menüyü açın
- “PHP server Server Project” seçerek başlatın.
- Seçili browser üzerinde dosya çalışmaya başlayacaktır.
- Projenin çalışmasını durdurmak için “PHP Server: Stop Server” seçimi ile proje durdurulur.



METHOD 2 – XAMPP Kurulum

- Kullandığınız işletim sistemine göre indirme seçeneklerinden birini indirin.
 - Windows (derste kullanım)
 - Linux
 - Mac
- Windows için: “xampp-windows-x64-8.2.12-0-VS16-installer” kurulum dosyasını başlatın.
- Sisteminizde antivirus program ve firewall bulunuyorsa kurulum boyunca kapalı(disabled) kalması kurulumu kolaylaştıracaktır.
- Kurulum Sonrası antivirus uygulamaları ve diğer kapattığınız uygulamaları tekrar açın.

 **XAMPP** Apache + MariaDB + PHP + Perl

What is XAMPP?

XAMPP is the most popular PHP development environment

XAMPP is a completely free, easy to install Apache distribution containing MariaDB, PHP, and Perl. The XAMPP open source package has been set up to be incredibly easy to install and to use.


XAMPP

[Download](#)
Click here for other versions

 **XAMPP for Windows**
8.2.12 (PHP 8.2.12)

 **XAMPP for Linux**
8.2.12 (PHP 8.2.12)

 **XAMPP for OS X**
8.2.4 (PHP 8.2.4)



**New XAMPP
release 8.2.12,
8.1.25 and 8.0.30**

Hi Apache Friends!
We just released a new version of XAMPP for Windows for PHP versions 8.2.12, 8.1.25 and 8.0.30. New versions for Linux and OS X will come soon! You can download these new installers...

[Read more »](#)

About Apache Friends

Apache Friends is a non-profit project to promote the Apache web server and is home to the XAMPP project. XAMPP is an

Community

XAMPP has been around for more than 10 years – there is a huge community behind it. You can get involved by joining our

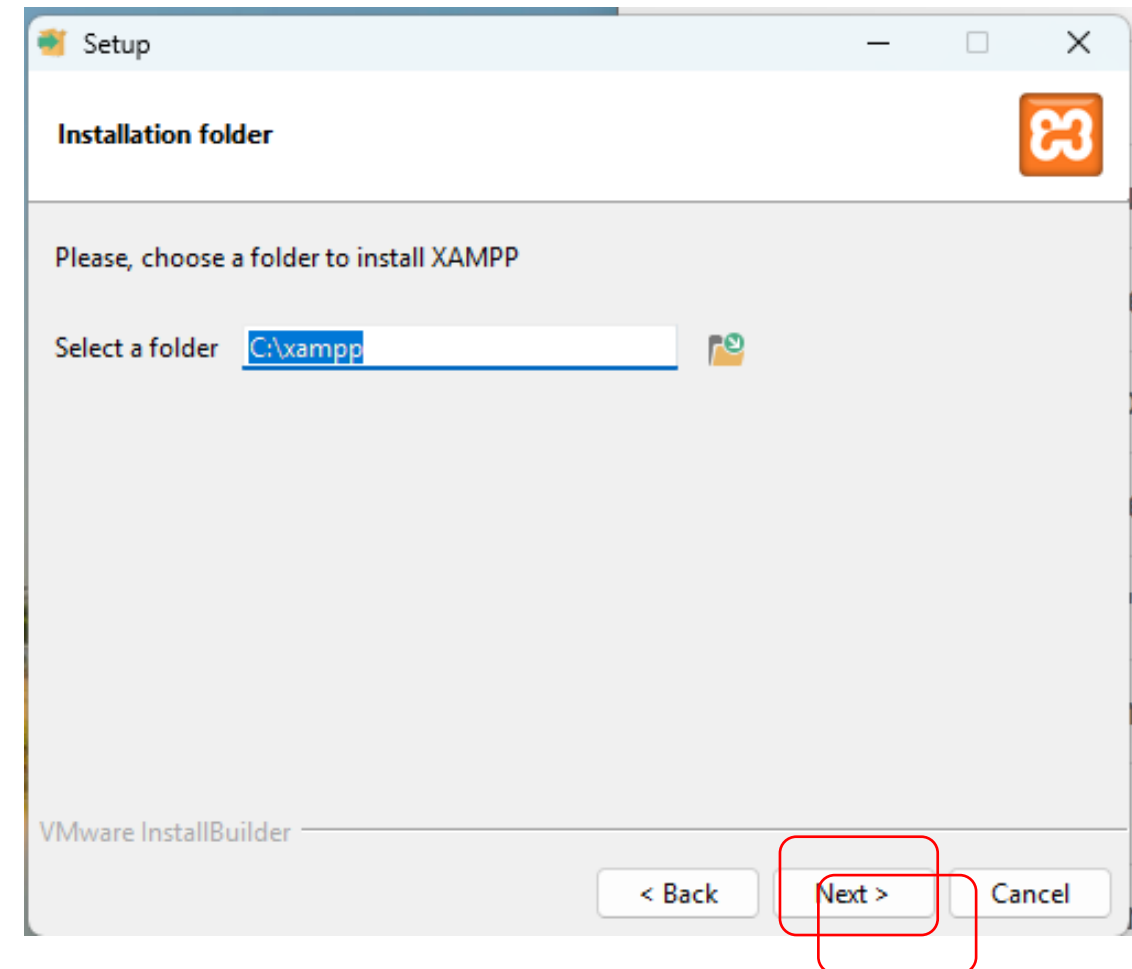
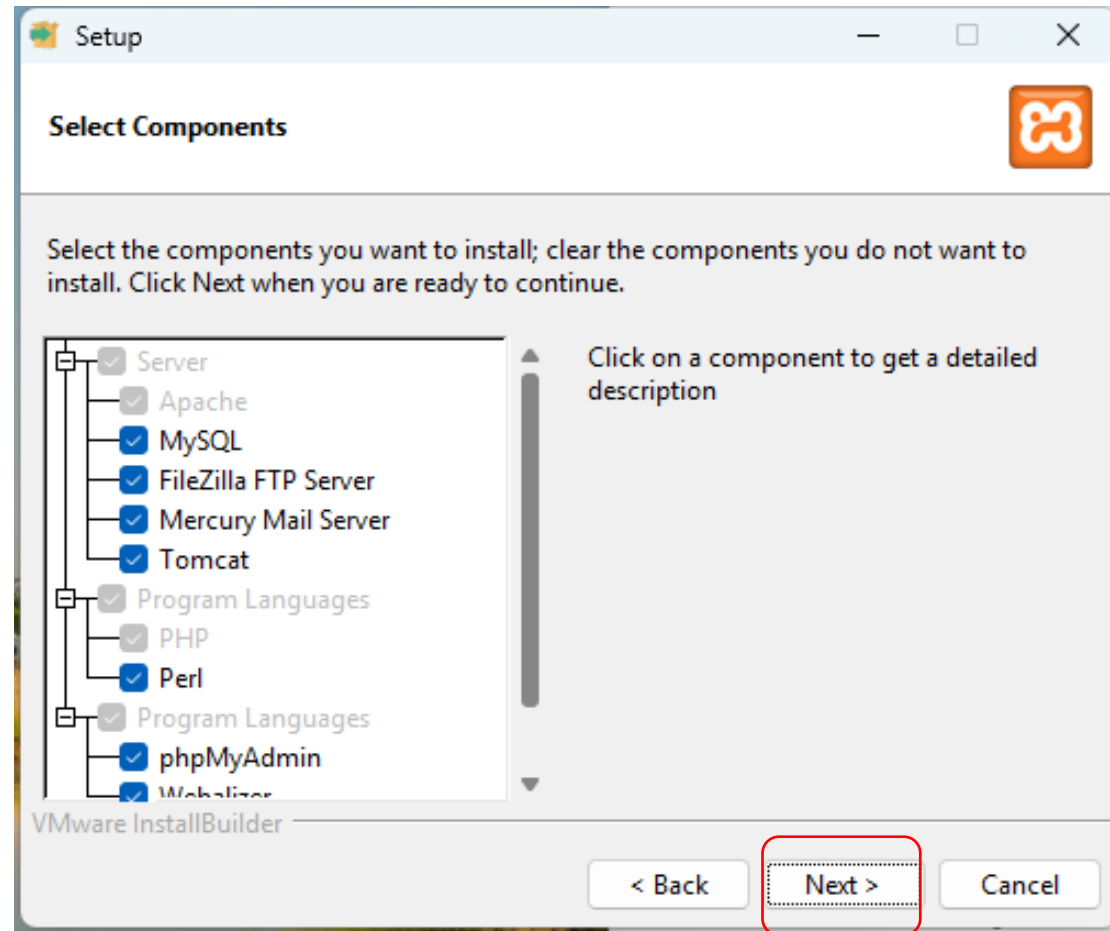
Recent Discussions

Loading discussions...

XAMPP Kurulum

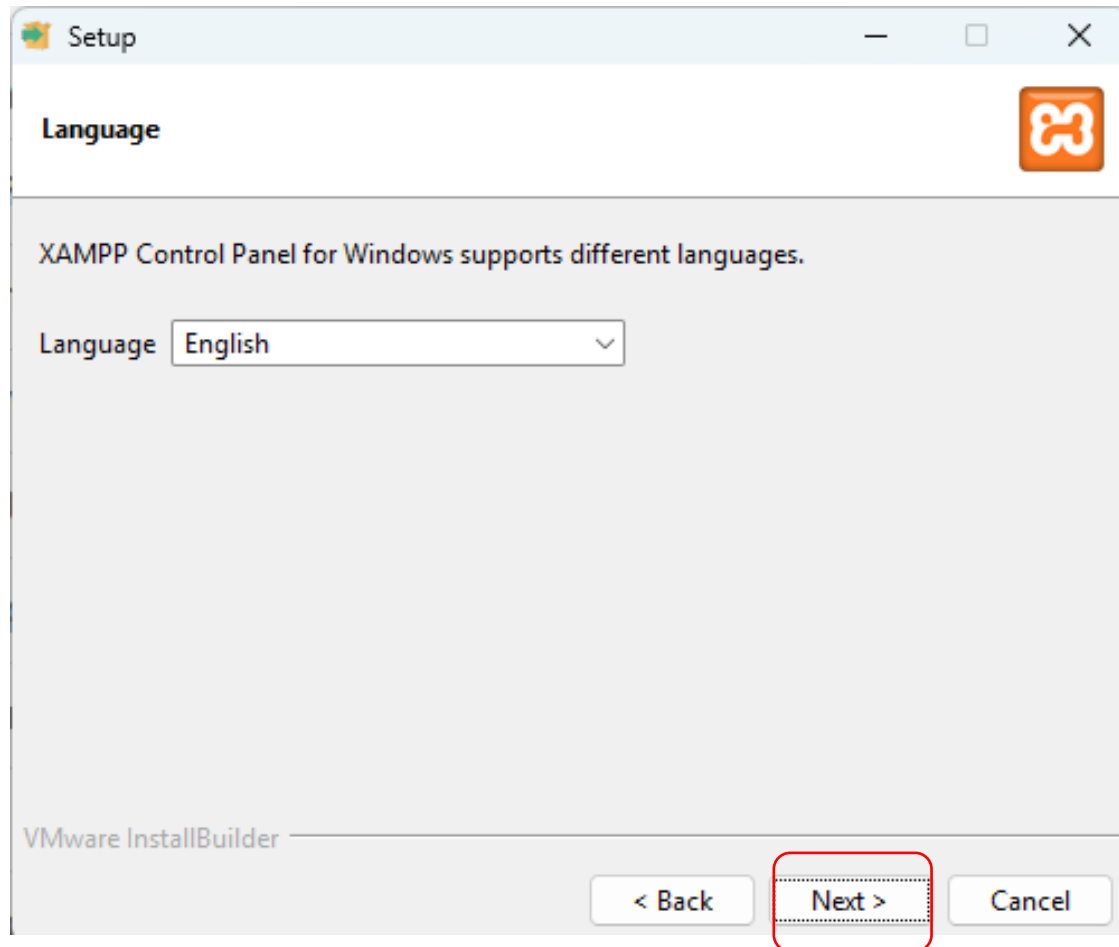
Standart seçili bileşenleri
değiştirmeyin (MySQL,
phpMyAdmin, vb. Olsun.)

Kurulum dosyasını seçili bırakın
veya seçin

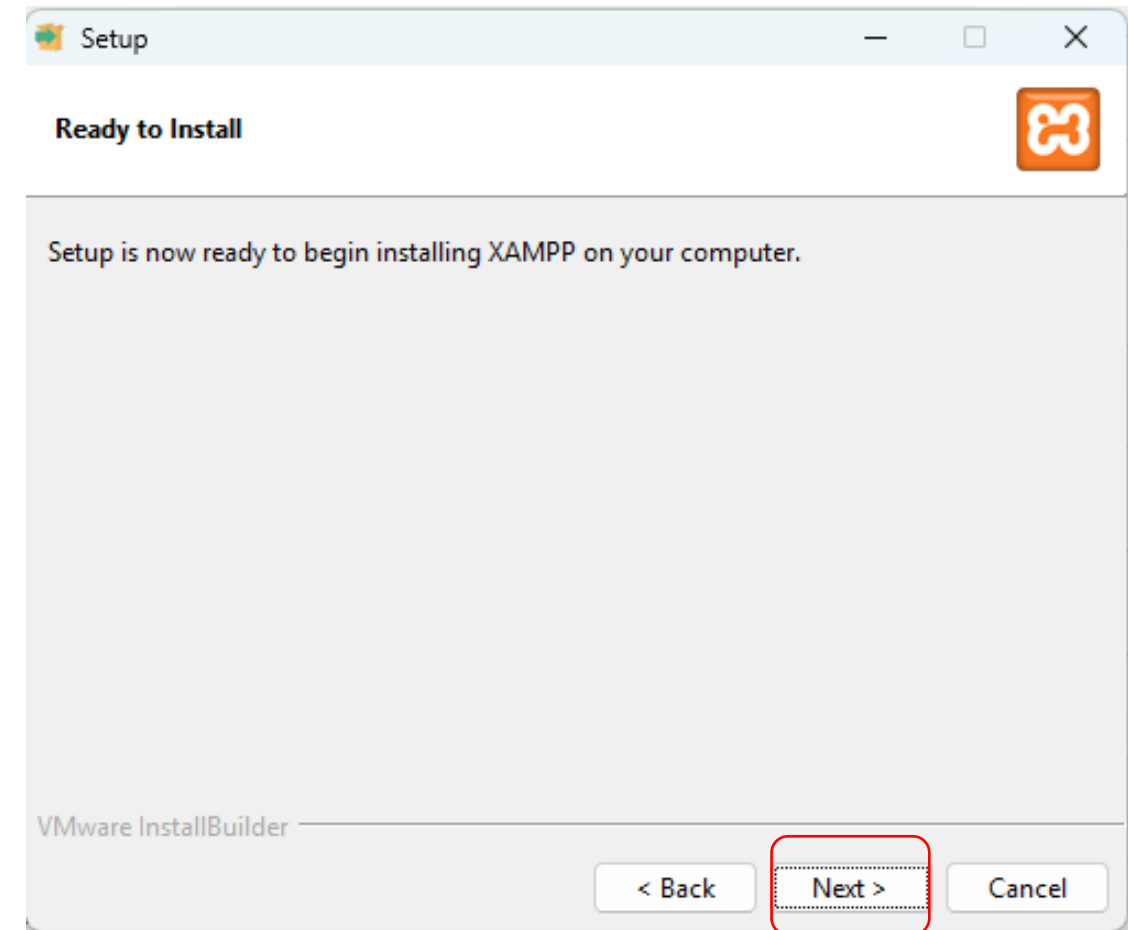


XAMPP Kurulum

Dil seçimi - English

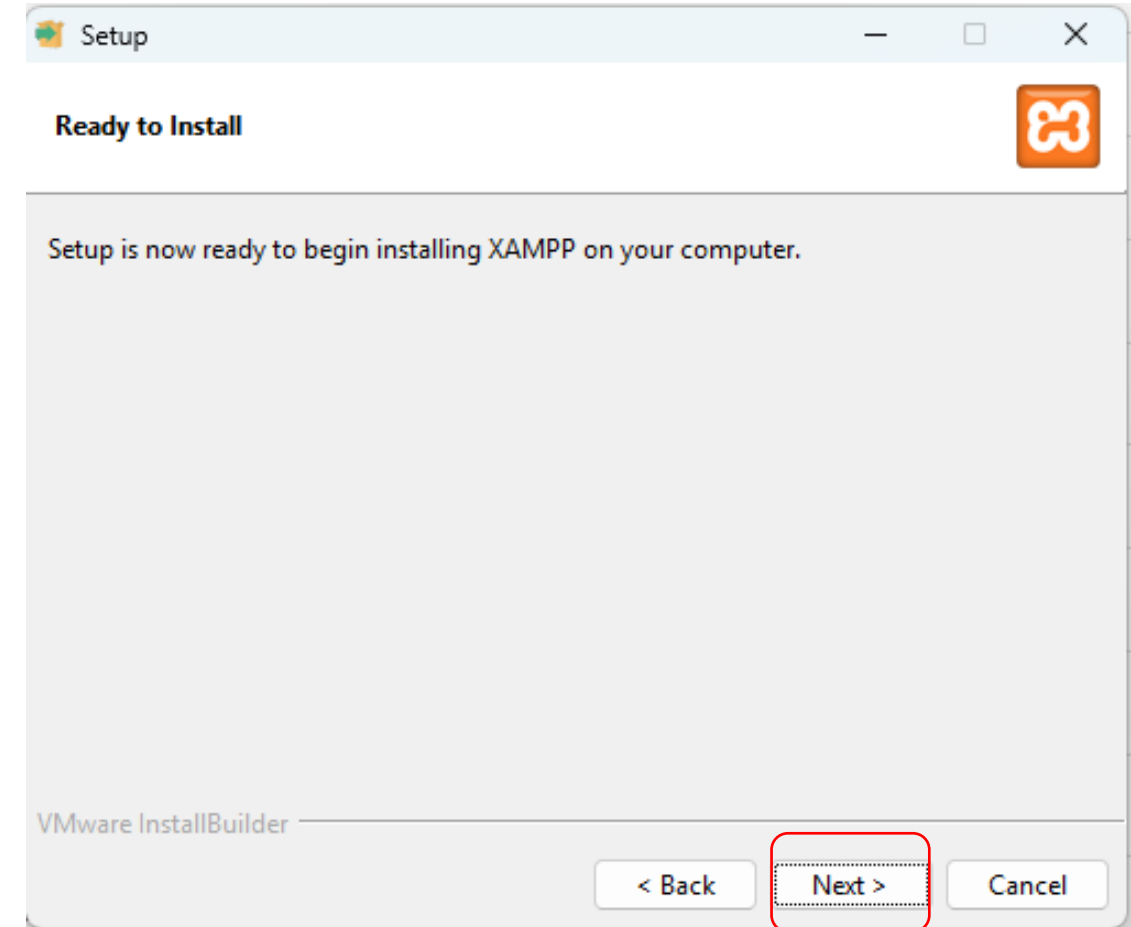
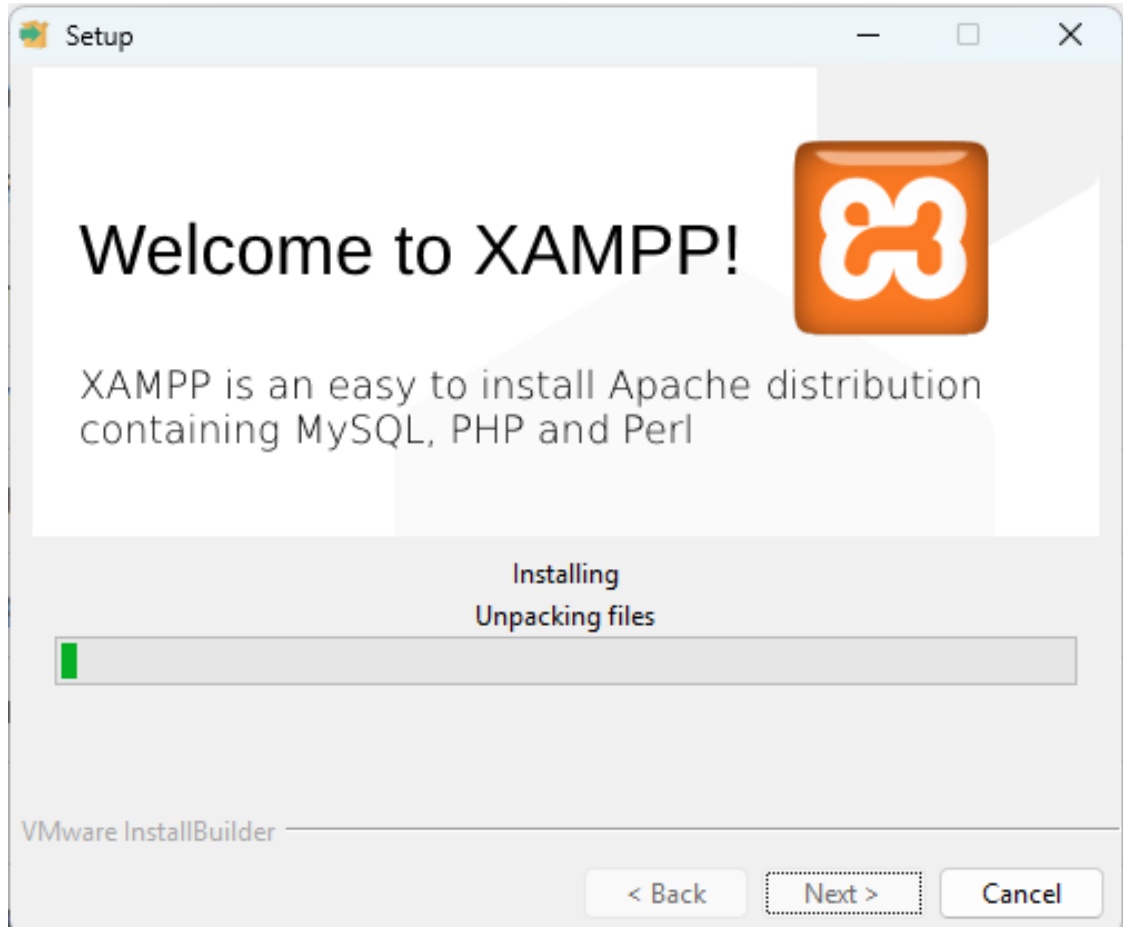


“Next”



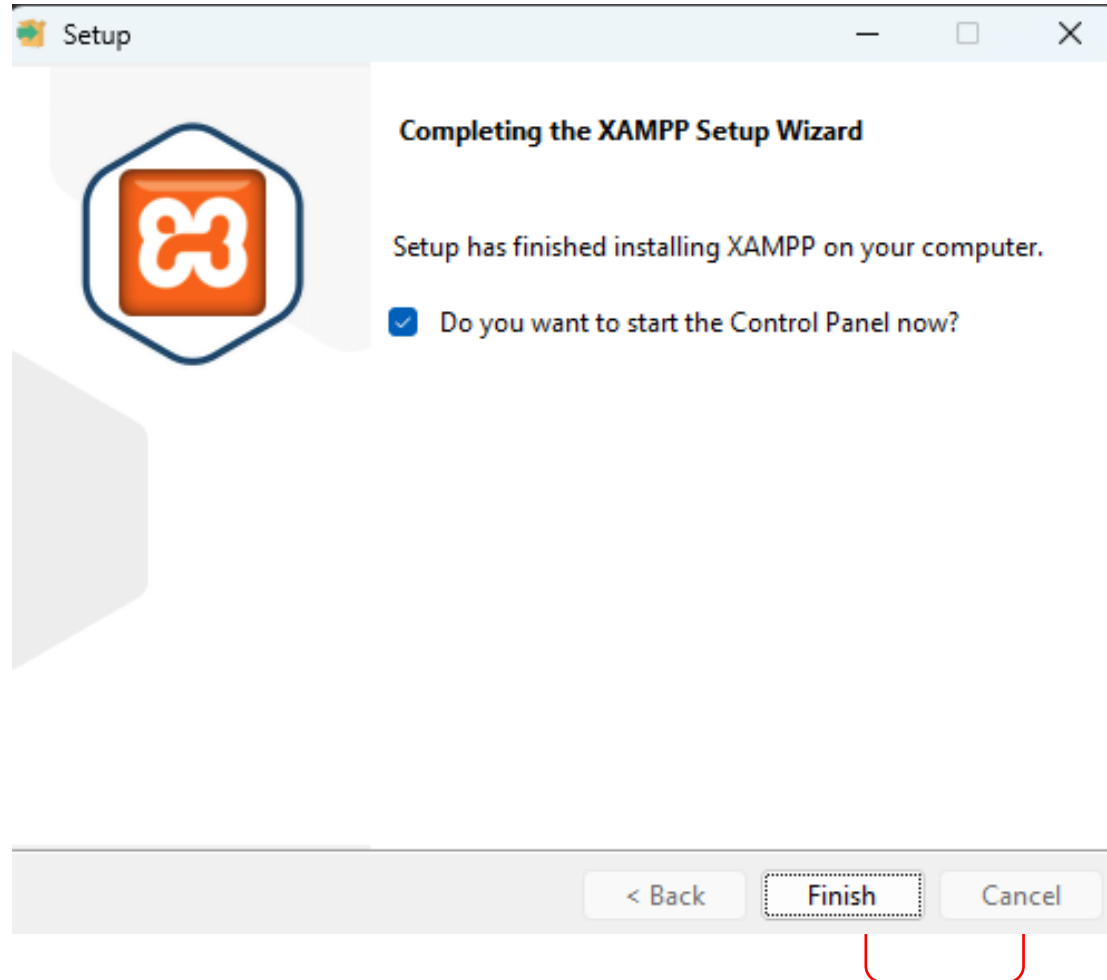
XAMPP Kurulum

Kurulumu bekleyin – Bilgisayar hızına göre 3-5 dk.

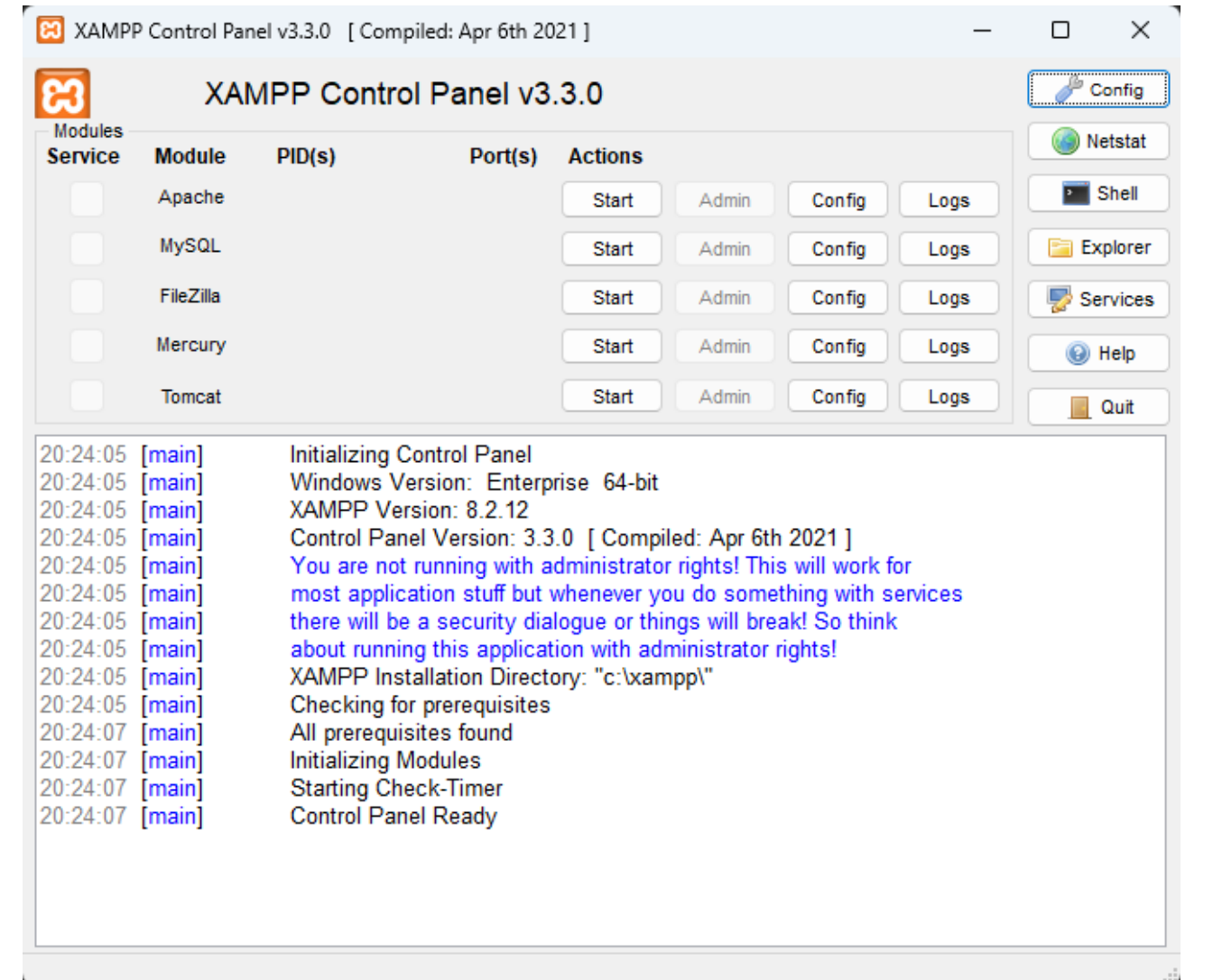


XAMPP Kurulum

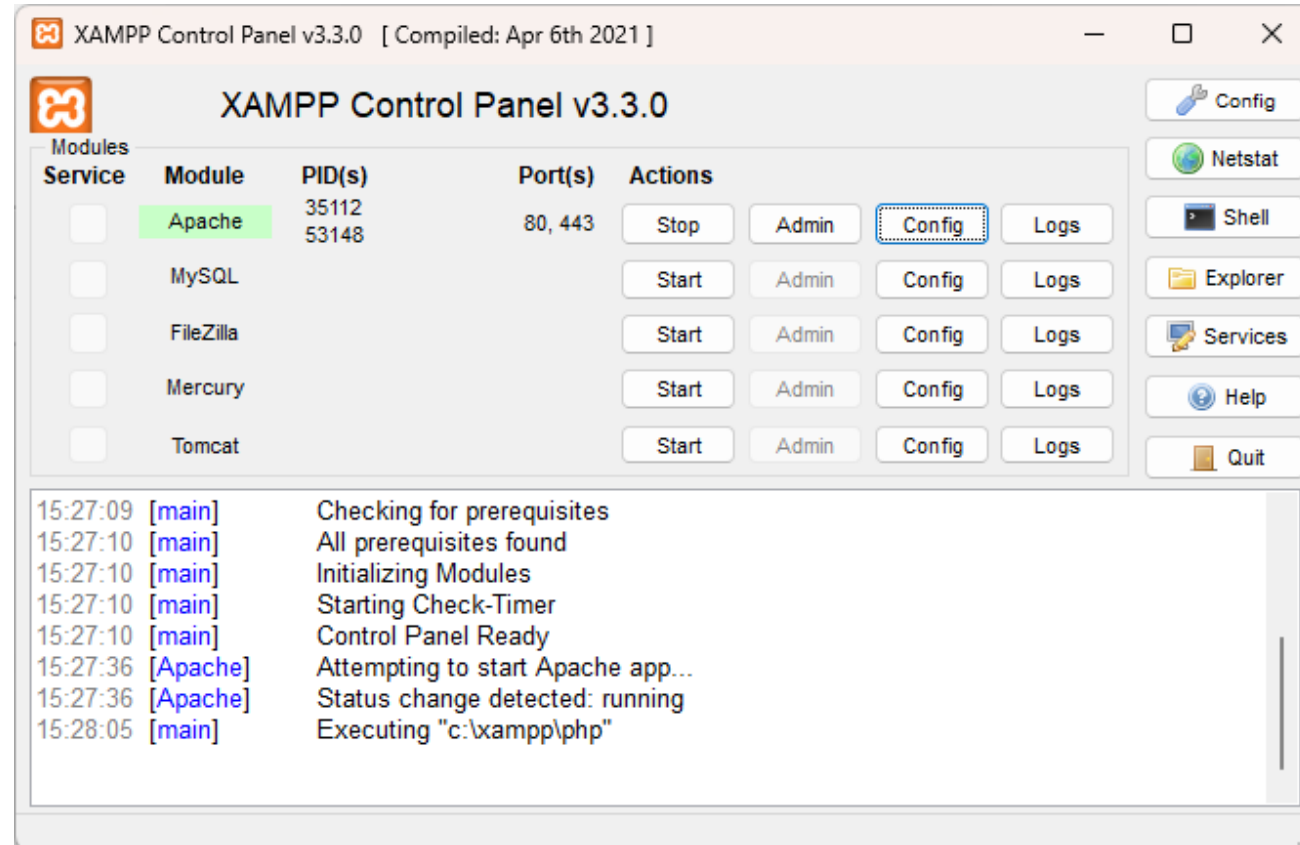
Kurulum Tamamlandı



XAMPP Çalıştırılması



XAMPP'da Yeni Bir PHP Projesi Nasıl Başlatılır?



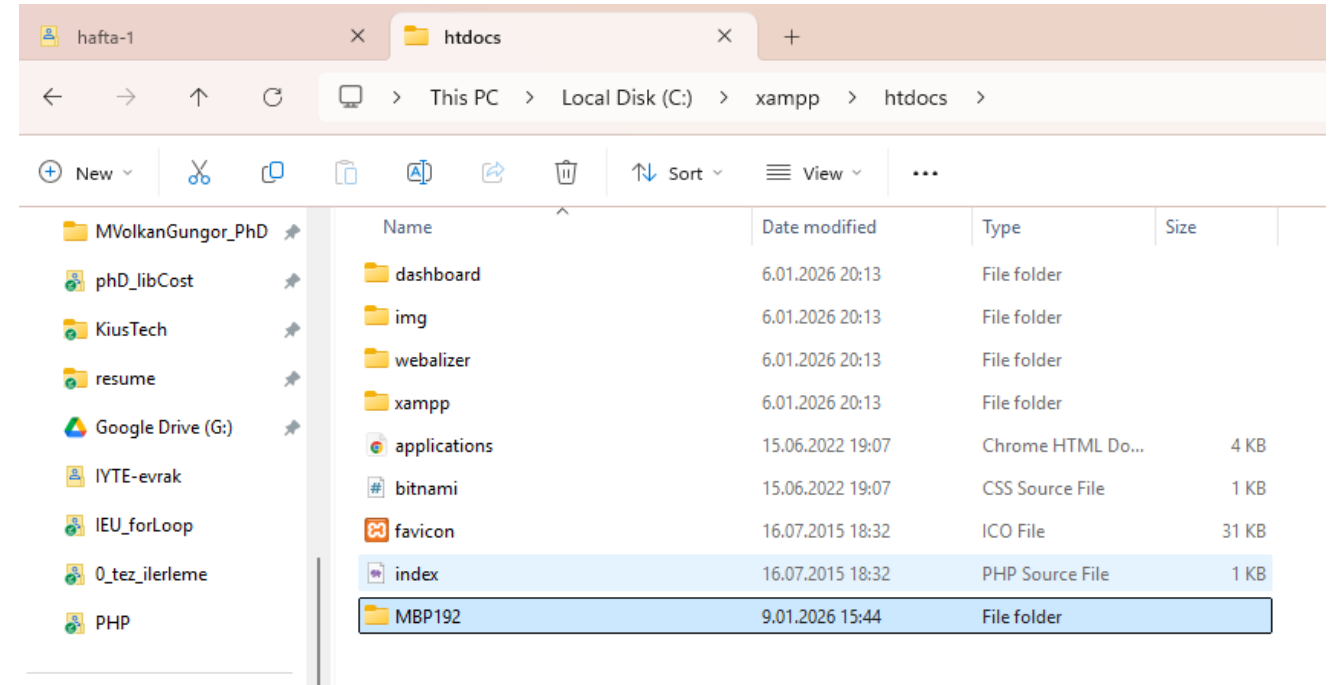
- Herhangi bir PHP programı yazmaya veya çalıştırmaya başlamadan önce **Apache** ve **MySQL**(veritabanı kullanılacaksa) servislerini başlatmalısınız.
- Her iki sunucuyu da başlattıktan sonra, Not Defteri'nde (veya bir kod editöründe) bir program yazmanız gerekir.
- Yazdıktan sonra bu dosyayı "**index.php**" olarak kaydedin.
- Ardından bu **index.php** dosyasını **C:/Program Files/XAMPP/htdocs** dizinine kopyalayın.
- Tarayıcıyı açın ve **http://localhost** yazın.
- Şimdi kodunuzu tarayıcıda çalıştırın.

XAMPP Kullanarak PHP Kodu Nasıl Çalıştırılır?

Bir PHP dosyasını çalıştırmadan önce onu nereye yazacağınızı bilmelisiniz.

- **C:\xampp\htdocs** dizinine gidin ve içinde bir klasör oluşturun. Adına **"MBP192"** diyelim. Üzerinde çalıştığınız her proje için yeni bir klasör oluşturmak iyi bir uygulamadır.
- PHP dosyaları **C:/Program Files/XAMPP/htdocs/ mbp192** içine kaydedilir.
- Demo klasörünün içinde **"index.php"** adında yeni bir metin dosyası oluşturun ve içine PHP kodunuzu yazın.
- Şimdi dosya çıktısını görmek için **XAMPP Kontrol Paneli**'ni açın ve betiğimizizin çalışacağı yerel web sunucusunu barındırmak için **Apache**'yi başlatın.
- Çıktıyı görüntülemek için tarayıcınıza gidin ve adres çubuğuna **"localhost/mbp192/"** yazın.

XAMPP dizininde "htdocs" adında bir klasör bulunur. Web sayfalarına ait tüm programlar burada çalıştırılır/saklanır.



Kurulum Kaynakları

- https://www.youtube.com/watch?v=aOPIB_gzolk
- <https://www.simplilearn.com/tutorials/php-tutorial/php-using-xampp>

ÖNERİLEN OKUMA

- ▶ <https://ikas.com/tr/blog/url-nedir-ne-ise-yarar>
- ▶ <https://www.patika.dev/blog/web-development-nedir-ne-anlama-gelir>
- ▶ <https://boosmart.com/http-durum-kodlari-nedir-seo-icin-neden-onemlidir>
- ▶ <https://accesto.com/blog/is-php-still-relevant-in-2025/>