

# Why R? Turkey

## R Programlama Diline Giriş

Ahmet Uraz Akgül

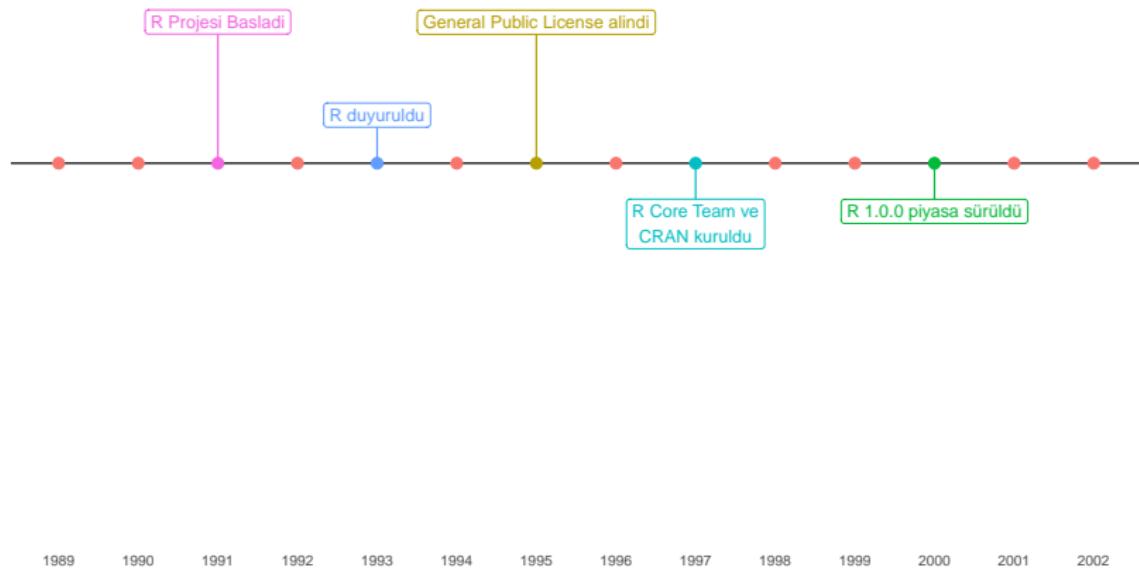
Nisan 15, 2022

## (R)oss Ihaka ve (R)obert Gentleman



Figure 1: Ross Ihaka ve Robert Gentleman, The R Project - Ross Ihaka

# R'ın Kısa Bir Tarihi



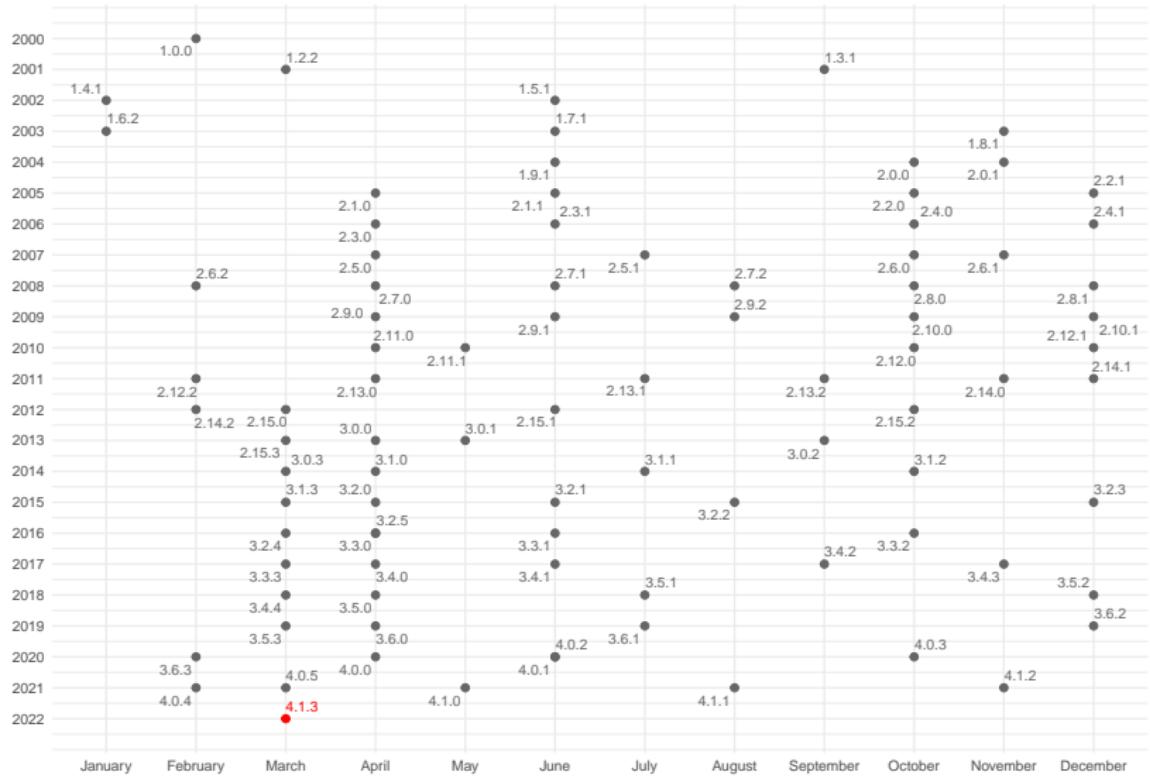
$$R = S \text{ (görünüm)} + \text{Scheme (uygulama&anlamsal)}$$

## 1. INTRODUCTION

This article discusses some issues involved in the design and implementation of a computer language for statistical data analysis. Our experience with these issues occurred while developing such a language. The work has been heavily influenced by two existing languages—Becker, Chambers, and Wilks' S (1985) and Steel and Sussman's Scheme (1975). We felt that there were strong points in each of these languages and that it would be interesting to see if the strengths could be combined. **The resulting language is very similar in appearance to S, but the underlying implementation and semantics are derived from Scheme.** In fact, we implemented the language by first writing an interpreter for a Scheme subset and then progressively mutating it to resemble S.

Figure 2: R: A Language for Data Analysis and Graphics, Ross IHAKA and Robert GENTLEMAN

# R Sürümelerinin Çıkış Tarihleri



Veriler <https://cran.r-project.org/bin/windows/base.old/> adresinden alınmıştır.

# R'ı (+ R Studio'yu) neden seviyorum?

- ▶ Ücretsiz
- ▶ Kodlama imkanı ~ özgürlük
- ▶ Katkı sağlanabiliyor; sağlayanlardan faydalanaabiliyor
- ▶ Güçlü iletişim, hızlı dönüş
- ▶ İstatistik ve Ekonometri alanında iddialı
- ▶ Görselleştirme gücü
- ▶ Diğer sunduğu imkanlar
- ▶ Entegre olabiliyor

# R'ın Popüleritesi

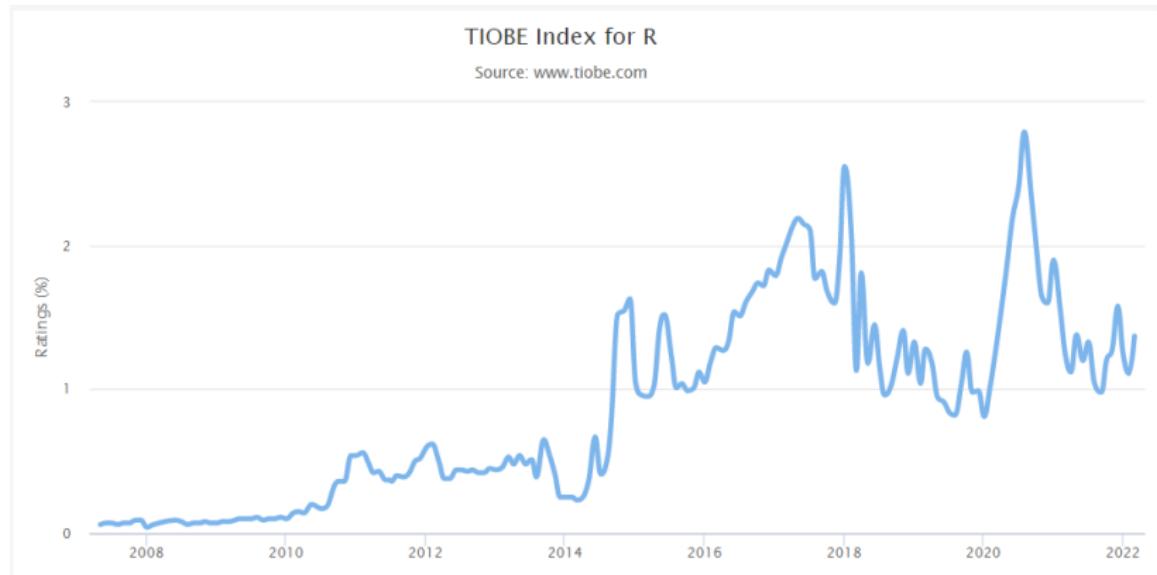


Figure 3: <https://www.tiobe.com/tiobe-index/r/>

# R'ın Popüleritesi

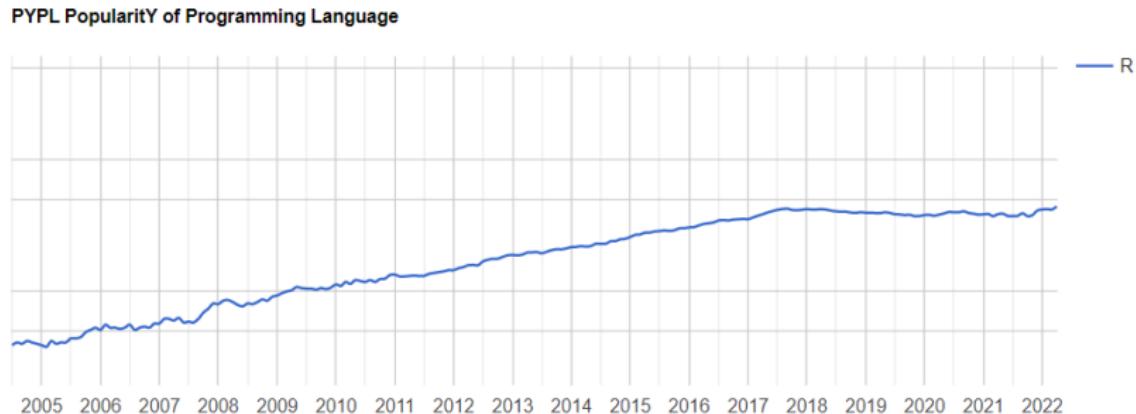


Figure 4: <https://pypl.github.io/PYPL.html?country=>

# R'in Popüleritesi

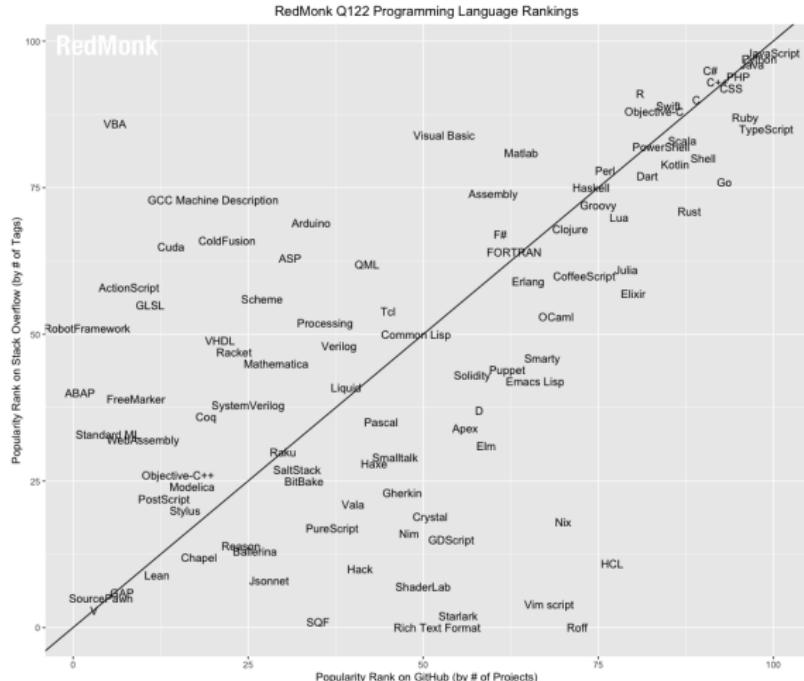


Figure 5:

<https://redmonk.com/sogrady/2022/03/28/language-rankings-1-22/>

# R'ın Popüleritesi

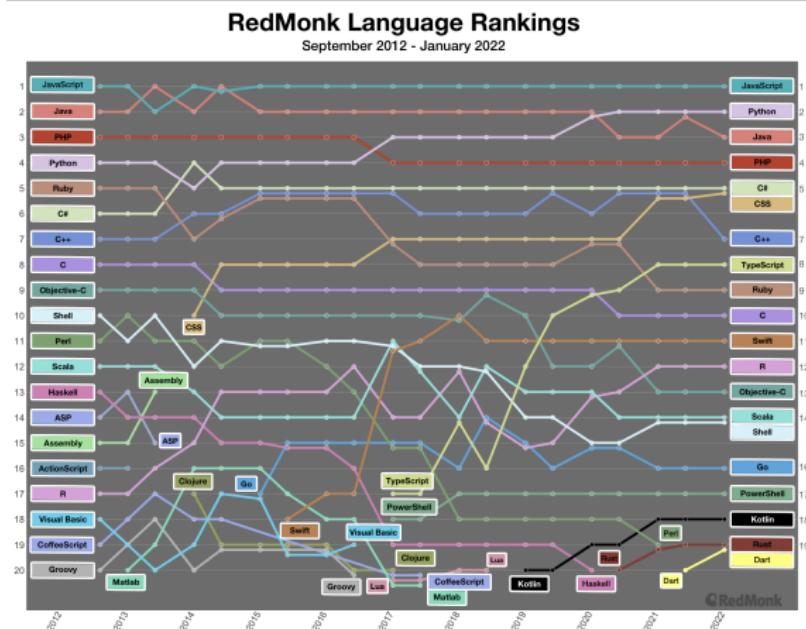


Figure 6: <https://redmonk.com/rstephens/2022/03/28/top-20-jan-2022/>

# Akademi İlk Sırada!

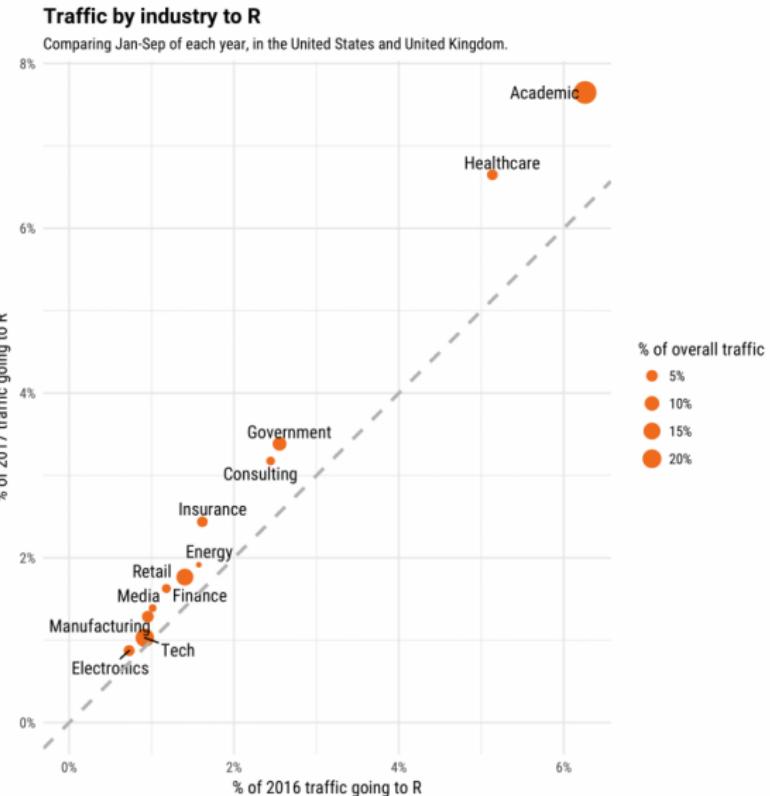


Figure 7: <https://stackoverflow.blog/2017/10/10/impressive-growth-r/>

# Neler Öğreneceğiz?

- ▶ R ve R Studio'nun Kurulumu
- ▶ Dark vs. Light Mode
- ▶ İlk Komutlar, Atama, Obje
- ▶ Operatörler
- ▶ #Yorum Satırı
- ▶ Paketler
- ▶ “Basit” Hatalar
- ▶ Script

# Neler Öğreneceğiz?

- ▶ Veri Tipleri (Numeric, Integer, Character, Logical)
- ▶ Veri Yapıları (Scalar&Vector, Matrix, Array, List, Factor, Data Frame)
- ▶ NULL, NaN, NA
- ▶ Fonksiyonlar (Base, Oluşturma, Üç Nokta)
- ▶ If - Else If - Else Koşul İfadeleri
- ▶ For Döngüsü
- ▶ While Döngüsü
- ▶ Paket İncelemesi (TCMB Örneği)

# R ve R Studio'nun Kurulumu

https://www.r-project.org/



[Home]

Download

CRAN

R Project

About R  
Logo  
Contributors  
What's New?  
Reporting Bugs  
Conferences  
Search  
Get Involved: Mailing Lists  
Get Involved: Contributing  
Developer Pages  
R Blog

R Foundation

## The R Project for Statistical Computing

### Getting Started

R is a free software environment for statistical computing and graphics. It compiles and runs on a wide variety of UNIX platforms, Windows and Mac OS. To [download R](#), please choose your preferred [CRAN mirror](#).

If you have questions about R like how to download and install the software, or what the license terms are, please read our answers to [frequently asked questions](#) before you send an email.

### News

- [R version 4.2.0 \(Vigorous Calisthenics\) prerelease versions](#) will appear starting Tuesday 2022-03-22. Final release is scheduled for Friday 2022-04-22.
- [R version 4.1.3 \(One Push-Up\)](#) has been released on 2022-03-10.
- [R version 4.0.5 \(Shake and Throw\)](#) was released on 2021-03-31.
- Thanks to the organisers of useR! 2020 for a successful online conference. Recorded tutorials and talks from the conference are available on the [R Consortium YouTube channel](#).
- You can support the R Foundation with a renewable subscription as a [supporting member](#)

### News via Twitter

# R ve R Studio'nun Kurulumu

## Turkey

<https://cran.pau.edu.tr/>  
<https://cran.gedik.edu.tr/>  
<https://cran.ncc.metu.edu.tr/>

## UK

<https://www.stats.bris.ac.uk/R/>  
<https://cran.ma.imperial.ac.uk/>

## USA

<https://mirror.las.iastate.edu/CRAN/>  
<http://ftp.usgs.ju.edu/CRAN/>  
<https://rweb.crdma.ku.edu/cran/>  
<https://repo.miserver.it.umich.edu/cran/>  
<http://cran.wustl.edu/>  
<https://archive.linux.duke.edu/cran/>  
<https://cran.case.edu/>  
<https://ftp.osuosl.org/pub/cran/>  
<http://lib.stat.cmu.edu/R/CRAN/>  
<https://cran.mirrors.hoobly.com/>  
<https://mirrors.nies.utk.edu/cran/>  
<https://cran.microsoft.com/>

Trimonav

## Pamukkale University, Denizli

Istanbul Gedik University

Middle East Technical University Northern Cyprus Campus, Mersin

University of Bristol

Imperial College London

Iowa State University, Ames, IA

Indiana University

University of Kansas, Lawrence, KS

MBNI, University of Michigan, Ann Arbor, MI

Washington University, St. Louis, MO

Duke University, Durham, NC

Case Western Reserve University, Cleveland, OH

Oregon State University

Statlib, Carnegie Mellon University, Pittsburgh, PA

Hoobly Classifieds, Pittsburgh, PA

National Institute for Computational Sciences, Oak Ridge, TN

Revolution Analytics, Dallas, TX

# R ve R Studio'nun Kurulumu

## The Comprehensive R Archive Network

### Download and Install R

Precompiled binary distributions of the base system and contributed packages. **Windows** and **Mac** users most likely want one of these versions of R:

- [Download R for Linux \(Debian, Fedora/Redhat, Ubuntu\)](#)
- [Download R for macOS](#)
- [Download R for Windows](#)

R is part of many Linux distributions, you should check with your Linux package management system in addition to the link above.

Source Code for all platforms

# R ve R Studio'nun Kurulumu

## R for Windows

Subdirectories:

[base](#)

Binaries for base distribution. This is what you want to [install R for the first time](#).

[contrib](#)

Binaries of contributed CRAN packages (for R >= 3.4.x).

[old\\_contrib](#)

Binaries of contributed CRAN packages for outdated versions of R (for R < 3.4.x).

[Rtools](#)

Tools to build R and R packages. This is what you want to build your own packages on Windows, or to build R itself.

Please do not submit binaries to CRAN. Package developers might want to contact Uwe Ligges directly in case of questions / suggestions related to Windows binaries.

You may also want to read the [R FAQ](#) and [R for Windows FAQ](#).

Note: CRAN does some checks on these binaries for viruses, but cannot give guarantees. Use the normal precautions with downloaded executables.

# R ve R Studio'nun Kurulumu

R-4.1.3 for Windows (32/64 bit)

[Download R 4.1.3 for Windows](#) (87 megabytes, 32/64 bit)

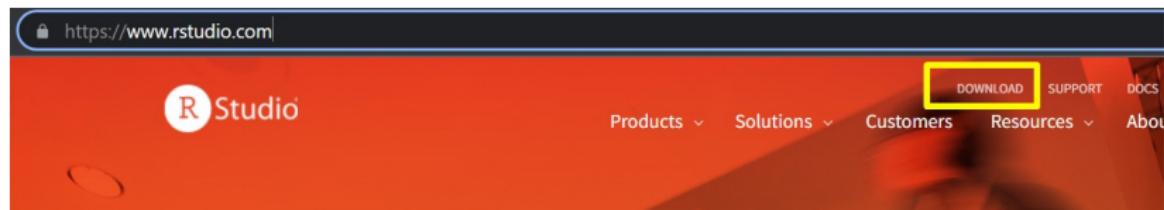
[Installation and other instructions](#)

[New features in this version](#)

If you want to double-check that the package you have downloaded matches the package distributed by CRAN, you can compare the [MD5](#) version of md5sum for windows: both [graphical](#) and [command line versions](#) are available.

[Frequently asked questions](#)

# R ve R Studio'nun Kurulumu



# R ve R Studio'nun Kurulumu

## RStudio Desktop

Open Source License

**Free**

**DOWNLOAD**

[Learn more](#)

## RStudio Desktop Pro

Commercial License

**\$995**

/year

**BUY**

[Learn more](#)

## RStudio Server

Open Source License

**Free**

**DOWNLOAD**

[Learn more](#)

## RStudio Workbench 1

Commercial License

**\$4,975**

/year

(5 Named Users)

**BUY**

[Evaluation](#) | [Learn more](#)

# R ve R Studio'nun Kurulumu

RStudio Desktop 2022.02.1+461 - [Release Notes ↗](#)

1. Install R. RStudio requires R 3.3.0+ ↗.
2. Download RStudio Desktop. Recommended for your system:

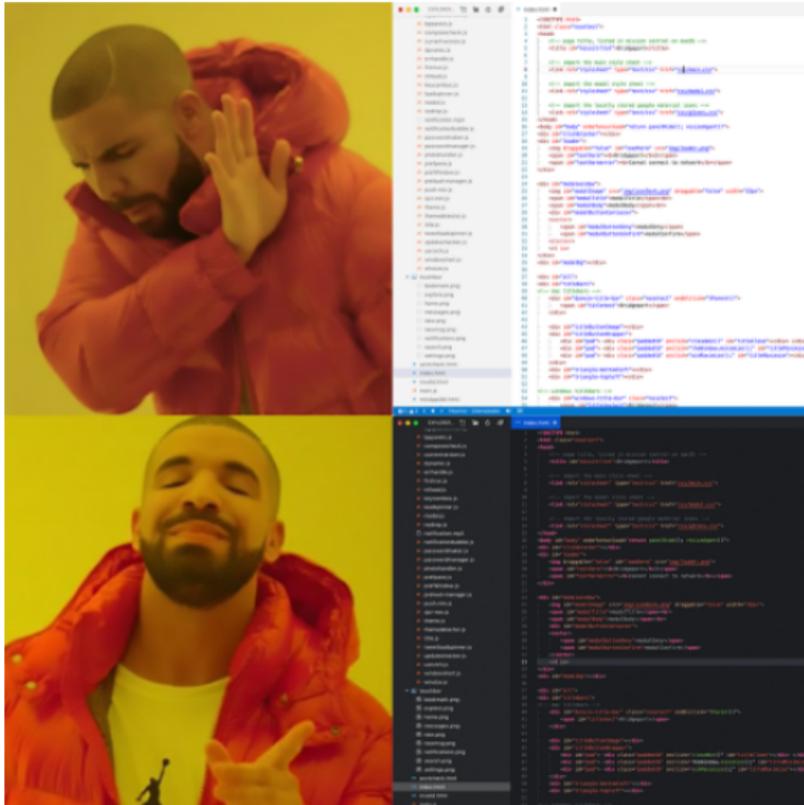


**DOWNLOAD RSTUDIO FOR WINDOWS**

2022.02.1+461 | 177.27MB

Requires Windows 10/11 (64-bit)

# Dark vs. Light Mode



# Interpreted vs Compiled Programlama Dilleri

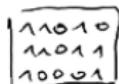
R, interpreted bir programlama dilidir.

Source code:  
hello.c



→ COMPILER →

Machine code:

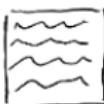


Program (also  
called binary,  
executable ...)

→ result  
run the  
program



Source code:  
hello.py



→ INTERPRETER → result



Figure 8: <https://medium.com/@skreweverything>

## İlk Komutlar, Atama, Obje

```
print("Hello, World!")
```

## İlk Komutlar, Atama, Objeler

-> ->>	rightwards assignment
<- <<-	assignment (right to left)
=	assignment (right to left)

Figure 9: <https://rdrr.io/r/base/Syntax.html>

# İlk Komutlar, Atama, Objeler



## snake\_case

Pros: Concise when it consists of a few words.  
Cons: Redundant as hell when it gets longer.  
`push_something_to_first_queue, pop_what, get_whatever...`



## PascalCase

Pros: Seems neat.  
`GetItem, SetItem, Convert, ...`  
Cons: Barely used. (why?)



## camelCase

Pros: Widely used in the programmer community.  
Cons: Looks ugly when a few methods are n-worded.  
`push, reserve, beginBuilding, ...`

# Operatörler

- ▶ Atama: <- = «- -> -»
- ▶ Aritmetik: + - \* / %% %/ % ^ \*\*
- ▶ İlişkisel: > < == <= >= !=
- ▶ Mantıksal: & | ! && ||
- ▶ Diğer: : %in% %\*%

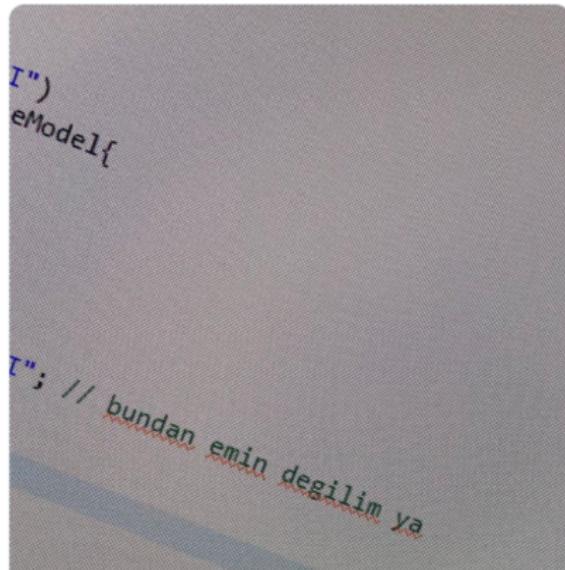
# #Yorum Satırı



null  
@muztsuz

...

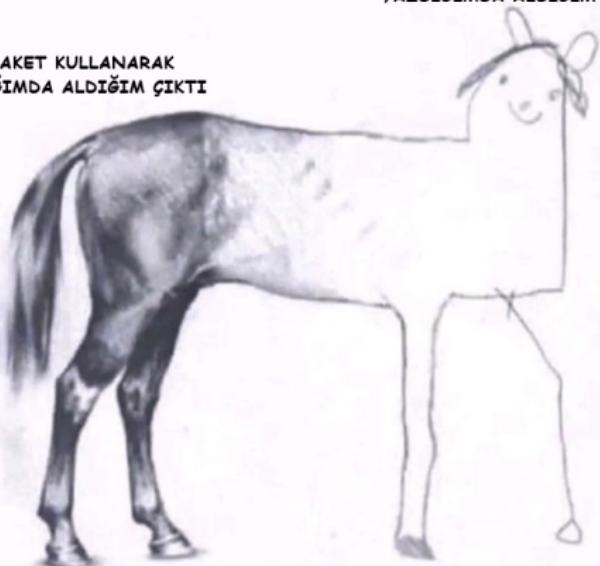
Boyle commitlemisim.. ben en az 5 yil daha jr kalicam  
gibi



# Paketler

KODU PAKET KULLANMADAN  
YAZDIĞIMDA ALDIĞIM ÇIKTI

KODU PAKET KULLANARAK  
YAZDIĞIMDA ALDIĞIM ÇIKTI



## “Basit” Hatalar



## “Basit” Hatalar

- ▶ Yanlış fonksiyon ismi (Ör: sum()/sun())
- ▶ Objenin yanlış kullanımı (Ör: veri/verı)
- ▶ Noktalama hatası (Ör: , yerine . kullanmak)
- ▶ Yerleşimi yanlış yapmak (Ör: ,'den sonra eleman getirmemek)

# Script



## Veri Tipleri (Numeric, Integer, Character, Logical)

## Veri Tipleri (Numeric)

```
num1 <- 2022  
num2 <- 3.14
```

```
class(num1)  
class(num2)
```

```
typeof(num1)  
typeof(num2)
```

## Veri Tipleri (Integer)

```
int1 <- as.integer(num2) #L: Long

class(int1)
typeof(int1)

is.integer(2)
is.integer(2L)

typeof(1L + 2L)
typeof(1L / 2L)
```

## Veri Tipleri (Integer)

#Her Integer Numeric'tir; her Numeric Integer değildir.

```
class(1)  
is.numeric(1)  
is.integer(1)
```

```
class(1.0)  
is.numeric(1.0)  
is.integer(1.0)
```

```
class(1L)  
is.numeric(1L)  
is.integer(1L)
```

## Veri Tipleri (Character)

```
char1 <- "R"  
char2 <- '2.718'  
  
class(char1)  
class(char2)
```

## Veri Tipleri (Logical/Boolean)

```
as.numeric(TRUE)  
as.numeric(FALSE)
```

```
TRUE + FALSE
```

## Veri Tipleri (Logical/Boolean)

```
1 == 0  
1 != 0  
!(1 == 0)
```

```
1 > 0  
1 <= 0
```

```
TRUE | FALSE  
TRUE & FALSE
```

## Veri Tipleri (Logical/Boolean)

```
2.718 == '2.718' #character'e zorlar
```

```
6 == 3 | 9 # logical'a zorlar
```

```
6 == 3 & 9 # logical'a zorlar
```

# Veri Yapıları (Scalar&Vector, Matrix, Array, List, Factor, Data Frame)

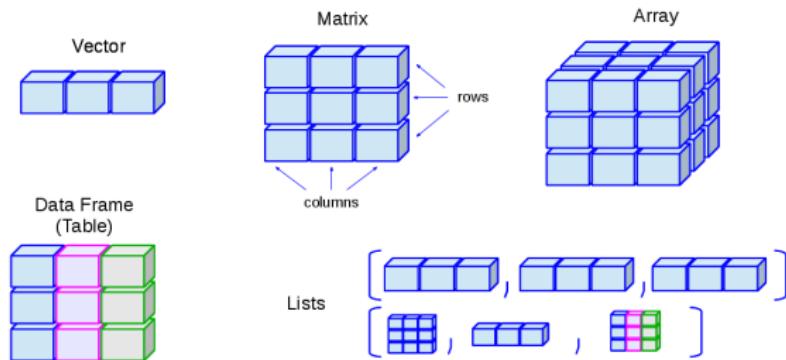


Figure 10: <http://venus.ifca.unican.es/Rintro/dataStruct.html>

## Veri Yapıları (Scalar&Vector)

```
skaler1 <- "R"  
skaler2 <- 2022  
  
vektor1 <- c("R",2022)  
class(vektor1) #character'e zorlar  
as.numeric(vektor1) #numeric'e zorlar  
  
vector(mode = "numeric", length = 5)
```

## Veri Yapıları (Scalar&Vector)

```
vektor2 <- c(34,10,22,48)  
  
vektor2[2]  
vektor2[c(1,3)]  
vektor2[-4]  
vektor2[5] <- 100
```

## Veri Yapıları (Scalar&Vector)

```
vektor3 <- c("Dil1"="R Programlama Dili",
            "Dil2"="Python Programlama Dili")
length(vektor3)
nchar(vektor3)

vektor4 <- c(7,0,11,50,-30)
sort(vektor4)
sort(vektor4, decreasing = TRUE)
```

## Veri Yapıları (Matrix)

```
matris <- matrix(data = 0, nrow = 5, ncol = 3)
dim(matris)

m_vektor <- 1:15
matris2 <- matrix(data = m_vektor,
                    nrow = 3,
                    ncol = 5,
                    byrow = TRUE)
```

## Veri Yapıları (Matrix)

```
matris2[2,]  
matris2[,2]
```

```
matris2[c(1,2),]  
matris2[,c(1,2)]
```

```
matris2[-1,-c(2,3)]
```

```
5 %in% matris2  
20 %in% matris2
```

```
m1 <- matrix(data = c(1,3,5,7), ncol = 2)  
m2 <- matrix(data = c(2,4,6,8), ncol = 2)
```

```
m1 %*% m2
```

## Veri Yapıları (Array)

```
vektor1 <- sample(1:100,10)
vektor2 <- sample(1:100,10)

dizi <- array(c(vektor1,vektor2), dim = c(2,5,2))
#dim: satır,sütun,matrİs

length(dizi)

dizi[2,4,1]
dizi[, , 2]

43 %in% dizi
70 %in% dizi
```

## Veri Yapıları (Array)

```
liste <- list(  
  
  c("a","b","c"),  
  c(1,2,3,4,5),  
  c(TRUE,FALSE,TRUE),  
  matrix(data = 1:15, nrow = 5, ncol = 3)  
  
)  
  
str(liste)
```

## Veri Yapıları (Array)

```
liste[3]
liste[[3]]

liste[[2]] <- c(2,3,5,7)

length(liste)

"c" %in% liste
"c" %in% liste[1]
"c" %in% liste[[1]]

append(liste, list(c("x","y","z")), after = 1)
```

## Veri Yapıları (Factor)

```
faktor <- factor(c("Lisans", "Yuksek Lisans", "Doktora",
                    "Yuksek Lisans", "Doktora", "Doktora",
                    "Doktora", "Lisans", "Yuksek Lisans"))
faktor2 <- factor(faktor,
                   levels = c("Lisans",
                             "Yuksek Lisans",
                             "Doktora"))
faktor2[2]
```

## Veri Yapıları (Data Frame)

```
df <- data.frame(  
  
    sutun1 = seq_len(12),  
    sutun2 = LETTERS[1:12],  
    sutun3 = rep(c(TRUE,FALSE), each = 6),  
    sutun4 = seq(as.Date("2022-04-04"),  
                as.Date("2022-04-15"),  
                "days"))  
  
)  
  
str(df)  
dim(df)  
ncol(df)  
nrow(df)  
length(df)
```

## Veri Yapıları (Data Frame)

```
df$sutun1
df[[1]]
df[["sutun1"]]

df[12,4]

names(df)
names(df)[4] <- "tarih"

df$sutun5 <- sample(1:100,12)
df <- cbind(df, sutun6 = sample(1:100,12))

df$sutun6 <- NULL
```

## Veri Yapıları (Data Frame)

```
df2 <- data.frame(  
  sutun1 = 22L,  
  sutun2 = "H",  
  sutun3 = TRUE,  
  tarih = as.Date("2022-04-16"),  
  sutun5 = 100L  
)  
str(df2)  
  
df3 <- rbind(df,df2)  
str(df3)
```

## NULL, NaN, NA

- ▶ NULL: Sonuç tanımlanamıyor
- ▶ NaN: Not a Number/sayısal olmayan
- ▶ NA: Not Available/kayıp

## Fonksiyonlar (Base, Oluşturma, Üç Nokta)

## Fonksiyonlar (Base)

- ▶ `cat()`
- ▶ `grep()`
- ▶ `seq()`, `seq_along()`, `seq_len()`
- ▶ `rep()`
- ▶ `readline()`
- ▶ `paste()`, `paste0()`
- ▶ `tolower()`, `toupper()`
- ▶ `round()`, `ceiling()`, `floor()`
- ▶ `max()`, `min()`, `mean()`, `sd()`, `sqrt()`, `log()`, `exp()`

## Fonksiyonlar (Oluşturma)

```
fonksiyon_ismi <- function(parametre){  
  aksiyon  
  return(cikti)  
}
```

## Fonksiyonlar (Oluşturma)

```
set.seed(1)
random_sayi <- sample(1:100,20)

mean(random_sayi)

ortalama <- function(sayi){

    toplam <- sum(sayi)
    elemanSayisi <- length(sayi)
    sonuc <- toplam / elemanSayisi

    return(sonuc)

}

ortalama(sayı = random_sayi)
```

## Fonksiyonlar (Üç Nokta)

```
# Parametre girmek opsiyoneldir.  
# Parametre sayısı çok fazla olabilir.  
  
topla <- function(sayi1, sayi2){  
  
    return(  
        sum(sayi1,sayi2)  
    )  
  
}  
  
topla(3,5)  
topla(3,5,8)
```

## Fonksiyonlar (Üç Nokta)

```
# Parametre girmek opsiyoneldir.  
# Parametre sayısı çok fazla olabilir.  
  
topla <- function(...){  
  return(  
    sum(...)  
  )  
}  
  
topla(3,5)  
topla(3,5,8)
```

## Fonksiyonlar (Üç Nokta)

# Parametre girmek opsiyoneldir.  
# Parametre sayısı çok fazla olabilir.

```
set.seed(1)
random_sayi <- c(sample(1:100,20),NA)

ortalama <- function(sayi){

    toplam <- sum(sayi)
    elemanSayisi <- length(sayi)
    sonuc <- toplam / elemanSayisi

    return(sonuc)

}

ortalama(sayi = random_sayi, na.rm = TRUE)
```

## Fonksiyonlar (Üç Nokta)

# Parametre girmek opsiyoneldir.  
# Parametre sayısı çok fazla olabilir.

```
set.seed(1)
random_sayi <- c(sample(1:100,20),NA)

ortalama <- function(sayı,...){

    toplam <- sum(sayı,...)
    elemanSayisi <- length(sayı)
    sonuc <- toplam / elemanSayisi

    return(sonuc)

}

ortalama(sayı = random_sayi, na.rm = TRUE)
```

## If - Else If - Else Koşul İfadeleri

Eğer çalışma bittiysse kahve yap (if)

Eğer çalışma bittiysse kahve yap; bitmediyse devam et (if else)

Eğer çalışma bittiysse kahve yap; bitmediyse ama yorulduysan hava al; yoksa devam et (if-else if-else)

```
if(kosul1){  
    aksiyon1  
} else if(kosul2){  
    aksiyon2  
} else {  
    aksiyon3  
}
```

Girilen sayının tek mi çift mi olduğunu söyleyen bir pratik

## If - Else If - Else Koşul İfadeleri

```
ortalama <- function(sayi) {  
  if (is.numeric(sayi)) {  
    toplam <- sum(sayi)  
    elemanSayisi <- length(sayi)  
    sonuc <- toplam / elemanSayisi  
  
    return(sonuc)  
  } else {  
    message("Numeric olmayan bir eleman girişi!")  
  }  
  
}  
  
ortalama(sayi = c(1, 2, 3, 4, "5"))
```

## For Döngüsü

```
for(i in 1:10){  
    print(i)  
}  
  
for(i in seq_len(10)){  
    print(i)  
}  
  
iller <- c("istanbul","ankara","izmir")  
for(i in seq_along(iller)){  
    print(iller[i])  
}  
  
for(i in 1:7){  
    for(j in 1:4){  
        print(paste0("i: ",i," j: ",j))  
    }  
}
```

## For Döngüsü

```
# next, break

harfler <- c("A", "B", "C", "Ç", "D", "E", "F", "G", "Ğ", "H")

for(harf in harfler){
  if(harf %in% c("Ç", "Ğ")){
    next
  }
  print(harf)
}

for(harf in harfler){
  if(harf %in% c("Ç", "Ğ")){
    break
  }
  print(harf)
}
```

# While Döngüsü

```
x <- 0

while(x < 10){
    print(x)
} #sonsuz

while(x < 10){
    print(x)
    x = x + 1
}
```

## Paket İncelemesi (TCMB Örneği)

```
anahtar <- function(api_anahtar){  
  Sys.setenv(myKey = api_anahtar)  
  anahtarim <- Sys.getenv("myKey")  
}
```

Figure 11: <https://github.com/rpydaneogrendim/TCMB>

# Paket İncelemesi (TCMB Örneği)

```
yardim <- function(){
  print("https://evds2.tcmb.gov.tr/index.php?evds/login adresine gidin.", quote = F)
  print("Uye ekraninda giris yaptiktan sonra Kullanici ismine tiklayip gelen menudeki Profil secenegine tıklayın.", quote = F)
  print("Gelen ekrandan API Anahtari dugmesine tiklayarak ihtiyacınız olan degere ulasabilirsiniz.", quote = F)
  print("Paketi cagirdiktan sonra anahtar() fonksiyonu ile anahtarınızı tanitin.", quote = F)
}
```

Figure 12: <https://github.com/rpydaneogrendim/TCMB>

# Paket İncelemesi (TCMB Örneği)

```
parametreler <- function(){

    gozlem <- data.frame(
        "Tip" = c("Ortalama",
                  "En Dusuk",
                  "En Yuksek",
                  "Baslangic",
                  "Bitis",
                  "Kumulatif"),
        "Kisa_Kod" = c("avg",
                      "min",
                      "max",
                      "first",
                      "last",
                      "sum")
    )

    formul <- data.frame(
```

Figure 13: <https://github.com/rpydaneogrendim/TCMB>

# Paket İncelemesi (TCMB Örneği)

```
kategoriler <- function(){

  if(exists("anahtarim")){
    url <- paste0("https://evds2.tcmb.gov.tr/service/evds/categories?key=",anahtarim,"&type=json")
    kategori <- fromJSON(url) %>%
      as.data.frame() %>%
      select(CATEGORY_ID, TOPIC_TITLE_TR) %>%
      rename("Kategori_Kodu" = "CATEGORY_ID", "Konu_Basligi"="TOPIC_TITLE_TR")
  } else {
    stop("Lutfen TCMB'den API anahtarınızı alın. Detaylı bilgi için yardım() fonksiyonunu çalıştırın.")
  }
}
```

Figure 14: <https://github.com/rpydaneogrendim/TCMB>

Katılımınız ve sabrınız için teşekkür ederim :)

- ▶ Sorularınız için:

*Twitter: @rpydaneogrendim*

*Mail: rpydaneogrendim@gmail.com*

- ▶ Sunum ve diğer kod paylaşımılarına ulaşmak için:

*Github: <https://github.com/rpydaneogrendim>*