

Go net/http Paketi - Detaylı Örneklerle

Fonksiyon / Tip	Açıklama	Örnek Kod
http.ListenAndServe	HTTP sunucusu başlatır	http.ListenAndServe(":8080", nil)
http.HandleFunc	Bir route için handler atar	http.HandleFunc("/", func(w http.ResponseWriter, r *http.Request) { fmt.Fprintln(w, "Merhaba") })
http.Handle	Bir handler objesini route'a bağlar	mux := http.NewServeMux() mux.Handle("/static/", http.FileServer(http.Dir("./public")))
http.Request	Gelen HTTP isteğini temsil eder	func(w http.ResponseWriter, r *http.Request) { fmt.Println(r.Method) fmt.Println(r.URL.Path) }
http.ResponseWriter	HTTP cevap yazmak için kullanılır	fmt.Fprintln(w, "Merhaba Dünya")
http.ServeMux	Router – birden fazla handler yönetir	mux := http.NewServeMux() mux.HandleFunc("/", homeHandler)
http.FileServer	Static dosya sunar	fs := http.FileServer(http.Dir("./public")) http.Handle("/", fs)
http.Redirect	HTTP yönlendirmesi yapar	http.Redirect(w, r, "/login", http.StatusFound)
http.Error	HTTP hata cevabı döner	http.Error(w, "Bir hata oluştu", http.StatusInternalServerError)
http.Client	HTTP istek gönderen istemci	client := &http.Client{} resp, _ := client.Get("https://example.com")
http.Get	GET isteği gönderir	resp, _ := http.Get("https://example.com") defer resp.Body.Close()
http.Post	POST isteği gönderir	resp, _ := http.Post("https://example.com", "application/json", body)
http.Cookie	Cookie nesnesi	http.SetCookie(w, &http.Cookie{Name: "token", Value: "123"})
http.ServeFile	Dosya gönderir	http.ServeFile(w, r, "index.html")
http.NewRequest	Yeni HTTP isteği oluşturur	req, _ := http.NewRequest("GET", "https://example.com", nil)
http.DefaultServeMux	Varsayılan router	http.HandleFunc("/", handler)
http.Server	HTTP sunucusunu temsil eder (timeout vs. ayarları)	srv := &http.Server{Addr: ":8080", ReadTimeout: 5*time.Second} srv.ListenAndServe()