

# Uvod

---

R je programski jezik i softversko okruženje za statističku analizu, grafičko predstavljanje podataka, pravljenje izveštaja.

Kreirali su ga Ross Ihaka i Robert Gentleman sa University of Auckland, New Zealand.

Prva verzija je objavljena 1993. godine.

Od 1997. godine, R Core Team se bavi razvojem ovog programskog jezika.

R je interpreterski jezik.

Ključne karakteristike:

- Jednostavan i efikasan programski jezik koji podržava uslovno izvršavanje, cikluse, korisnički definisane rekurzivne funkcije i olakšan ulaz/izlaz
- Efikasan rad sa podacima i njihovo smeštanje
- Podržava operatore za rad sa nizovima, listama, vektorima i matricama
- Omogućava grafičku predstavu podataka

R je najpopularniji i najviše korišćen programski jezik za statističke proračune.

## Instalacija i pokretanje grafičkog okruženja R

---

Instalacija R-a se vrši u dva koraka:

1. Potrebno je dobiti odgovarajući R instalacioni paket sa <https://cran.r-project.org/index> i pratiti detalje instalacije za odgovarajući sistem. Postoje verzije za Linux, Mac OS X i Windows, koje su skoro potpuno identične. Pretpostavimo da smo instalirali R na Windows sistemu. Nakon instalacije, u odabranom instalacionom folderu mogu se pronaći verzije 32-bitne i 64-bitne Rgui.exe datoteke koje su sasvim dovoljne za dalji rad.
2. (Opcioni korak) Ukoliko želimo da budemo više produktivni u R-u, dostupan je i besplatan set integrisanih alati pod nazivom RStudio koji se može dobiti sa adrese <https://www.rstudio.com/products/rstudio/download/>. RStudio je dostupan na svim trima platformama, takođe.

## Kako pokrenuti R?

---

R se pokreće klikom na ikonicu R ili u nekom od grafičkih okruženja operativnog sistema (Architect, JGR, R Commander, RStudio, itd.) ili putem komandne linije u Linux terminalu ili na udaljenom serveru: `$ R`

Osnovni R IDE (integrated development environment) je rudimentaran, pa ćemo koristiti Rstudio u nastavku.

## RStudio interfejs

---

RStudio interfejs se deli na četiri glavna panela. Oni obuhvataju:

1. Editor R skripte / koda

U editoru je moguće kreiranje, otvaranje i editovanje fajlova koji sadrže R skripte. Ako na primer želite da kreirate novu R skriptu, selektujte File, New File, R Script ili jednostavnije klikom na ikonu dokumenta sa znakom plus lociranu direktno ispod glavnog menija (gornji levi ugao ispod File stavke menija).

## 2. R konzola

Prozor sa konzolom je mesto na kome upisujete šta želite R da uradi i u ovom delu se takođe prikazuju rezultati vaših komandi. U njemu možete da vidite šta se upravo dešava nakon izvršenog koda.

Komade se mogu direktno napisati u konzoli ali se zaboravljaju kad se zatvori sesija. Predlog je da svakako upišete komande kroz Script Editor R koda i da uvek sačuvate skriptu (možete je podeliti, nastaviti kasnije itd).

Takođe, tokom rada i analize imaćete potrebu da proverite na primer strukturu objekta bez potrebe da sačuvate taj rezultat. Samim tim možete ukucati ove komande direktno u R konzolu i dobiti rezultat.

## 3. Environment / History / Connections / Git

- Environment Tab podrazumeva listu svih aktivnih R objekata, vrednosti, funkcija ili svega ostalog što je kreirano tokom R sesije.
- History Tab čuva istoriju svih vaših prethodnih komandi.
- Connections Tab predstavlja konektor za vaše konekcije (ODBC i Spark).
- Git Tab predstavlja sve opcije za komitovanje, pull, push sa statusima izmena. Ovaj Tab će se pojaviti ako ste uspešno povezali Git instalaciju (više o njoj i samom setupu u nastavku teksta).

## 4. Files / Plots / Packages / Help

- Files Tab prikazuje fajlove u okviru radnog direktorijuma.
- Plots Tab prikazuje sve vizualizacije koje su kreirane tokom R sesije, one se mogu po potrebi sačuvati kao JPEG ili u PDF formatu klikom na Export dugme.
- Packages Tab prikazuje sve eksterne R biblioteke (packages) koje su instalirane na vašem lokalnom sistemu. Ako su biblioteke čekirane one su učitane međutim ako nisu one bi trebalo da budu učitane ako želite da ih koristite. Takođe moguće je i klikom na Packages/Install Packages instalirati željenu biblioteku.

# RStudio Projects

---

RStudio nudi korisnu opciju pod imenom Projects koja omogućava korisnicima da jednostavno i lako promene projekte u okviru njega. Svaki projekat ima drugačije radne foldera i fajlove. Korišćenje projekata se svakako preporučuje zbog bolje organizacija fajlova i skripti na kojima radite.

Treba napomenuti da nakon svakog pokretanja, startuje se sa definisanim radnim direktorijumom. Ukucavanjem `getwd()` kroz Editor možete proveriti koji je trenutni radni direktorijum.

Kako budete sve više koristili sam program, kroz rad i analizu imaćete zasigurno dosta različitih projekata sa različitim fajlovima i skriptama samim tim ovaj radni default direktorijum možete promeniti kroz Session / Set Working Directory / Choose directory.

# RStudio themes

---

RStudio pruža podršku za tri globalne teme koje prilagođavaju sam vizuelni izgled interfejsa.

1. Classic
2. Modern (default)

### 3. Sky

Kroz meni možete promeniti temu klikom na Tools / Global Options / Appearance. Kroz padajući meni selektujte temu koja vam najviše odgovara.

## RStudio & R Markdown

---

R Markdown (.rmd) predstavlja notebook koji sadrži vašu analizu u R kodu, sve vizualizacije koje ste tokom analize kreirali kao i komentare i uvide (insights) u vidu teksta. Sa njim je moguće generisanje kvalitetnih izveštaja koji se mogu brzo i lako podeliti sa drugim korisnicima ili kolegama, kako u vidu fajla tako i u vidu na primer PDF dokumenta.

## Konzola vs. skripta

---

Kao što smo već rekli postoje dva osnovna načina izvršavanja instrukcija u R programskom jeziku:

- Pisanje u konzoli: piše se jedna po jedna instrukcija i tako se i izvršava
- Pisanje u skripti (fajlu): piše se ceo program i kao takav se i izvršava

Konzola se prepoznaje po znaku prompta >.

Primer: Napisaćemo program kojim se ispisuje pozdravna poruka „Zdravo svete“.

```
> print("Zdravo svete!")  
[1] "Zdravo svete!"
```

Skripta se kreira pomoću File → New File → R Script (ili prečicom Ctrl + Shift + N). Skripta ima ekstenziju .R

Primer: Napisati isti program, samo u skripti. **RCode0**

Klikom na Run, dobija se isti ispis u konzoli kao za prethodni primer.

Za pokretanje linije koda (ako koristite RStudio) može se koristiti i Ctrl+Enter.

VAŽNO: Izvršava se deo koda koji je označen. Da bi se izvršio ceo program, potrebno je označiti sve linije koda

## Instalacija paketa

---

U septembru 2018. broj R paketa je prevazišao 13000 (za broj trenutno dostupnih paketa na CRAN pogledati <https://cran.r-project.org/web/packages/>). Instalaciji određenih paketa se može pristupiti kroz GUI (grafički korisnički interfejs - eng. graphical user interface) RStudio-a. Neophodno je iz kartice Packages izabrati dugme Install (ili Tools menija opciju Install Packages) i nakon toga upisati ime paketa i postarati se da Install dependencies opcija bude štiklirana.

Instalacija paketa se takođe može izvršiti i pomoću funkcije `install.packages("imepaketa")`.

Može se reći da su paketi u R-u slični bibliotekama u C, C++ ili Python-u. Paketi u R-u sadrže funkcije, pomoćne tekstove, primere i test podatke. Paketi se prilikom uobičajene instalacije povlače sa CRAN sajta, međutim moguća je instalacija paketa sa nekog drugog mesta na internetu (URL-a) ili GitHub-a. U tome može poslužiti paket devtools. Instalacija sa GitHub-a može biti korisna kada se radi o upotrebi funkcija koje su u razvoju, s obzirom da autori paketa svoje radne verzije često postavljaju tamo pre nego što dostignu potpunu stabilnost, nakon čega se dopunjuju na CRAN repozitorijum. Grupu paket

određene oblasti možete instalirati pomoću paketa `ctv` i komande `install.views("ime grupe")`. Neke od dostupnih grupa paketa za specijalizovane oblasti su:

- mašinsko učenje (<https://CRAN.R-project.org/view=MachineLearning>)
- računarstvo visokih performansi (<https://CRAN.R-project.org/view=HighPerformanceComputing>)
- kliničke studije (<https://CRAN.R-project.org/view=ClinicalTrials>)
- genetika (<https://CRAN.R-project.org/view=Genetics>)
- numerička matematika (<https://CRAN.R-project.org/view=NumericalMathematics>)
- medicinski imidžing (<https://CRAN.R-project.org/view=MedicalImaging>)
- prostorne analize (<https://CRAN.R-project.org/view=Spatial>)
- društvene nauke (<https://CRAN.R-project.org/view=SocialSciences>), itd.

Za čitav spisak paketa grupisanih u oblasti pogledajte <https://cran.r-project.org/> -> Task views.

## Učitavanje paketa

---

Posle instalacije paketa i pokretanja nove sesije u R-u, neophodno je učitati pakete. Nakon učitavanja dostupne postaju određene funkcije, podaci i uputstva sadržana u tom paketu. Učitavanje paketa je neophodno, jer bi u suprotnom svi instalirani paketi prilikom svakog pokretanja R-a bili učitavani, što bi zahtevalo dodatno vreme i resurse. Postojanje paketa odvojenih od osnovne instalacije R-a daje i mogućnost zamene pojedinačnih paketa novim verzijama bez ponovne instalacije. Paketi se učitavaju pomoću komande `library(imepaketa)`.

Svaki put prilikom pokretanja sesije, neophodno je učitati pakete. Ime paketa može biti navedeno sa i bez znakova navoda. Da bi izvršili isključivanje nekog od paketa, dovoljno je pokrenuti funkciju `detach(package=imepaketa)`.

Isključivanje paketa je nekad potrebno kada se više paketa istovremeno koriste u okviru jednog projekta, a imaju funkcije istog imena a različite namene. Međutim, to možemo prevazići istovremenim pozivanjem određenog paketa i komande `imepaketa : : imekomande`.

Koje pakete imamo instalirane u R-u, proveravamo pokretanjem funkcije `library()`. **RCode1**

## Osnovna sintaksa i operatori

---

R razlikuje velika i mala slova, što je slučaj sa većinom programa baziranih na UNIX-u. Nazivi promenljivih ne smeju počinjati brojkom. Najčešći znakovi korišćeni u R su:

- Tačka-zarez `;` služi za odvajanje naredbi u okviru jednog reda. U slučajevima kada se naredbe nalaze u različitim redovima, nije ih potrebno razdvajati tačka-zarezom.
- Taraba `#` označava početak komentara koji obuhvata sve napisano do kraja reda, koji R ne izvršava.
- Vitičasta zagrada `{ }` grupiše izraze u blokove.
- Plus `+` R sam ispisuje ako naredba nije stala u jedan red.

Neke od korisnih opštih komadi:

- `imeobjekta` - prikazuje sadržaj objekta
- `length(imeobjekta)` - ispisuje dužinu objekta od interesa
- `attributes(imeobjekta)` - daje osnovne informacije o objektu

- `class(imeobjekta)` – ispisuje klasu objekta
- `str(imeobjekta)` - prikazuje internu strukturu R objekta.
- `c(imeobjekta,imeobjekta,...)` - kombinuje objekte u vektor
- `cbind(imeobjekta,imeobjekta,...)` - kombinuje objekte kao kolone
- `rbind(imeobjekta,imeobjekta,...)` - kombinuje objekte kao redove
- `ls()` - izlistava dostupne objekte
- `rm(imeobjekta)` - briše objekat
- `novi_objekt <- edit(imeobjekta)` - menja i čuva sadržinu u novi objekat
- `fix(imeobjekta)` - menja postojeći objekt

Operatori su simboli koji upućuju program (njegov kompajler koji prevodi komande u mašini razumljiv jezik) da izvrši određene matematičke ili logičke operacije.

R je programski jezik bogat operatorima.

R poseduje aritmetičke operatore, relacije i logičke, operatore dodele i ostale operatore .

#### Aritmetički operatori

Operator	Opis
+	sabiranje
-	oduzimanje
*	množenje
/	deljenje
<sup>^</sup> ili **	eksponent
%%	Daje ostatak od deljenja dva broja (vektora) – mod
%/%	Daje količnik dva broja u integeru
x & y	x i y

### Relacioni i logički operatori

Operator	Opis
<	manje
<=	manje ili jednako
>	veće
>=	veće ili jednako
==	jednako
!=	različito (nije jednako)
!x	nije x
x   y (x    y)	x ili y (kraća forma uzima sve elemente vektora; duža samo prvi)
x & y (x && y)	x i y (kraća forma uzima sve elemente vektora; duža samo prvi)
isTRUE(x)	testira da li je x TAČNO
\$ i @	Ekstrakcija komponenti
::	Pristup promenljivim u određenom imenskom prostoru

### Operatori dodele

Operator	Opis
<- ili = ili <<-	Levi operator dodele
-> ili ->>	Desni operator dodele

Tabela 3.4: Ostali operatori

Operator	Opis
:	Dvotačka kreira niz brojeva prilikom pravljenja vektora
%in%	Koristi se za identifikaciju da li element pripada vektoru
%*%	Operator za množenje matrica

## Osnovne komande u R-u

R se može iskoristiti kao kalkulator.

```
> 4+4
[1] 8
> (2*2)+1
[1] 5
```

```
> 5/0  
[1] Inf
```

Mogu se koristiti i neke od ugrađenih funkcija:

```
> exp(23)  
[1] 9744803446  
> sqrt(9)  
[1] 3  
> cos(3)  
[1] -0.9899925
```

Oznaka [1] govori da se radi o prvom članu u nizu. Tako je i kod naredne komande koja ispisuje sve intedžere (cele brojeve) u intervalu od 1 do 55, gde se brojevi u uglastim zagradama, [ ], odnose na redni broj članova niza na početku reda.

```
> 1:55  
[1] 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26  
27 28 29 30 31 32 33 34 35  
[36] 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55
```

Ista komanda se može zapisati i kao:

```
"":(1,55)
```

Slično, moguće je izvršiti komandu za ispis 20 nasumičnih brojeva koji se nalaze između 30 i 35

```
> runif(20,30,35)
```

Vratićemo se kasnije na sve ovo, ovde je navedeno samo kao primer šta sve možemo da radimo.

Izvršenje komande može potrajati. Njen prekid se može izvršiti komandom ctrl + c u Linux-u ili Esc u Windows-u ili Mac-u. Nekada prekidanje komande može potrajati duže vreme.

Za ispis korisničkog uputstva o korišćenju ugrađenih funkcija, koristi se funkcija help(), npr.help("print").

```
> help(cos)  
> example(exp)
```

Jednolinijski komentari počinju sa znakom #. **RCode2**

Višelinijski komentari zvanično ne postoje, ali se mogu dobiti stavljanjem teksta pod jednostruke ili dvostruke znake navoda. **RCode3**

## Osnovne matematičke funkcije

---

R je razvijen od strane statističara usled nedostatka adekvatnih programskih jezika za jednostavnu primenu složenijih matematičkih operacija. Pre korišćenja naprednijih funkcija, neophodno je upoznati se sa osnovnim operacijama:

- sum(x) – suma elemenata vektora x
- mean(x) – srednja vrednost

- `median(x)` – medijana
- `quantile(x)` – kvantili
- `var(x)` – varijansa
- `sd(x)` – standardna devijacija
- `log(x)` – prirodni logaritam
- `exp(x)` – eksponencijalna funkcija (`ex`)
- `max(x)` – najveći element
- `min(x)` – najmanji element, itd.

## Promenljive

---

Osnovni tipovi podataka su `numerical`, `character`, `logical`. Podaci se čuvaju u promenljivim. Promenljive su integralni deo svakog programskog jezika. Za razliku od nekih drugih programskih jezika, R ne zahteva prethodno imenovanje promenljivih. R omogućava korisniku da podatke različitih oblika čuva unutar R promenljivih.

Promenljiva je uslovno ime na osnovu koga pozivamo grupu podataka sačuvanih u jednom od mogućih osnovnih tipova (pojedinačni brojevi/slova/reči, vektori, matrice, tabele i liste). Promenljive mogu biti bilo kog tipa podataka i mogu menjati vrednosti, kao i sam tip tokom rada. Ime promenljive može sadržati slova, brojeve, donju crtu, tačku, ali ne sme počinjati brojem ili tačkom koju prati broj, a ne sme biti ni jedna od rezervisanih reči (`if`, `else`, `repeat`, `while`, `function`, `for`, `TRUE`, `FALSE`, `NULL`, `Inf`, `NaN`, `NA`, itd.; o nekim od ovih reči bavićemo se u nastavku). Kada se postojećem R objektu dodeljuju nove vrednosti, on zaboravlja stare i prima poslednji unos.

Operator dodele vrednosti u programskom jeziku R je `<-`.

Alternativno, mogu se koristiti i operatori `=` i `<=` za dodelu vrednosti. [RCode4](#)

Primer: Kreirati promenljivu `jabuke` i dodeliti joj vrednost 5. Potom kreirati promenljivu `kruške` i dodeliti joj vrednost 6. Ispisati sumu obe promenljive i kreirati promenljivu `voće`. [RCode5](#)

Ispis svih promenljivih trenutno učitanih se ostvaruje pozivom funkcije `ls()`.

Pretraga promenljive po početku imena vrši se sa parametrom `pattern`.

Brisanje promenljive se vrši pozivom funkcije `rm()`.

Brisanje svih promenljivih se vrši pomoću parametra `list`. [RCode6](#)

## Tipovi podataka

---

Promenljive se u R-u ne deklariraju po tipu, eksplicitno. U najčešće korišćene tipove podataka u R-u spadaju: primitivni tipovi podataka (atomske vektori), vektori (`vectors`), matrice (`matrix`), tabele (`data.frame`), nizovi (`array`) i liste (`lists`).

Svaki od tipova objekta ima svoje prednosti i mane vezano za upotrebu u različitim situacijama.



### Tipovi objekata u R-u, opisi i njihove klase

Objekt	Opis	Klasa
Broj ili tekst	Broj(slovo, reč)	numeric ili character
Vektori	Niz brojeva (slova, reči)	numeric ili character
Matrice	Matrica sa kolonama i redovima	numeric ili character
Tabela	Tabele sa kolonama i redovima	data.frame
Liste	Grupe drugih objekata (vektora, matrica, itd.)	list
Faktori	Vektori koji sadrže intedžere i niz karaktera	factor
Nizovi	Objekti sa više od dve dimenzije	array
Datum i vreme		POSIXct, POSIXt

Format `data.frame` je najbliži tabelama koje srećemo u Excelu i drugim statističkim programima, dok su vektori format sa kojim se R najbolje snalazi.

U bazičnoj instalaciji R-a postoji veći broj setova test podataka, kao i u pojedinačno instaliranim paketima. Lista podataka u bazičnoj instalaciji R se ispisuje funkcijom `data()`.

Možemo ispitati klasu objekta funkcijom `is()`, npr. `is(x, "vector")` ili skraćeno: `is.vector(imeobjekta)`

što se može primeniti i na druge

klase, `is.matrix(imeobjekta)`, `is.data.frame(imeobjekta)`, `is.list(imeobjekta)` gde se na upit o klasi podataka dobija logički odgovor (TRUE/FALSE).

Takođe, ime klase se može dobiti korišćenjem funkcije `class()`.

Objekte za koje je to predviđeno možemo prevesti u drugu klasu naredbom `as()`, `as(objekt, "data.frame")` ili skraćeno `as.data.frame(x)` što je moguće primeniti i u drugim slučajevima:

`as.array(x)`

`as.numeric(x)`

`as.logical(x)`

`as.complex(x)`

`as.character(x)`

itd.

Za proveru svih dostupnih tipova funkcija `is()` i `as()` može se iskoristiti funkcija `methods(is)`, odnosno `methods(as)`.

U isto vreme, komandom `length(imeobjekta)` dobijamo podatke o dužini vektora, matrice ili tabele (broj kolona), dok funkcijom `dim(imeobjekta)` dobijamo podatke o dimenziji objekta.

Podatak u R-u može biti predstavljen sa `NaN` ili `NA`. `NaN` je skrećenica (akronim) za `Not a Number` (0/0), odnosno govori da podatak nije broj, dok `NA` predstavlja skrećenicu za `Not Available`. Jasno je

da NaN i NA nisu isto i da postoji potreba za obe oznake. Upitom `is.na()` dobija se TRUE za NA i NaN, dok se upitiom `is.nan()` dobija TRUE za NaN (0/0) i FALSE za NA.