

# R Temelleri

## R Paketleri

- Comprehensive R Archive Network (CRAN)’den R indirildiğinde “base” (temel) R sistemi indirilir.
- Temel R sistemi, R dilini kullanmak için gerekli olan temel fonksiyonları içermektedir.
- R’ın kullanışlı olmasının bir nedeni, büyük bir R paketi koleksiyonu ile temel fonksiyonlarının geliştirilebilmesidir.
- R paketleri, R kullanıcıları tarafından yazılmakta ve yayınlanmaktadır.

## R Paketlerini İndirme (Download)

- R paketleri için temel kaynak [CRAN](#)’dır.
- Biyolojik uygulamalar için [Bioconductor Project](#) üzerinde birçok paket bulunmaktadır.
- CRAN üzerinde yer alan paketler hakkında bilgi almak için `available.packages()` komutunu kullanabilirsiniz.
- CRAN üzerinde yaklaşık olarak 19000 farklı paket bulunmaktadır.
- CRAN üzerinde ayrıca [Task Views](#) adı altında belirli konular ile ilişkili paketlerin listesi yer almaktadır.
- [Github](#), [Bitbucket](#) ve [Gitlab](#) benzeri sitelerde de kullanıcılar yazdıkları paketleri paylaşmaktadırlar. Bu sitelerde paylaşılan paketleri kurmak için `devtools` paketi sisteminizde kurulu olmalıdır.

## R Paketlerini Kurma (Install)

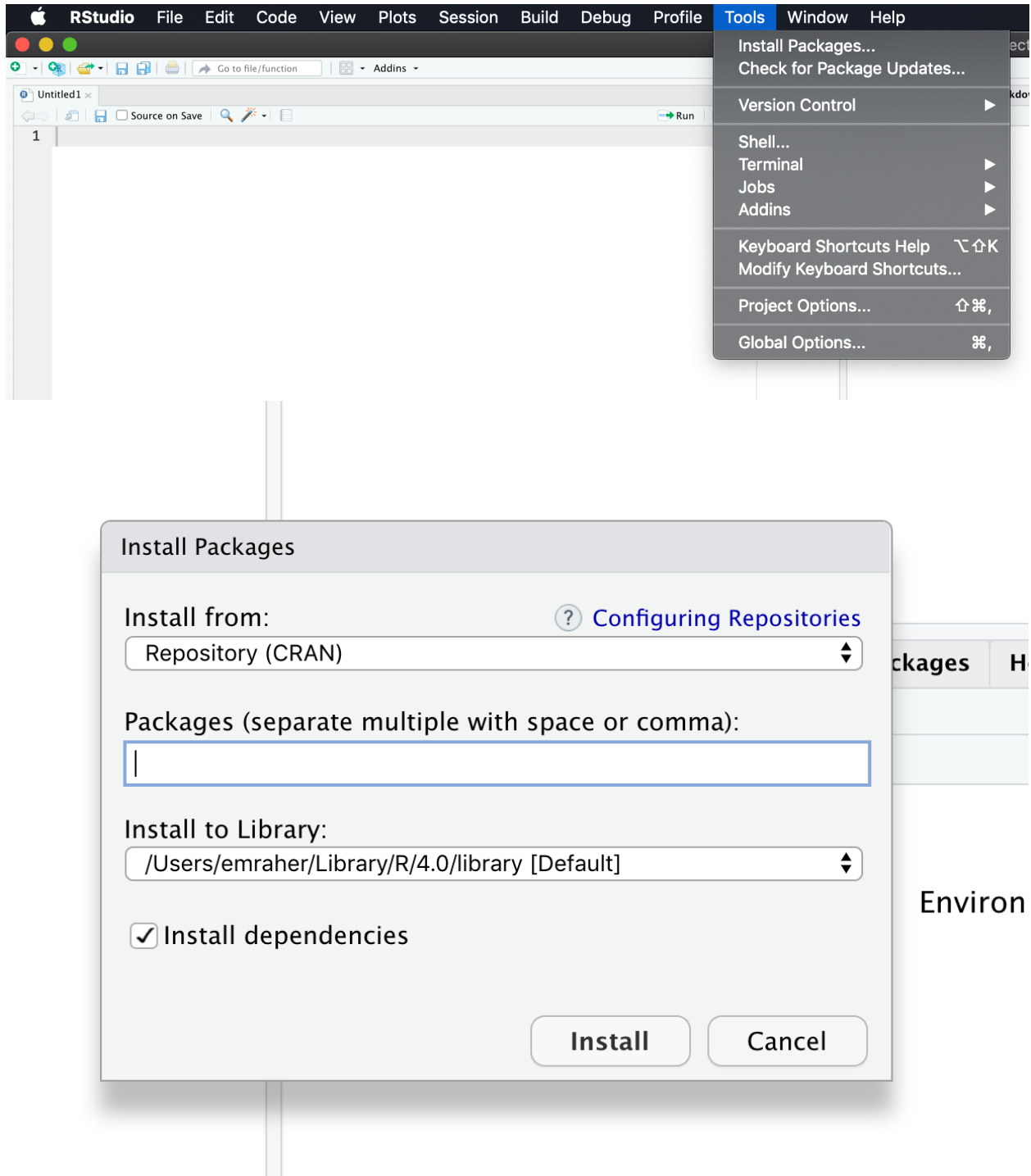
- Paketler `install.packages()` fonksiyonu ile kurulabilir.
- Bir paketi kurmak için paket adı `install.packages()` fonksiyonuna ilk argüman olarak girilmelidir.
- Örneğin aşağıda yer alan kod CRAN üzerinden **tidyverse** paketini kurmak için kullanılır.

```
install.packages("tidyverse")
```

- Bu komut **tidyverse** paketini CRAN üzerinden indirir ve bilgisayarınıza kurar.
- Paketin bağlı olduğu diğer paketler de indirilir ve kurulur.
- Birden fazla paket yüklemek için ise aşağıda yer alan komut benzeri bir komut kullanılabilir.

```
install.packages(c("slidify", "ggplot2", "devtools"))
```

## RStudio'da Paket Kurma



## R Paketlerini Yükleme (Load)

- R paketlerini kurmak o paketleri R'da hemen kullanılır hale getirmemektedir. Paketleri kullanabilmek için paketler R'a yüklenmelidir.
- `library()` fonksiyonu paketleri R'a yüklemek için kullanılır.

- **ggplot2** paketini R'a yüklemek için aşağıdaki komut kullanılır.

```
library(ggplot2)
```

- Paketin bağımlı olduğu paketler yüklendikten sonra paketin yüklenmesi gerçekleşir.
- NOT: Paket ismi tırnak içerisinde konulmamalıdır!
- Bazı paketler yükledikten sonra konsolda mesaj gösterebilirler. Bunu önlemek için aşağıdaki komut kullanılabilir.

```
suppressPackageStartupMessages(library(tidyverse))
```

- Paketler R'a yüklendikten sonra paket içerisindeki fonksiyonlar kullanılabilir hale gelir.
- [require vs. library](#)

## Github Üzerinden Paket Kurma

```
install.packages("devtools")  
library(devtools)  
install_github("emraher/cbRt")
```

## İlk R betiği

Kurulumu başarıyla gerçekleştirdiyseniz RStudio programını açın.

Menülerden **File (Dosya) > New (Yeni) > R Script (R Betiği)** seçeneğini seçin.

Bu size boş bir yazı belgesi açacaktır.

Bu belgeye aşağıda yer alan komutları yazın.

Metni fare ile seçtikten sonra **Run (Çalıştır)** tuşuna basın.

Eğer herşey sorunsuz çalışıyorsa, konsolda **TRUE** yazısını göreceksiniz.

```
x <- 5  
x == 5
```

### Tebrikler! İlk R betiğinizi (script) yazdınız!

Dosyayı kaydetmek için menüden **File (Dosya) > Save As (Farklı Kaydet)** seçeneğini seçin ve istediğiniz bir dosya ismi verin.

Dosyalarınızın uzantısının **.R** olduğuna dikkat edin.

**.R** uzantılı dosyaları RStudio ile açabilirsiniz.

Betiğinizde yer alan komutları satır satır da çalıştırabilirsiniz.

İmlecini satır sonuna getirin ve **Ctrl + Enter** tuşlarına aynı anda basın.

Diğer önemli bir kısayol da tüm metni seçen **Ctrl + A** kısayoludur.

Windows işleminden aşına olduğunuz bazı kısayolları (**Ctrl + C**, **Ctrl + V**, vs.) kullanabilirsiniz.

RStudio'da kullanılabilecek tüm kısayolları görmek için ise kısayolların kısayolu **Alt + Shift + K** kombinasyonunu kullanabilirsiniz.

## Aritmetik

```
# Toplama  
1 + 1
```

```
## [1] 2
```

```
# Çıkarma  
8 - 4
```

```
## [1] 4
```

```
# Bölme  
13/2
```

```
## [1] 6.5
```

```
# Çarpma  
4 * pi # pi, pi sayısını göstermektedir
```

```
## [1] 12.56637
```

```
# Üstel  
2^10
```

```
## [1] 1024
```

## Mantıksal Karşılaştırmalar

```
3 < 4
```

```
## [1] TRUE
```

```
3 > 4
```

```
## [1] FALSE
```

```
# 3 = 4 karşılaştırması  
3 == 4
```

```
## [1] FALSE
```

```
# != "eşit değildir" anlamına gelmektedir  
3 != 4
```

```
## [1] TRUE
```

```
4 >= 5
```

```
## [1] FALSE
```

```
4 <= 5
```

```
## [1] TRUE
```

```
2 + 2 == 5
```

```
## [1] FALSE
```

```
10 - 6 == 4
```

```
## [1] TRUE
```

## Metin (yazı)

Sayılar bilgisayarların temelidirler fakat biz ölümlüler için anlamayı olanaklı kılan metinlerdir.

```
'İstatistik harika'
```

```
## [1] "İstatistik harika"
```

```
# R'da metinler tek veya çift tırnak içerisinde yazılır.  
"R'da istatistik daha da harika"
```

```
## [1] "R'da istatistik daha da harika"
```

## Değişkenler

Cebirde olduğu gibi R'da da değişkenler çok iyi kısaltmalardır.

Her seferinde 3.1415926... yazmak yerine sadece `pi` yazmak mümkündür.

---

Bir değişkene değer atama sağdan sola gerçekleştirilir.

Sağda yer alan değer solda yer alan *isim*'e atanır.

R'da bazı istisnalar hariç herşey değişken ismi olarak kullanılabilir.

Bu kurallar şunlardır:

1. `.` ve `_` işaretleri kullanılabilir, bunlar haricindeki diğer semboller kullanılamaz.
2. Değişken isimleri sayı veya `_` sembolüyle başlayamaz (`2kare` ve `_bir` illegal değişken isimleridir).

---

R betikleri yazılırken bazı stil kurallarına uymakta fayda vardır.

Bu kurallar hem sizin hem de yazdığınız betiği okuyanların işini kolaylaştırır.

Yazdığınız kodların Türkçe karakter içermemesi tavsiye edilir.

Betik yazılırken en önemli kural tutarlı olmaktır.

R'da kod yazma stili konusunda ayrıntılı bilgiye [The tidyverse style guide](#)'dan ulaşabilirsiniz.

```
x <- 42
x / 2
```

```
## [1] 21
```

```
# Eğer x'e farklı bir değer atarsak,
# önceki atanmış değer kaybolur.
x <- "Angara'nın Bağları"
x
```

```
## [1] "Angara'nın Bağları"
```

```
x <- 5
x == 5
```

```
## [1] TRUE
```

```
foo <- 3 # Bakınız: https://eksisozluk.com/foobar
bar <- 5
foo.bar <- foo + bar
foo.bar
```

```
## [1] 8
```

```
foo.bar2 <- 2 * foo.bar
foo.bar2
```

```
## [1] 16
```

```
foo_bar <- foo - bar
foo_bar
```

```
## [1] -2
```

R'da <- işareti “atama işleci” olarak adlandırılır.

İnternette karşınıza x = 42 şeklinde yazılmış R kodları çıkabilir.

= ve <- birbirlerine çok benzer olmalarına rağmen aralarında ufak farklılıklar vardır.

R kodu yazarken “atama işleci” olarak biz <- işaretini kullanacağız.

Bu işareti RStudio'da **Alt + -** tuşlarına basarak yazabilirsiniz.