

Praktikum Eksplorasi dan Visualisasi Data

Pertemuan 1

Pengenalan R

1. Instalasi dan Pengenalan R

Instalasi R-Gui

Download di website: <https://cran.r-project.org/>

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.

Tekan/ *Click* link didalam kotak merah

*Download sesuai *Operating System* dari Laptop / *Personal Computer*

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 \geq 2.13.x; managed by Uwe Ligges). There is also information on third party software available for CRAN Windows services and corresponding environment and make variables.
old contrib	Binaries of contributed CRAN packages for outdated versions of R (for R $<$ 2.13.x; managed by Uwe Ligges).
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.

Untuk *Operating System* Windows *Click link* di dalam kotak merah

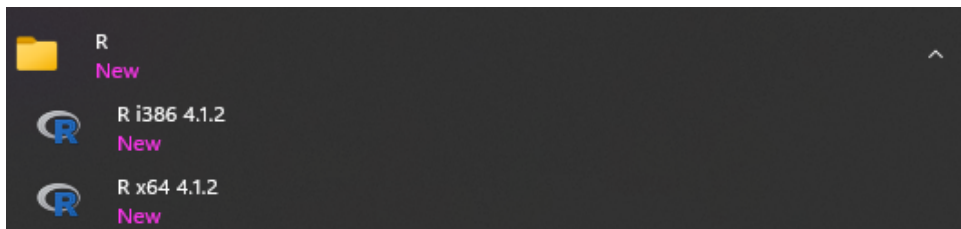
R-4.1.2 for Windows (32/64 bit)

[Download R 4.1.2 for Windows](#) (86 megabytes, 32/64 bit)
[Installation and other instructions](#)
[New features in this version](#)

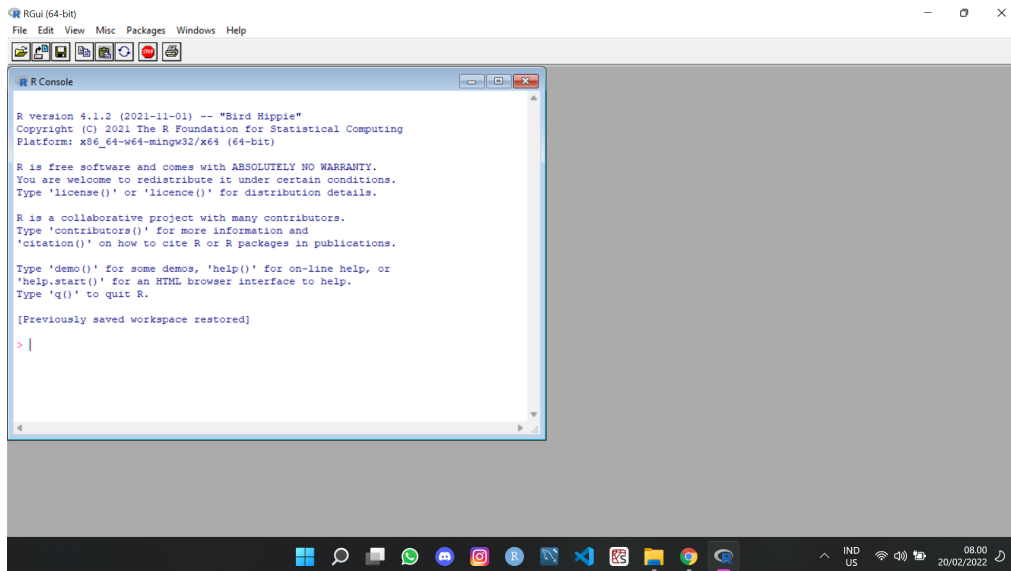
Lalu download dan lakukan instalasi seperti menginstal *software* lainnya

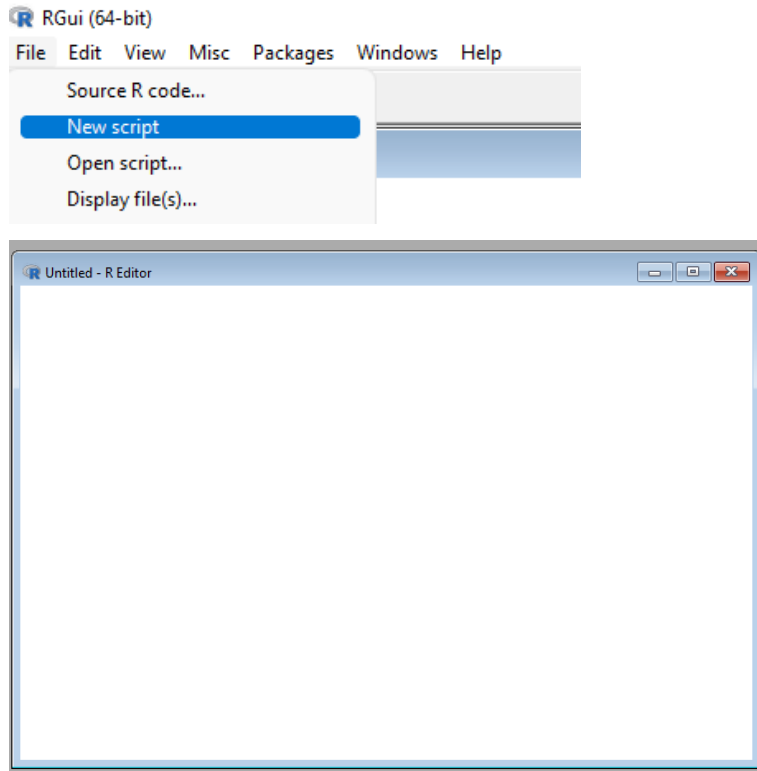
Membuka R-GUI

Membuka R-Gui dapat dilakukan melalui shortcut yang sudah ada pada layar desktop, atau dengan *searching* R di *Start Windows*.

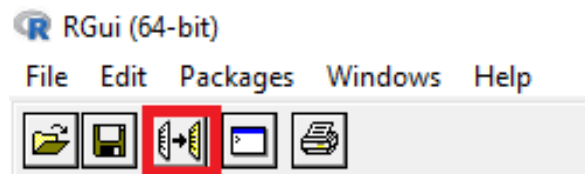


Tampilan awal R-Gui





Pada R Script, kita dapat menuliskan terlebih dahulu seluruh *syntax* sampai selesai. Untuk menjalankan *syntax* dapat dengan `ctrl + R` atau klik pada icon dalam kotak merah. Syntax dapat dijalankan per baris atau bisa keseluruhan dengan mengeblok seluruh syntax. Outputnya akan muncul di R-Console.



2. Data dan Tipe Data

Tidak seperti C atau Java yang perlu mendeklarasikan variabel dengan tipe data tertentu pada awal sintaks, pada R Language variabel tidak perlu dideklarasikan dengan tipe data tertentu dan dapat diubah setelah dideklarasikan. untuk memeriksa tipe data dapat menggunakan fungsi `class()`. Ada beberapa tipe data dalam R. diantaranya yaitu :

Tipe data	Contoh	Keterangan
Logical	TRUE, FALSE	Nilai Boolean
Numeric	12.3, 5, 99999, 125.08	Segala Jenis Angka
Integer	23L, 97L, 3L	Bilangan Integer

Complex	$2i, 1 + 2i, 2 + 8i$	Bilangan Kompleks
Character	"135", "c", "1", "hello	Karakter/String
Raw	"Hello world"	Segala jenis data yang disimpan sebagai raw bytes <pre>v <- charToRaw("Hello world") print(class(v))</pre> Output : <pre>[1] "raw"</pre>

3. Nama Variabel

Nama dapat terdiri dari kombinasi dari huruf besar, huruf kecil, angka, underscore, dan titik, akan tetapi kita tidak dapat menggunakan Extended ASCII character atau karakter spesial seperti `$%#*` dan lainnya. Nama hanya dapat diawali dengan huruf dan titik, jika diawali dengan titik tidak dapat dilanjutkan dengan angka. Berikut contoh dari nama objek yang valid :

mydata

my.data

my.data.2

Data.test.no.2

Berikut contoh nama objek yang tidak valid :

4data (dimulai dari angka)

data-1 (operator '-' tidak dapat digunakan)

Selain itu kita juga harus menghindari penggunaan nama fungsi/command built-in R (contoh: `mean()`, `median()`, `as.data.frame()`). Apabila kita menggunakan nama fungsi/data built-in maka kita tidak dapat mengakses fungsi dan data built-in ini sampai kita menghapus nama objek yang dibuat. Perlu diperhatikan pula bahwa nama di R *case sensitive*.

4. Variabel di R

Variabel di R dapat berisikan hanya 1 nilai, dapat juga berisikan lebih dari 1 nilai berupa vektor.

Variabel dengan 1 nilai

Variabel dengan 1 nilai dapat dituliskan dengan “ = ” atau “ <- ”

```
> dataku=5
```

```
> dataku
```

```
[1] 5
```

```
> dataku <- 6
```

```
> dataku
```

```
[1] 6
```

Variabel berisikan vektor (data array satu dimensi)

Vektor merupakan struktur data paling dasar yang dikenal dalam R, yang merupakan suatu array/himpunan dari bilangan, character/string, logical value, dll. Yang perlu diingat kita harus menggunakan satu tipe data yang sama untuk data kita, yakni kita tidak dapat menggabungkan dua data atau lebih yang berbeda tipe ke dalam satu objek vektor. Jika ini dilakukan, maka R akan mengubah data ke mode yang terumum.

```
> c(1,2,F) #vektor terdiri dari data dengan mode logical  
dan numeric
```

```
[1] 1 2 0
```

```
> c("X",T)#vektor terdiri dari data dengan mode character  
dan logical
```

```
[1] "X"      "TRUE"
```

```
> c("X",2,F)#vektor terdiri dari data dengan mode  
character, numeric dan logical
```

```
[1] "X"      "2"      "FALSE"
```

```
> x = c(2,3) #x berisikan 2 dan 3
```

```
> print(x)
```

```
[1] 2 3
```

*tanda pagar “#” merupakan komentar yang berarti kata - kata setelah tanda # tidak akan dieksekusi R

5. Ekstraksi sebagian Data Vector

Misalkan kita memiliki sebuah vektor yang terdiri dari 6 elemen, yaitu :

```
> y= c(3,9,15,21,6,5)
```

```
> y
```

```
[1] 3 9 15 21 6 5
```

Kita dapat membentuk data baru dengan melakukan ekstraksi data. Sebagai contoh :

- Menampilkan elemen pertama

```
> y[1]  
[1] 3
```
- Menampilkan elemen pertama dan ketiga

```
> y[c(1,3)]  
[1] 3 15
```
- Menampilkan semua elemen kecuali elemen ketiga

```
> y[-3]  
[1] 3 9 21 6 5
```
- Menampilkan semua elemen kecuali elemen pertama dan ketiga

```
> y[-c(1,3)]  
[1] 9 21 6 5
```

6. Operator Operasi di R

Terdapat 3 macam operator operasi di R, yakni aritmatika, perbandingan, dan logika, *syntax* dan penjelasan lebih lanjutnya adalah sebagai berikut :

Operator Aritmatika

Simbol	Keterangan
+	<i>Addition</i> - operasi penjumlahan
-	<i>Subtraction</i> - operasi pengurangan
*	<i>Multiplication</i> - operasi perkalian
/	<i>Division</i> - operasi pembagian
^	<i>Exponentiation</i> - operasi pemangkatan
%%	<i>Modulus</i> - mencari sisa pembagi
%/ %	<i>Integer</i> - mencari bilangan bulat hasil pembagian saja

Operator Perbandingan

Simbol	Keterangan
==	sama dengan, bernilai TRUE jika kedua objek bernilai sama

!=	tidak sama dengan
>	lebih besar dari
<	lebih kecil dari
>=	lebih besar sama dengan
<=	lebih kecil sama dengan

Pada operator perbandingan, akan bernilai TRUE jika hasilnya benar dan bernilai FALSE jika hasilnya salah.

Operator Logika

Simbol	Keterangan
& &	operator logika AND
	operator logika OR
!	operator logika NOT
&	operator logika AND element wise
	operator logika OR element wise

Matriks dan Data

Frame

Matriks	Data Frame
Setiap kolom atau variabel harus terdiri dari tipe data yang sama, yakni numerik.	Setiap kolom atau variabel tidak harus terdiri dari tipe data yang sama, dapat memiliki tipe data yang berbeda seperti numerik, karakter, dll.
Dapat digunakan untuk melakukan segala jenis operasi aljabar linear.	Tidak dapat digunakan untuk melakukan segala jenis operasi aljabar linear.
Kurang nyaman digunakan jika sering merujuk ke sebuah kolom dengan nama.	Lebih nyaman digunakan jika sering merujuk ke sebuah kolom dengan nama.
Lebih hemat memori.	Kurang hemat memori.

Matriks

Pembentukan Matriks

```
> matriks1<-matrix(c(1,2,3,4,5,6,7,8,9),nrow=3, ncol=3)
```

```
> matriks1
```

```
      [,1] [,2] [,3]
[1,]    1    4    7
[2,]    2    5    8
[3,]    3    6    9
```

```
> matriks2<-matrix(1:9,3)
```

```
> matriks2
```

```
      [,1] [,2] [,3]
[1,]    1    4    7
[2,]    2    5    8
[3,]    3    6    9
```


Secara default, data akan diisikan kolom perkolom, seperti yang dapat dilihat sebagai berikut :

```
> data=c(2,3,4,5,6,7)
> matriks3<-matrix(data=data,nrow=3,ncol=2)
> matriks3
      [,1] [,2]
[1,]    2    5
[2,]    3    6
[3,]    4    7
```

Untuk melakukan pengisian menurut baris perbaris, maka dapat digunakan optional argumen **byrow=T atau byrow=TRUE** pada command matriks. Lihat contoh berikut :

```
> matriks3<-matrix(data=data,nrow=3,ncol=2, byrow=TRUE)
> matriks3
      [,1] [,2]
[1,]    2    3
[2,]    4    5
[3,]    6    7
```

Operator untuk operasi matriks

Operator Matriks	Keterangan
*	Perkalian elemen demi elemen matriks
%*%	Perkalian matriks
%o%	Outer
solve	Invers matriks
t	Transpose matriks
crossprod	Cross product yakni $X^T X$, untuk X suatu matriks

Menggabungkan satu kolom baru atau satu baris baru ke matriks lain

Dalam R bila kita ingin menggabungkan satu kolom atau satu baris baru ke dalam satu objek matriks lain dapat menggunakan perintah **rbind** (untuk menambahkan baris) dan **cbind** (untuk menambahkan kolom). Sebagai contoh :

```
> mat<-matrix(c(4,7,8,6,4,4), nrow=2)
```

```
> mat
```

```
      [,1] [,2] [,3]
[1,]    4    8    4
[2,]    7    6    4
```

```
> #menambahkan kolom
```

```
> mat1=cbind(mat,c(1,1))
```

```
> mat1
```

```
      [,1] [,2] [,3] [,4]
[1,]    4    8    4    1
[2,]    7    6    4    1
```

```
> mat2=cbind(c(1,1),mat)
```

```
> mat2
```

```
      [,1] [,2] [,3] [,4]
[1,]    1    4    8    4
[2,]    1    7    6    4
```

```
> #menambahkan baris
```

```
> mat3=rbind(mat,c(1,1,1))
```

```
> mat3
```

```
      [,1] [,2] [,3]
[1,]    4    8    4
[2,]    7    6    4
[3,]    1    1    1
```

```
> mat4=rbind(c(1,1,1),mat)
```

```
> mat4
```

```

      [,1] [,2] [,3]
[1,]    1    1    1
[2,]    4    8    4
[3,]    7    6    4

```

Dapat dilihat bahwa posisi vektor yang ditambahkan dalam argumen mempengaruhi tata letak vektor pada matriks

Ekstraksi matriks

```

> matrik=matrix(2:7,2)
> matrik
      [,1] [,2] [,3]
[1,]    2    4    6
[2,]    3    5    7

> #menampilkan baris ke 2 kolom 3 dari matrik
> matrik[2,3]
[1] 7

> #menampilkan elemen baris ke 2
> matrik[2,]
[1] 3 5 7

> #menampilkan elemen kolom 1
> matrik[,1]
[1] 2 3

```

Data Frame

Objek data frame dapat dibuat menggunakan perintah **data.frame**. Contoh pembuatan data frame dapat dilihat berikut :

```

> nama = c("meme", "mimi", "mumu", "momo")
> usia = c(19, 20, 21, 22)
> kelas = c("A", "B", "C", "C")
> domisili = c("bantul", "sleman", "gunung kidul", "bantul")
> siswa = data.frame(nama, usia, kelas, domisili)
> siswa

```

	nama	usia	kelas	domisili
1	meme	19	A	bantul
2	mimi	20	B	sleman
3	mumu	21	C	gunung kidul
4	momo	22	C	bantul

Mengubah nama kolom pada data frame

Bisa dilakukan dengan fungsi names, sebagai berikut

```

> names(siswa)[2]="umur"
> siswa

```

	nama	umur	kelas	domisili
1	meme	19	A	bantul
2	mimi	20	B	sleman
3	mumu	21	C	gunung kidul
4	momo	22	C	bantul

Ekstrasi Data Frame

```

> #Menampilkan elemen kelas
> #cara 1
> siswa["kelas"]

```

	kelas
1	A
2	B
3	C
4	C

```

> #cara 2
> siswa[3]

```

	kelas
1	A
2	B
3	C
4	C

```
> #jika ingin output berupa vektor
```

```
> #cara satu
```

```
> siswa[,3]
```

```
[1] "A" "B" "C" "C"
```

```
> #cara dua
```

```
> siswa$kelas
```

```
[1] "A" "B" "C" "C"
```

```
> #output sama tetapi nama kolom berbeda
```

```
> as.data.frame(siswa[,3])
```

```
siswa[, 3]
```

```
1          A
```

```
2          B
```

```
3          C
```

```
4          C
```

```
> #menyimpan nama kolom yang baru ke dalam objek bernama coba
```

```
> coba = as.data.frame(siswa[,3])
```

```
> coba
```

```
siswa[, 3]
```

```
1          A
```

```
2          B
```

```
3          C
```

```
4          C
```

```
> #menyimpan elemen baris 2 sampai 4 dan kolom 1 sampai 2
```

```
> new.df=siswa[2:4,1:2]
```

```
> new.df
```

```
nama umur
```

```
2 mimi    20
```

```
3 mumu    21
```

```
4 momo    22
```

```
> #menampilkan elemen baris 1,3 dan kolom 2,4 dari data frame siswa
```

```
> siswa[c(1,3),c(2,4)]
```

```
umur      domisili
```

```
1   19          bantul
```

```
3   21 gunung kidul
```

```
> #menampilkan selain kolom ke tiga dari data frame siswa
```

```
> siswa[,-3]
```

```
nama umur      domisili
```

```
1 meme    19          bantul
```

```
2 mimi    20          sleman
```

```
3 mumu    21 gunung kidul
4 momo    22          bantul
```

```
> #menampilkan selain baris ke 4 dari dataframe siswa
```

```
> siswa[-4,]
  nama umur kelas    domisili
1 meme   19     A    bantul
2 mimi   20     B    sleman
3 mumu   21     C gunung kidul
```

```
> #menampilkan selain baris 3,4 dan kolom 1 dari data frame siswa
```

```
> siswa[c(-3,-4),-1]
  umur kelas domisili
1   19     A  bantul
2   20     B  sleman
```

Mengimpor Data, Fungsi Statistika Dasar dan R Studio

Mengimpor Data

Mengimpor data dari file excell

```
#sebelumnya harus diinstall package, sebagai berikut, penginstalan
hanya dilakukan sekali saja
install.packages("openxlsx")
```

```
#memanggil library yang diperlukan
library(openxlsx)
```

```
#membaca data dari file xlsx
dealing=read.xlsx("D:/KULIAH/Asprak/Pertemuan 1/data dealing.xlsx")
```

Mengimpor data dari difile.csv

```
#membaca data dari file.csv
data.train=read.csv("D:/KULIAH/Asprak/Pertemuan 1/data train.csv")
```

Fungsi Statitika Dasar

Kita akan membahas fungsi `head()`, `tail()`, dan `str()`. Kita dapat menggunakan fungsi *head* untuk memperoleh output data hanya pada sebagian data pertama. *Default* banyak data yang muncul adalah 6, tetapi dapat dimodifikasi sesuai keinginan dengan menambahkan angka seperti berikut :

```
> head(dealing)
> head(dealing)
  1 2 0 1 12
1 1 2 0 1 756
2 1 2 0 1 1048
3 1 2 0 1 1048
4 1 3 0 2 1482
5 1 2 0 1 1482
6 1 2 0 1 1482

> head(dealing, 4)
> head(dealing, 4)
  1 2 0 1 12
1 1 2 0 1 756
2 1 2 0 1 1048
3 1 2 0 1 1048
4 1 3 0 2 1482
```

Selanjutnya fungsi `tail()` dapat digunakan untuk mendapatkan 6 data terakhir, tetapi juga bisa dimodifikasi sesuai kebutuhan.

```
> tail(data.train)
> tail(data.train)
  buyer_id adults children room
335659      9999      2        2     1
335660      9999      2        2     1
335661      9999      2        2     1
335662      9999      1        0     1
335663      9999      2        2     1
335664      9999      2        2     1
... ..

> tail(data.train, 7)
> tail(data.train, 7)
  buyer_id adults children room
335658      9999      2        2     1
335659      9999      2        2     1
335660      9999      2        2     1
335661      9999      2        2     1
335662      9999      1        0     1
335663      9999      2        2     1
335664      9999      2        2     1
```

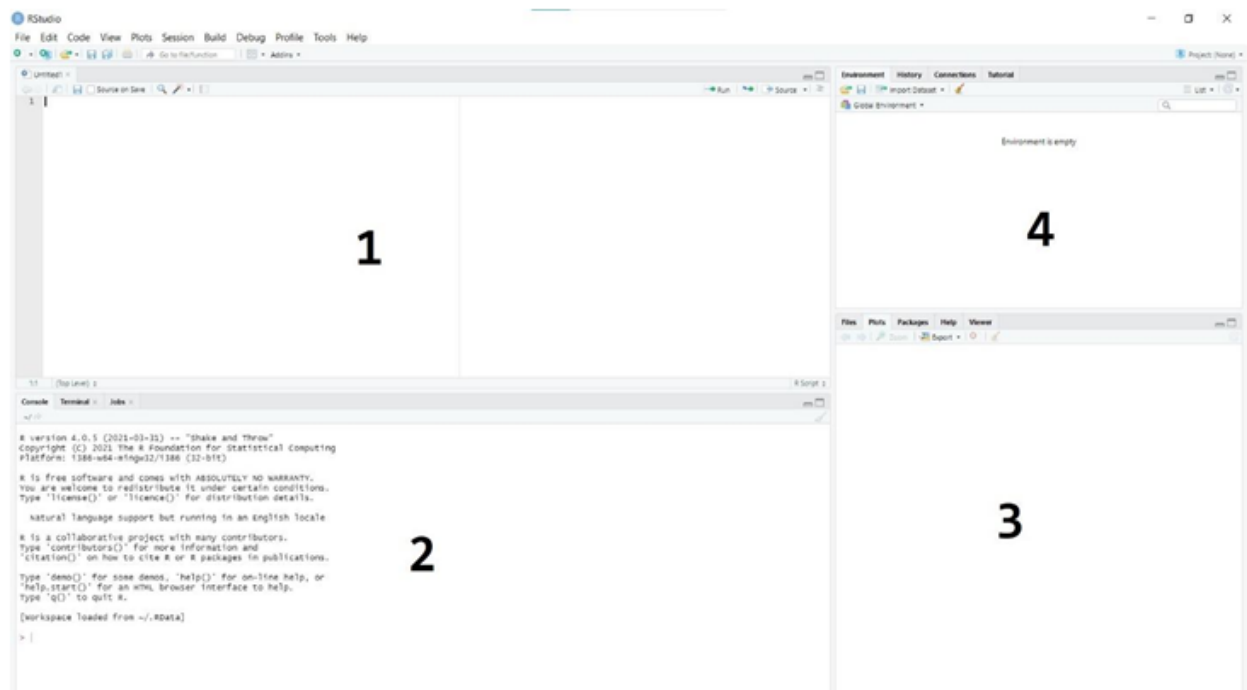
Selanjutnya fungsi `str()` dapat digunakan untuk melihat data kita seperti apa, misalnya bisa dilihat tipe data, jumlah baris dan kolom, dan nama kolom.

```
> str(data.train)
```

```
'data.frame': 335664 obs. of 4 variables:
 $ buyer_id: int  5 5 5 5 5 5 5 5 5 7 ...
 $ adults   : int  1 1 2 2 1 1 1 1 1 2 ...
 $ children: int  1 1 1 1 1 1 1 1 1 0 ...
 $ room     : int  1 1 1 1 1 1 1 1 1 1 ...
```

R Studio

R Studio merupakan pengembangan IDE (*Integrated Development Environment*) untuk R, yang memberikan tampilan *interface* R yang lebih baik dan cocok digunakan untuk pekerjaan yang lebih profesional dan kompleks. Perlu diketahui dalam menjalankan R Studio kita perlu menginstall R terlebih dahulu.



Keterangan :

- Nomor 1 : R Script
- Nomor 2 : R Console
- Nomor 3 : *Global environment* dari *working space* kita
- Nomor 4 : Menunjukkan hasil plot, file dalam direktori kerja, fitur help, dan *viewer*.

Secara keseluruhan tidak ada perbedaan dalam menuliskan *syntax* yang ada dalam R pada R Studio. Adapun, dalam menjalankan *syntax* secara langsung dalam script pada R Studio dapat menggunakan `ctrl + Enter`.

