

1.1 Görsel Alet Çantaları

GUI Toolkits

GUI, İngilizce dilinde *Graphical User Interface* deyiminin kısaltmasıdır. Türkçe’de *Görsel Arayüz* ya da *Grafiksel Kullanıcı Arayüzü* diye söylenir. İlk zamanlar, bilgisayar üst düzey dillerinin ortaya çıkışı ile yazılmaya başlanan programların hemen hepsi text olarak girilen komutlarla (command line) yönetilirdi. Programın yaptığı işler için, çoğunlukla işletim sisteminin ilgili komutları çağrılırdı. Örneğin, bir dosyayı silmek, kopyalamak, farklı kayıt ortamlarına kaydetmek gibi işler hep komut satırından verilen komutlarla yerine getirilirdi. Bu işler, bir ölçüde işletim sisteminin komutlarını bilmeyi gerektiriyordu. Ayrıca, programla etkileşime girebilmesi için, kullanıcının, program akışı sırasında gerekli komut ya da verileri girmesi gerekiyordu.

Kişisel bilgisayarların (PC) yaygınlaşmasıyla birlikte, kullanıcı sayısı artmaya başladı. Ama kullanıcıların çoğu, bilgisayarı ya da programı komut satırından yönetebilecek bilgiye sahip değildi. Ortaya çıkan bu zorluk, görsel arayüzlerin (GUI) hazırlanmasıyla aşılabildi.

Görsel arayüz (GUI), program ile kullanıcı arasında ekranda görülen bir araçtır. Arayüzde, programı yönetmek için gerekli *menüler*, *düğmeler*, *kaydırma çubukları*, *etiketler*, *yazı alanları*, *tablolar* gibi aletler (widget¹) yer alır. *Klavye*, *fare* ya da dokunmatik ekranda *dokunuşlarla* eyleme geçiri-

¹widget terimi İngilizcedeki *Window* teriminin ilk iki harfi ile *gadget* teriminin son dört harfinin birleştirilmesiyle türetilmiştir. *görsel arayüz aracı* anlamında kullanılıyor.

lebilirler. Bu gün yaygın işletim sistemlerinin hemen hepsi görsel arayüzlere sahiptir. Örneğin, Windows işletim sistemini, işletim sistemine ait hiç bir komut yazmadan, görsel arayüzü ile yönetebilmekteyiz. İşletim sistemi dışında pek çok program görsel arayüze sahiptir.

MS Office paketine ait *Word*, *Excel*, *Power Point* gibi programların hepsi kendi görsel arayüzlerine sahiptirler. Onları kullanan kullanıcı, hiç komut yazmadan, ekranda görünen bileşenlere (widget) tıklayarak istediği işi yapabilir. Windows dışındaki platformlar için de benzer olgu vardır. Tabii akıllı telefonlar, tabletler gibi taşınabilir (mobil) cihazları da bu gruba katmak gerekir.

1.2 WIMP

Görsel Arayüz Dili

Giderek, görsel arayüz araçlarından oluşan bir tür grafiksel dil oluştu. WIMP ("window, icon, menu, pointing device") denilen bu araçlar, bu gün oldukça standart hale gelmiştir; hemen her arayüz onları kullanır. İngilizce'deki *gadget* sözcüğüne benzetilen *widget* terimi, görsel arayüzlerde kullanılan *düğmeler*, *menüler*, *ağaçlar*, *listeler*, *tablolar*, *text alanları* gibi görsel aletler topluluğunu ifade eden genel terimdir.

1.3 MVC

MVC: Model-View-Controller

Model-View-Controller (MVC), Yazılım Mühendisliği'nde kullanılan bir yöntemdir. Bir çok alana uygulanır. MVC'nin asıl amacı içerik (model) ile görüntüsünü (view) birbirinden ayırmaktır. Bu iş başarılırsa, aynı içeriği farklı görüntülerle sunma olanağı doğacaktır. Gerçekte, bu mimari yapıyı, çoğunlukla farkında olmadan her gün kullanıyoruz. Örneğin, bir text dökümanının içeriğini hiç değiştirmeden, onu *MS Word* ile farklı biçimlerde sunabiliriz. Aynı işi *HTML* dökümanlarında da yapabiliriz. Tabii, içerik ile görüntüyü ayırdığımızda, onların birbirlerinden kopmamasını, ilişkilerinin devamını sağlayan denetime gerekseme vardır. Bu işi de MVC'nin C harfine karşılık gelen *Controller* yapar.

MVC'nin ortaya çıkışı 1979 yıllarına kadar geriye gider. 1996 yılında Netscape firması MVC mimarisine dayalı olan *Internet Foundation Classes (IFC)* kütüphanesini hazırladı. Hemen arkasından *Sun* şirketi *Java Foundation Classes (JFC)* kütüphanesine geçti. Java'nın AWT ile başlayan görsel

arayüz kütüphanesi yaratma çabaları *javax.swing* kütüphanesine uzandı. Giderek bir çok firma ve kişi GUI hazırlamaya yarayan programlar yazdı.

1.4 Alet Çantaları

GUI hazırlamaya yarayan programlara *arayüz alet çantası* (GUI toolkit) denilir. Arayüz alet çantalarının hemen hepsi nesne tabanlıdır. Her *widget*, ait olduğu alet çantası içinde bir sınıftır; dolayısıyla bir veri tipidir. Arayüz hazırlamaya yarayan çok sayıda alet çantası vardır. Bir önem sırasına konulmaksızın, görsel alet çantalarının (GUI toolkits) başlıcaları Liste 1.1'de verilmiştir.

Liste 1.1.

FLTK

İşletim sisteminden bağımsız (lightweight) araç kutusu.

FOX toolkit

İşletim sisteminden bağımsız (lightweight) araç kutusu.

FrugalWidgets

Win32 için küçük bir araç kutusu.

GNUstep

Cocoa'nın Objective-C için hazırladığı ücretsiz bir araç kutusu (öncülü NeXT OpenStep).

GTK+

GNOME uygulamaları için hazırlanmış GIMP araç kutusu.

GraphApp

C dili ile yazılmış, Windows ve Linux için kullanılabilen, açık kaynak kodlu araç kutusu.

IUP (software)

Platform bağımsız araç kutusu.

Juce

Platform bağımsız araç kutusu.

JX

Uygulama geliştiriciler için hazırlanmış araç kutusu.

Microsoft Foundation Classes

Windows için en yaygın kullanımı olan araç kutusu.

WTL

Microsoft Foundation Classes'e alternatif olan platform bağımsız araç kutusu.

Motif

Unix sistemlerde X Windows tabanlı bir widget araç kutusu.

Qt

KDE, Google Earth, Skype tarafından kullanılan LGPL lisanslı araç kutusu.

Standard Widget Toolkit (SWT)

Java için hazırlanmış alet çantası.

Swing

Java için hazırlanmış alet çantası.

Tk

Tcl, Perl ve Python programlarında kullanılan araç çantası.

Ultimate++

C++ için bir widget araç çantası.

VCL

(Visual Component Library) Microsoft Windows için hazırlanmış bir widget çantasıdır. Borland firması tarafından geliştirilen bu çanta Delphi, C++ Builder tarafından kullanılır. Object Pascal dili ile yazılmıştır.

wxWidgets

Platform bağımsız araç kutusu.

XForms

Platform bağımsız araç kutusu.

Zinc

C++ için hazırlanmış alet çantası.

1.5 tkinter

Python2 için yazılan Tkinter paketi Python3 için tkinter adını almıştır. Buna göre bazı modüllerin adları da değişmiştir. Modül adlarında yapılan başlıca değişiklikler şöyledir:

Tkinter	tkinter
Tkinter	tkinter
tkMessageBox	tkinter.messagebox
tkColorChooser	tkinter.colorchooser
tkFileDialog	tkinter.filedialog
tkCommonDialog	tkinter.commondialog
tkSimpleDialog	tkinter.simpledialog
tkFont	tkinter.font
Tkdnd	tkinter.dnd
ScrolledText	tkinter.scrolledtext
Tix	tkinter.tix
ttk	tkinter.ttk

Tablo 1.1: Tkinter ve tkinter'de modül adları

tkinter terimi Tk **interface**'in kısaltmasıdır. Tkinter Python2 için ve *tkinter*, Python 3 için GUI (Graphical User Interface) alet kutularındır (paket). Görsel arayüz hazırlamak için gerekli olan modülleri, sınıfları, fonksiyonları, sabitleri vb içerirler. Unix ve Windows işletim sistemleri üzerinde çalışabilirler. Tk alet çantası, aslında Python dilinin bir parçası değildir; ActiveState tarafından geliştiriliyor. Ama Python yüklendiğinde, *tkinter* paketi de birlikte yüklenir. Python ile GUI hazırlamak için Python dilinin bilinmesi gerekir. Python dilini bilmeyenler, önce o işi yapmalıydılar (bkz. [?], [?], [?], [?])

tkinter'in dayandığı Tk alet çantasının bir çok dile uyarlanmış sürümleri vardır. Bu kitapta *Python3* için *tkinter* programını öğreneceğiz. Python2 sürümü için *Tkinter* adını alan modül Python3 için *tkinter* adını almıştır (Python3 için modül adının küçük harfle yazıldığına dikkat ediniz). Anımsayacaksınız, Python2 sürümü ile Python3 sürümü arasında önemli farklar vardır. Dolayısıyla *Tkinter* modülü ile *tkinter* modülü arasında da önemli farklar vardır. Bu kitapta Ppython3 için yazılan *tkinter* modülünü ele alacağız.

Hemen belirtelim ki, Python ile GUI hazırlamak için çok sayıda alet çantası vardır. *tkinter* onlardan birisidir. Görsel arayüzlerde kullanılan ve adına *widget* denilen Window, Frame, , Label, Button, Radiobutton vb araçlar oldukça standart duruma gelmişlerdir. Dolayısıyla, GUI hazırlamaya yarayan alet çantalarının hemen hepsi Tablo 1.3 ile verilen standart widget'leri içerir. Üstelik, kullanılma yöntemleri de, ayrıntılar dışında esasta aynıdır. Dolayısıyla, bir GUI alet çantasını kullanmayı öğrenenler, başka birisine kolayca geçiş yapabilirler. Örneğin, Java programcılarının kullan-

dığı *AWT* ve *Swing*'i bilenlerin Python için *Tkinter*'e geçişleri çok kolay olacaktır. Python için yazılan başka GUI alet çantaları vardır. Başlıcaları şunlardır:

Camelot	Cocoa	Kivy
PyObjC	PySide	PyQT
pyjs desktop	Qt	wxPython

Tablo 1.2: Başlıca Python GUI Çantaları

Python ile görsel arayüz hazırlarken, çoğunlukla, *tkinter* paketi dışında bir şeye gerekmeniz olmaz. Daha çok ayrıntı için bkz. [?].

1.6 Neden tkinter

tkinter modülü *Python3* ile birlikte gelir; onu ayrıca yüklemeye gerek kalmaz. Böyle oluşu, programcı için bir kolaylıktır. Ayrıca, *tkinter*'in son sürümü, öteki GUI programlarıyla boy ölçüşecek kadar gelişkindir.

Ancak, bu kitabın yazıldığı sırada, *tkinter* alet çantası için çok gelişkin bir IDE (Integrated Development Environment) yoktu. Oysa, örneğin, *pyQT* için gelişkin bir IDE programı vardır. Kullanıcı GUI alet çantasını seçerken, lisanslı ya da lisanssız oluşu, iyi bir IDE'ye sahip olup olmadığı gibi etmenleri gözönünde tutmalıdır. Sözdizimlerdeki farklar bir yana bırakılırsa, bütün GUI çantaları aynı kurguyla çalışır. IDE yerine komutlar yazılacaksa, belki en basiti *tkinter* modülüdür. Ama IDE kullanılacaksa, *pyQT* benzeri araç kutuları tercih edilmelidir.

Tk, 1990 yılında *Tcl* script dili için *John Ousterhout* tarafından yazıldı. Kolay öğrenilen bir dil olduğu için hızla yaygınlık kazandı. Bir çok programcı Tk dilini *Tcl* dilinden bağımsız olarak kullanmaya başladı. Bu gün *Tkinter*'in *Perl*, *Ada*, *Python*, *Ruby*, *Common Lisp* gibi diller için yazılmış sürümleri vardır. Python için Tk diline, *Tkinter* denilir.

Tk dilindeki standart widgetleri Python için sırayla ele alacağız. *tkinter*'in standart widgetleri Tablo 1.3'de verilmiştir.

Tk üst düzey pencereler için Tablo 1.4'deki açılır pencereleri sunar:

Tk, widgetlerin nasıl konuşlanacağını belirleyen üç tane geometri yöneticisine sahiptir. Bunlar Tablo 1.5'de belirtilmiştir.

button	canvas	checkbutton
combobox	entry	frame
label	labelframe	listbox
menu	menubutton	message
notebook	tk_optionMenu	panedwindow
progressbar	radiobutton	scale
scrollbar	separator	sizegrip
spinbox	text	treeview

Tablo 1.3: Tk Dilindeki Standart Widget'ler

Adı	İşlevi
tk_chooseColor	Renk seçmeye yarayan pop up diyalog kutusu
tk_chooseDirectory	Dizin seçmeye yarayan pop up diyalog kutusu
tk_dialog	Modal diyalog yaratır ve yanıtı bekler
tk_getOpenFile	Dosya açmaya yarayan pop up diyalog kutusu
tk_getSaveFile	Dosya kaydetmeye yarayan pop up diyalog kutusu
tk_messageBox	Mesaj kutusu açar ve yanıtı bekler
tk_popup	Pop up menü açar
toplevel	Üst düzeyli widgetleri yaratır ve düzenler

Tablo 1.4: Tkinter'in Diyalog Araçları

Adı	İşlevi
place	Widget'in mutlak konumunu belirler
grid	Widgetleri bir ızgaraya (grid) yerleştirir
pack	Widgetleri paketleyip uygun boşluklara yerleştirir.

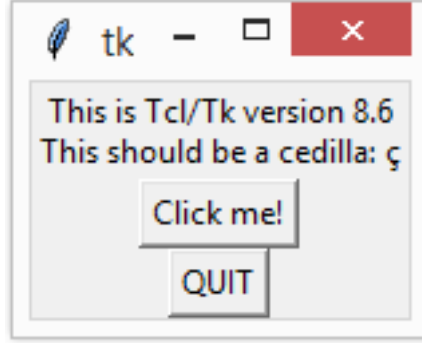
Tablo 1.5: Tkinterin Konuşlandırma Araçları

1.7 Bilmek İsterseniz Okuyun

Sisteminizde tkinter'in yüklü olup olmadığını anlamak için komut satırına

```
1| python -m tkinter
```

yazınız. Ekran Şekil 1.1 gibi basit bir Tk arayüzü geliyorsa, sisteminizde *tkinter* yüklüdür.



Şekil 1.1: tkinter Yüklü mü?

tkinter, Tk için gerekli arayüzler ile *python.constants* modülünü ve arayüz hazırlamak için gerekli olan başka modülleri içerir. Dolayısıyla, görsel arayüz hazırlarken yalnızca tkinteri çağırmak (import) yeterli olacaktır. Bu işi

```
import tkinter
import tkinter as T
from tkinter import *
```

deyimlerinden birisiyle yapabiliriz. Üçüncü deyim, tkinter'in aduzayında olan her şeyi çağırır. Çoğunlukla bu deyim kullanacağız. Bazı sistemlerde birinci ve ikinci deyim çalışmayabilir.

1.8 Rölyef Stilleri

Kenar İşlemeleri

Görsel arayüz için kullanılan düğme gibi araçların ekrandaki görüntüsü taş ya da tahta işlemlerde kullanılan oyma ya da kabartma resimlere benzetilir. Bilgisayar grafiklerinde widgetlerin kenarlarının görüntüsü için kullanılan beş farklı işleme türü vardır.



Şekil 1.2: Widgetler İçin Rölyef Stilleri

FLAT, RAISED, SUNKEN, GROOVE, RIDGE

(Düz, Kabartma, Oyma, Oluk, Avlu)

1.9 Pixel

İngilizce'de resim parçası anlamına gelen "picture element" birleşik kelimesinin ilk ikişer harfinden ürettiği "pixel" kelimesi, Türkçe karşılığı "gözet" olarak önerilmiştir. Sayısal (dijital) görüntülerin adreslenebilir en küçük parçasıdır. Adreslenebilir derken, kastedilen şey şudur: Görüntü birimi üzerinde, başlangıç noktası sol üst köşede olacak şekilde kurulan koordinat sisteminde yeri belirlenebilen en küçük parçaların her birine bir pixel denilir. Alışılmış biçimiyle sözkonusu koordinat sistemindeki her pixel bir noktadır. Ama bu noktalar, matematikte olduğu gibi eni-boyu olmayan sonsuz sayıda noktalardan oluşmaz. Genellikle, pixel küçük bir dikdörtgen biçimindedirler. Görüntü aygıtı fiziksel bir aygıttır. Onun gösterebileceği pixel sayısı, teknolojinin yetenekleriyle sınırlıdır ve şimdilik sonlu sayıdadır. Birim alana (genellikle inç kare) düşen pixel sayısı ne kadar çoksa, görüntü kalitesi o kadar iyi olur.

Görüntünün renklendirilmesi renk derinliği denilen bir sistemle olur. Yaygın iki sistem vardır. RGB (red, green, blue) sisteminde kırmızı, yeşil ve mavi renklerin karışımıyla her renk elde edilebilir. Karışıma giren renklerin şiddeti (intensity) rengi belirler. Her rengin şiddeti 0-255 arasında değişen bir sayı ile belirlenir. Örneğin (255,0,0) tam kırmızı rengi, (0,0,255) tam mavi rengi oluşturur. Matbaacılıkta RGB sistemi yerine, dört rengin karışımından oluşan CMYK(cyan, magenta,yellow,black) sistemi kullanılır.

1.10 Raster Graphics

Bilgisayar grafiğinde şekli *noktalar matrisiyle* (dot matrix data structure) yöntemiyle belirlenen şekillerdir. Görüntü aygıtında pixel'lerin birleşmesiyle, ekranda şekil (resim) oluşur.

1.11 Bitmaps

Bitmap, tekbitlik raster grafiklerinden oluşan şekildir. Bitmap (biteslem, ikieşlemlı) bilgisayarın kayıt ortamına resim kaydetme formatıdır. Bilindiği gibi, bilgisayarlarda resimler farklı formatlarda saklanır. Her birisinin kullanım amacı farklıdır.

Bitmap, .bmp formatının genelleşmişidir. Bilgisayar dilinde, bitler 0 ve 1 den oluşan veri tipidir. Daha alışılmış deyimle, bilgisayarda her şey *binary* sistemle yazılır. Teknik açıdan *bitmap*, bir resmi oluşturan bitleri bilgisayar ekranında görüntüye dönüştürür. Bu açıdan bakınca, bitmap, *bit*'leri resme dönüştüren bir dönüşümdür. Bu dönüşümün ayrıntıları bu kitabın değil, bilgisayar ekranında görüntü oluşturma konusuyla ilgilidir.

Uygulamada bitmap diye adlandırılan şekiller vardır. *Bitmap*, tekbitlik raster grafiklerinden oluşan şekildir. İkona benzerler, ama ikonlar çok katmanlı ve renkli olurlar. Bitmap ise tek renkli ve tek katmanlıdır. Şekil 1.3 bazı bitmap örneklerini gösteriyor.



Şekil 1.3: Bitmap Örnekleri

Soldan sağa doğru bitmap düğmeleri:

"error", "gray75", "gray50", "gray25", "gray12", "hourglass", "info",
"questhead", "question", "warning"