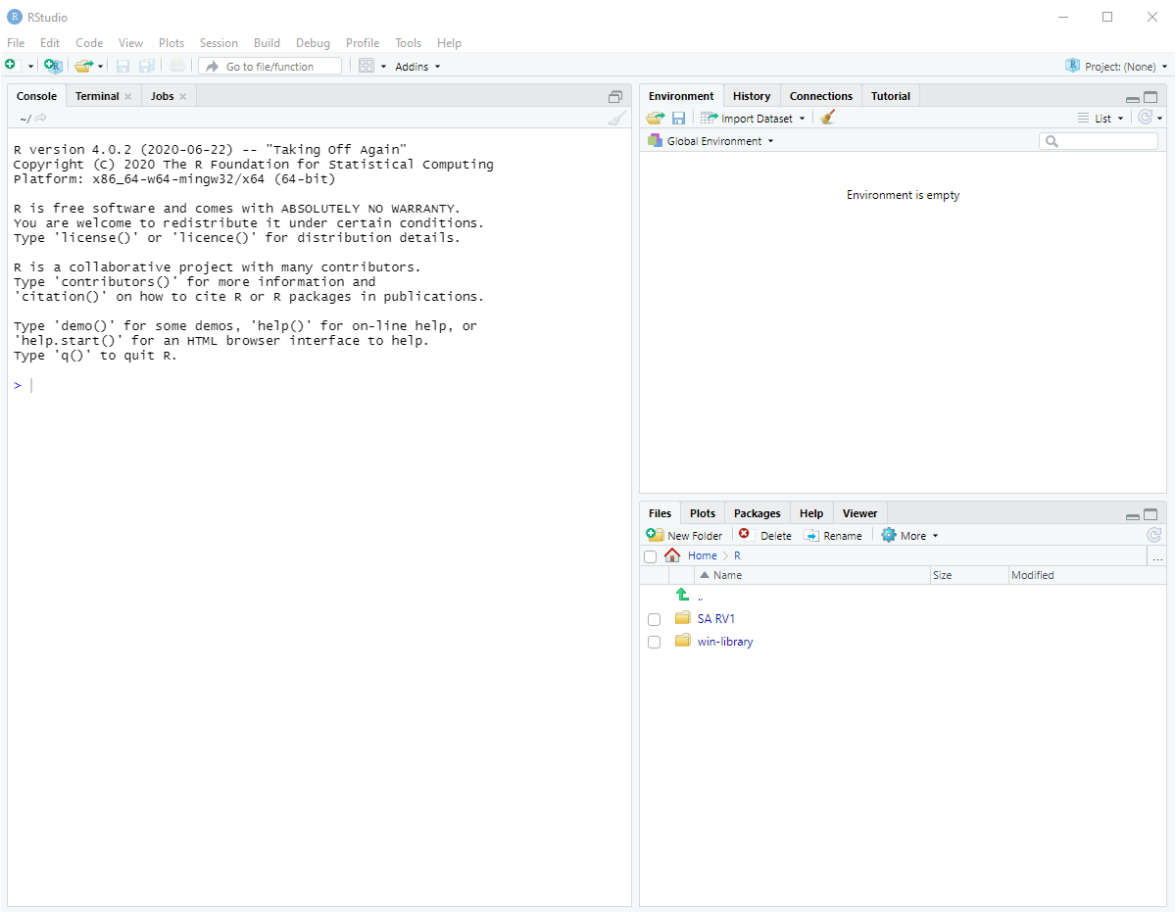
KRATEK UVOD V PROGRAMSKI PAKET R

V učbeniku uporabljamo zelo razširjen brezplačen statistični program R. Deluje na principu funkcij, ki se nahajajo v različnih knjižnicah `{library}` ter jih uporabljamo pri pripravi in statistični obdelavi podatkov. Podobno statistično obdelavo podatkov lahko izvedemo s pomočjo različnih funkcij, zato bomo pri zgledih uporabljali različne poti, da si bralec razširi poznavanje nabora pomembnejših funkcij, ki omogočajo statistično obdelavo podatkov na področju ekonomske in družboslovne znanosti.

Poleg programa R si namestimo tudi razvijalski vmesnik, ki olajša njegovo uporabo. V tem učbeniku uporabljamo uporabniku prijazen brezplačen vmesnik RStudio, ki ga lahko uporabljamo v okolju Windows, MacOS ali Linux. Na računalnik si s spletne strani https://cran.r-project.org/ namestimo zadnjo verzijo programa R, nato pa s spletne strani https://rstudio.com/ še vmesnik RStudio. Alternativno lahko RStudio uporabljamo tudi v oblaku (RStudio Cloud - https://rstudio.cloud/), kjer nam programa ni treba nameščati na računalnik, temveč celotno statistično analizo izvedemo prek spletnega brskalnika in uporabniškega računa na spletni strani RStudio Cloud. Slabost tega pristopa je, da smo omejeni s številom ur brezplačne mesečne uporabe. Ne glede na to, za kateri pristop uporabe se odločimo, je vmesnik RStudio identičen (slika 1).

Slika 1: Vmesnik RStudio



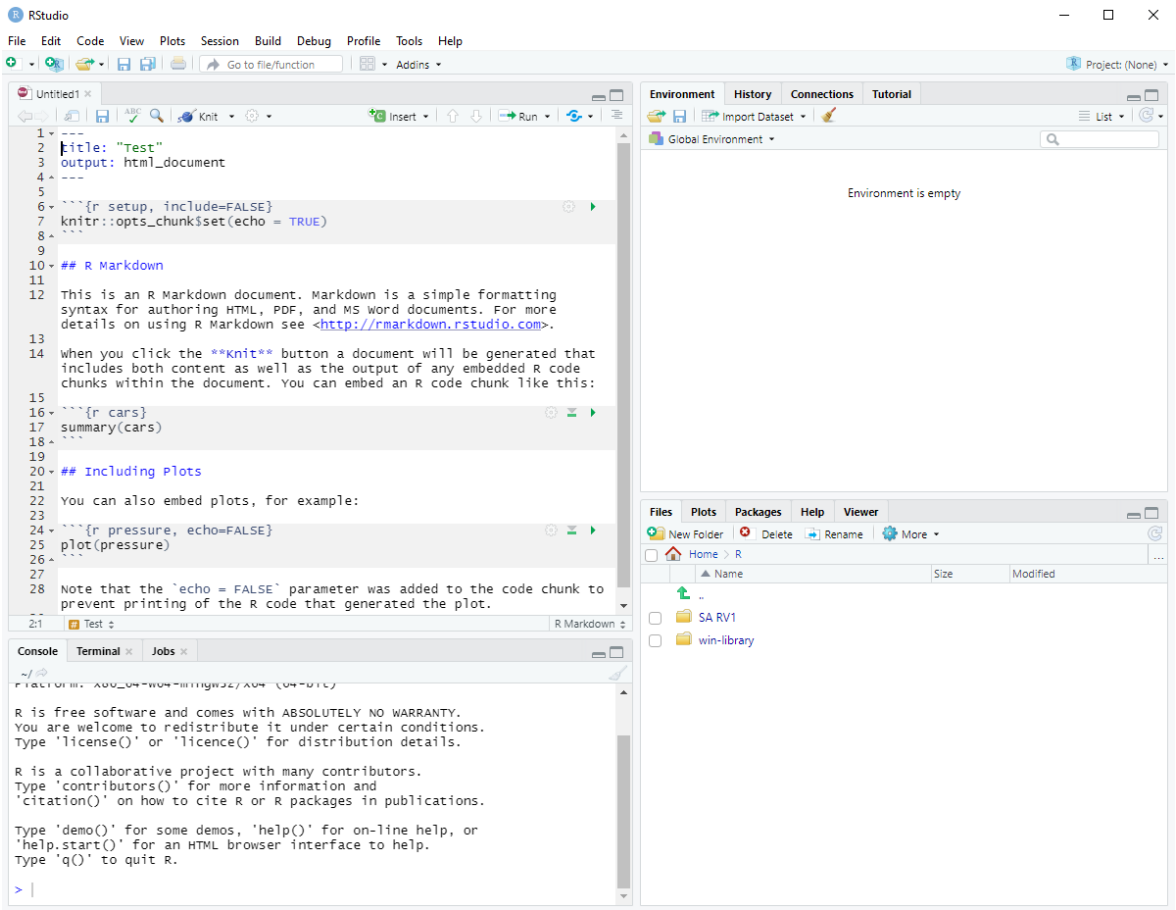
RStudio je sestavljen iz treh podoken. Na levi strani je podokno Console, kamor zapišemo R-ukaze, s katerimi statistično obdelujemo podatke. Na desni strani zgoraj je podokno Environment, kjer so navedene tabele s podatki, do katerih lahko RStudio neposredno dostopa. Na desni strani spodaj

poiščemo datoteko, ki jo želimo statistično obdelati oz. jo s funkcijo Upload naložimo v RStudio Cloud.

Čeprav lahko R-ukaze pišemo neposredno v konzolo, to ni najbolj praktičen pristop, saj se med pisanjem hitro zmotimo, popravljanje ukazov znotraj konzole pa je nepraktično. Boljša izbira je uporaba datoteke R Markdown (končnica .Rmd), ki omogoča preprosto pisanje in urejanje ukazov, nato pa generira poročilo statistične analize. Pomembni prednosti sta ponovljivost izvedbe statistične analize in deljivost z drugimi uporabniki. Novo datoteko R Markdown ustvarimo s funkcijo File - New File - R Markdown.

Ob prvi uporabi bomo morali namestiti določene knjižnice, kar naredi RStudio avtomatično (vpraša nas, če jih želimo namestiti). Poimenujemo R Markdown, pustimo privzeto nastavitev HTML in izberemo OK. Odpre se okno, prikazano na sliki 2. V njem je že podanih nekaj primerov R-ukazov, ki jih lahko poženemo na testnih podatkih, vključenih v okolje R. Ob ustvarjanju novega R Markdowna je ta vedno že vnaprej izpolnjen s testnim primerom, zato pri novi analizi iz datoteke izbrišemo zapise od vključno vrstice 10 naprej.

Slika 2: Primer dokumenta R Markdown znotraj RStudia



Datoteka R Markdown je sestavljena iz belih in sivih polj, slednje pa imenujemo R Chunk. V sivo polje pišemo R-ukaze, rezultati pa se privzeto prikažejo pod sivim poljem (Chunk output inline), v nastavitvah pa lahko tudi spremenimo, da se rezultati prikazujejo v konzoli (Chunk output in Console). Vsak R Chunk se prične z oznako ```` ```{r} ```` in konča z oznako ```` ``` ````, med oznakama pa zapišemo R-ukaz. Navedimo primer.

```
10 ```{r}
11 summary(cars)
12 ```
```

Če želimo v sivo polje navesti komentar, mu moramo spredaj dodati oznako #, saj ga sicer program obravnava kot R-ukaz. V bela polja lahko pišemo naslove, vsebinski tekst, razlage rezultatov ipd. Velikost naslovov lahko spreminjamo s številom znakov #. Več primerov uporabe sintakse si lahko pogledamo na https://www.markdownguide.org/basic-syntax. Navedimo primer.

```
21 ### Opisna statistika (naslov)
22 ```{r}
23 # Za tabelo s podatki, ki se imenuje cars, bomo ocenili izbrane parametre s funkcijo
   summary
24 summary(cars)
25 ```
26 Tukaj bomo razložili rezultate (vsebinski tekst)
```

Novo sivo polje ustvarimo tako, da se postavimo v belo polje, kjer želimo zapisati nov R-ukaz, nato pa izberemo opcijo Insert → R (bližnjica CTRL + ALT + i). Pojavi se nov R Chunk, kamor zapišemo ukaz. Posamezen del funkcije poženemo tako, da izberemo zeleno puščico na desni strani sivega polja ali pa z miško izberemo ustrezne vrstice in na tipkovnici izberemo tipki CTRL + Enter. Primer: Če poženemo sivo polje, v katerem je zapisan ukaz `summary(cars)`, se v belem polju prikažejo ocene statističnih parametrov za testne podatke o avtomobilih, ki so vključeni v okolje R.

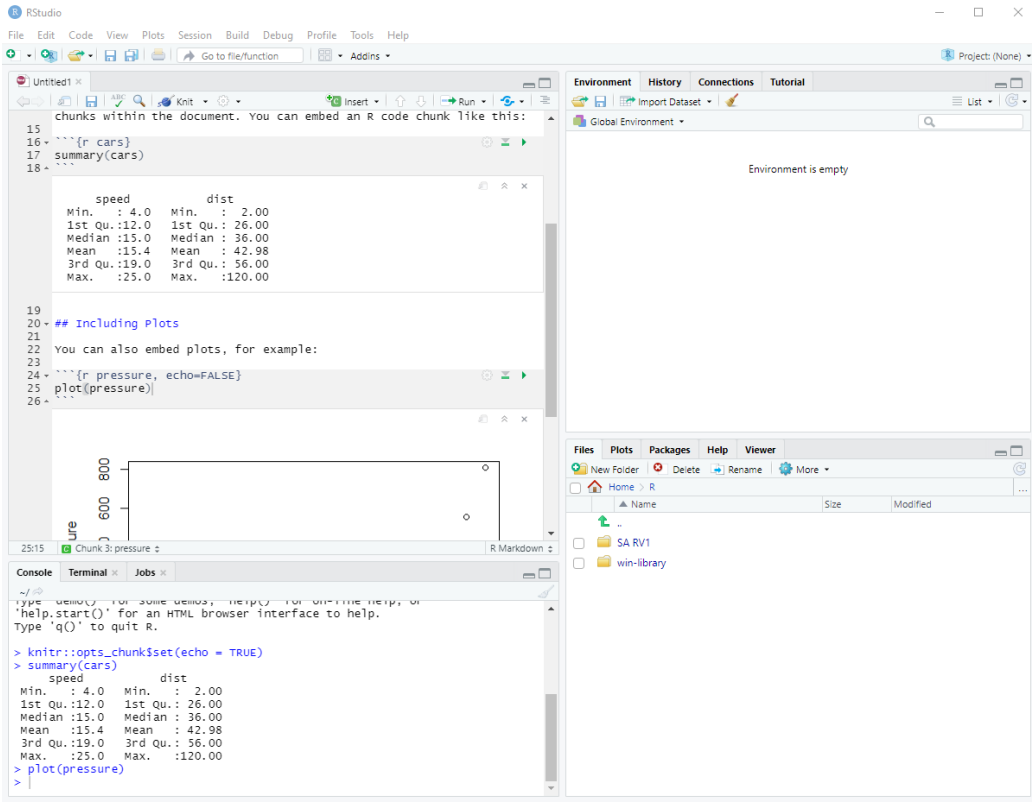
```
21 ### Opisna statistika (naslov)
22 ```{r}
23 # Za tabelo s podatki, ki se imenuje cars, bomo ocenili izbrane parametre s funkcijo
   summary
24 summary(cars)
25 ```

     speed           dist
 Min.   : 4.0   Min.   :  2.00
 1st Qu.:12.0   1st Qu.: 26.00
 Median :15.0   Median : 36.00
 Mean   :15.4   Mean   : 42.98
 3rd Qu.:19.0   3rd Qu.: 56.00
 Max.   :25.0   Max.   :120.00

26 Ocena povprečne hitrosti znaša 15,4 km/h, najkrajša zavorna pot pa znaša 2 metra.
```

Če to naredimo v vrstici `plot(pressure)`, se spodaj izriše testni grafikon (slika 3). Ko v dokument R Markdown zapišemo R-ukaze in vsebinske komentarje, na vrhu R Markdowna izberemo ukaz Knit. Prevede in izvede se celoten dokument in če je vse napisano pravilno, se odpre okno HTML z rezultati (možno je dobiti tudi rezultate v PDF in verziji Word). Dokument R Markdown moramo tudi poimenovati, saj ga program samodejno shrani. Če je prišlo pri kakšnem delu zapisa R-ukazov do napake, nas RStudio o tem obvesti in zapiše, v kateri vrstici je napaka. Ko ukaze ustrezno opravimo, se odpre okno HTML z rezultati statistične analize. Vsebino poljubno kopiramo ali pa celo objavimo na spletu s funkcijo Publish (RPubs) (slika 4).

Slika 3: Uporaba R Markdowna



Slika 4: Okno HTML z rezultati, dobljenimi s pomočjo R Markdowna in ukaza Knit

Testni primer

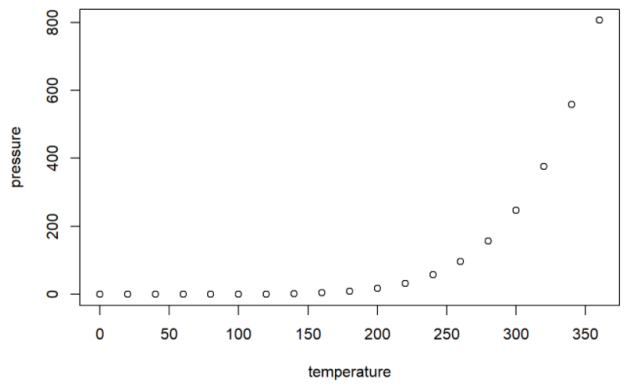
Opisna statistika (naslov)

```
# Za tabelo s podatki, ki se imenuje cars, bomo ocenili izbrane parametre s funkcijo summary
summary(cars)
```

```
##      speed           dist
##  Min.   : 4.0   Min.   :  2.00
##  1st Qu.:12.0   1st Qu.: 26.00
##  Median :15.0   Median : 36.00
##  Mean   :15.4   Mean   : 42.98
##  3rd Qu.:19.0   3rd Qu.: 56.00
##  Max.   :25.0   Max.   :120.00
```

Ocena povprečne hitrosti znaša 15,4 km/h, najkrajša zavorna pot pa znaša 2 metra. (vsebinski tekst)

```
plot(pressure)
```



Pri uporabi R-funkcij (npr. `summary`, `plot` itd.) si lahko pomagamo s prikazom pomoči, kjer so razloženi primeri njihovih ustreznih zapisov. Ukaz za pomoč aktiviramo tako, da z miško izberemo ime funkcije, ki smo jo zapisali v R Markdown, ter na tipkovnici izberemo tipko F1. V vmesniku RStudio se desno spodaj prikaže opis funkcije, ime knjižnice, v kateri se funkcija nahaja – ta je zapisana v zavitem oklepaju `{}`, ter kako se funkcija uporablja. Če si s tem ne znamo pomagati, iskano R-funkcijo preprosto vpišemo v spletni brskalnik, kjer bomo hitro našli primere njene uporabe.

Z namestitvijo programa R avtomatično namestimo najpomembnejše knjižnice (npr. knjižnica `{base}`), marsikatera uporabna knjižnica pa žal ni vključena v osnovno verzijo programa, temveč jo moramo ob prvi uporabi namestiti v RStudio. Ob naslednji uporabi programa namestitev ni več potrebna, temveč jo samo še aktiviramo. Nameščene knjižnice lahko preverimo v zavihku Packages v desnem spodnjem oknu RStudia (trenutno aktivne knjižnice so obkljukane). Knjižnice, naložene v User Library, so knjižnice, ki smo jih dodatno namestili, knjižnice, nameščene v System Library, pa so osnovne knjižnice, ki smo jih pridobili z namestitvijo programa R. Za določene knjižnice nas bo RStudio sam obvestil, da jih je treba namestiti, ostale pa bomo nameščali sproti. Knjižnico namestimo z ukazom:

```
install.packages("ime_knjižnice")
```

Ime knjižnice mora biti navedeno v narekovajih, pri tem smo pozorni na male in velike črke. Ko potrebujemo določeno knjižnico, ki je neaktivna, jo aktiviramo z ukazom:

```
library(ime_knjižnice)
```

Ime knjižnice navedemo brez narekovajev, pri tem smo pozorni na male in velike črke. S tem se knjižnica aktivira in lahko uporabljamo funkcije, ki jih vključuje.

Vse datoteke, ki jih uporabljamo v tem učbeniku, se nahajajo na spletni povezavi, objavljeni na avtorjevi fakultetni spletni strani Ekonomske fakultete UL. V mapah so trije tipi datotek:

- datoteka s končnico .Rmd: datoteka R Markdown, ki vključuje R-ukaze in komentarje;
- datoteka s končnico .html: izpis z rezultati, narejen s funkcijo Knit;
- datoteka s končnico .csv: datoteka s podatki v formatu .csv.

Na koncu uvoda še omenimo, da čeprav je morda uporaba programskega paketa R na začetku zahtevna, ga uporabnik zelo hitro usvoji in je po težavnosti popolnoma primerljiv z drugimi statističnimi programi, obenem pa je odprtokoden in brezplačen. Zaradi velike razširjenosti programa R lahko takrat, ko naletimo na težavo, preprosto uporabimo spletni brskalnik in zelo hitro bomo ugotovili, kaj je narobe v našem zapisu kode.

Na spletni povezavi, kjer se nahajajo datoteke s podatki, so objavljeni tudi videi, ki vas bodo vodili skozi postopek prve uporabe programa RStudio. Spoznali boste, kako se v program uvozijo podatki, ter osnovne principe delovanja programa. Avtorja svetujeva, da si te posnetke predhodno ogledate, saj boste tako lažje sledili učbeniku.

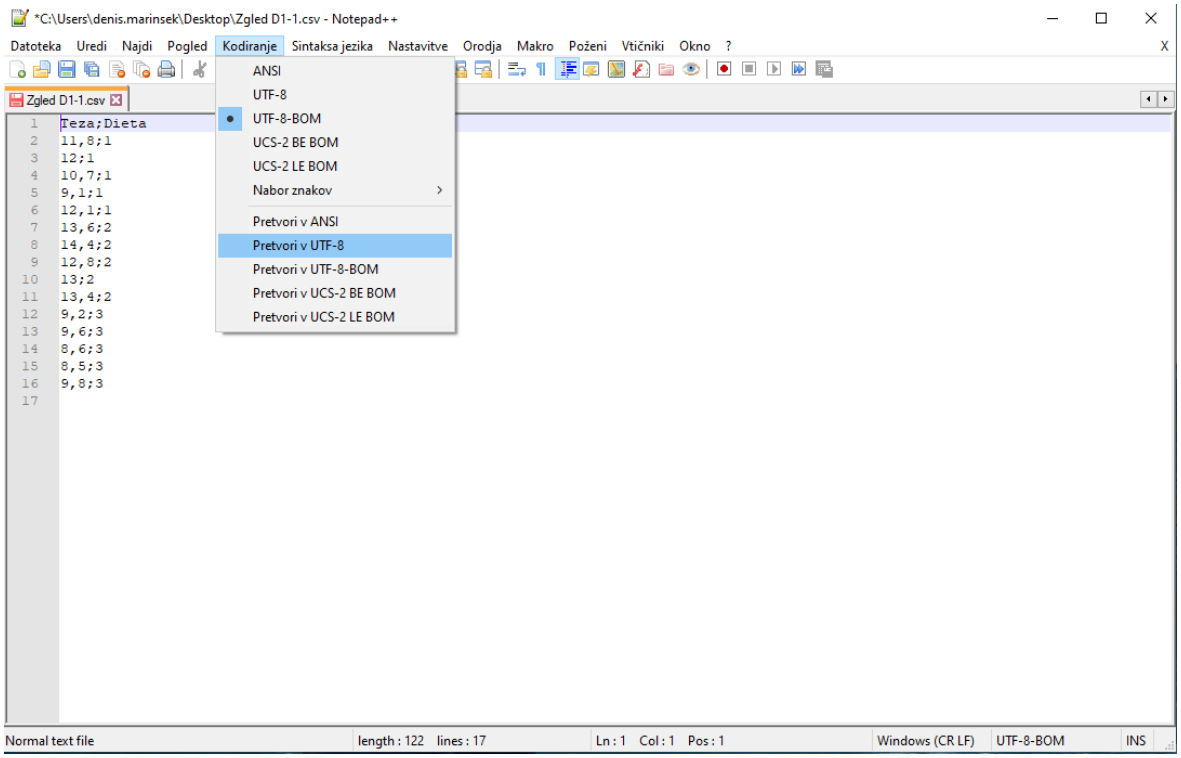
V nadaljevanju je prikazan še preprost primer uporabe RStudia prek R Markdowna z nekaj najosnovnejšimi funkcijami.

V prvem sivem polju definiramo obliko izpisa rezultatov.

```
knitr::opts_chunk$set(echo = TRUE)
options(width=70) #Določimo širino izpisa rezultatov
```

Uvozimo podatke, zaželena je oblika .csv v kodiranju UTF-8. Če imate podatke v datoteki Excel, glejte nalogo 1.4 v poglavju 1.B. Na začetku prikažimo, kako lahko spremenimo kodiranje datoteke .csv. Od kodiranja je namreč odvisno pravilno branje šumnikov, zato je priporočena uporaba kodiranja UTF-8. Kodiranje lahko spremenimo z brezplačnim programom Notepad++, kjer v opciji Kodiranje spremenimo kodiranje v format UTF-8 (Pretvori v UTF-8) ter datoteko shranimo (slika 5). Če pri branju spremenljivk nimamo težav z branjem šumnikov, lahko ta korak izpustimo.

Slika 5: Uporaba programa Notepad++ za spremembo kodiranja datoteke .csv v UTF-8



S funkcijo `read.table()` uvozimo podatke. Tabelo s podatki poimenujemo `naloga`. S funkcijo `head()` izpišemo prvih 6 opazovanj tabele podatkov. Z oznako `<-` dodelimo ukaz, zapisan na desni strani, objektu, zapisanemu na levi strani.

```
naloga <- read.table("/cloud/project/Uvod v R/Študent.csv", #Navedemo
        lokacijo datoteke
                     header = TRUE, #V prvi vrstici so podana imena
                     sep = ";", #Ločilo med spremenljivkami
                     dec = ",", #Decimalno ločilo
                     encoding = "UTF-8") #Uporabimo, v kolikor imamo težavo pri branju
        šumnikov

head(naloga) #Izpišemo prvih 6 vrstic podatkov
##   Višina Teža Tlak
## 1  179.0   70  105
## 2  178.0   68  105
## 3  174.0   64  109
## 4  174.0   63  112
## 5  173.5   61  100
## 6  173.0   60   99
```

Do spremenljivke `Višina`, na primer, dostopamo z ukazom `naloga$Višina` (`ime_tabele+$+ime_spremenljivke`). V podatkovno tabelo vstopamo z ukazom `ime_podatkovne_tabele[,]`. Za konkretni primer torej zapišemo `naloga[,]`. Ukazi, zapisani v oglatem oklepaju pred vejico, se nanašajo na vrstice; ukazi, zapisani za vejico, na stolpce. Tako lahko npr. razvrstimo enote po vrednosti določene spremenljivke.

```
head(naloga[order(naloga$Višina), ]) #Razvrstimo enote v rastočem zaporedju po
        spremenljivki Višina in prikažemo prvih 6 opazovanj
##    Višina Teža Tlak
## 20    166   55  128
## 8     168   57   90
## 9     168   56  100
## 7     170   57   98
## 18    170   58   98
## 19    170   57  107
```

V naslednjem koraku iz obstoječe podatkovne tabele naredimo novo tabelo z izbranima spremenljivkama.

```
naloga1 <- naloga[ , c("Višina", "Teža" )] #Izberemo dve spremenljivki in ju shranimo
        v novo tabelo

str(naloga1) #Prikažemo strukturo nove tabele
## 'data.frame':    20 obs. of  2 variables:
##  $ Višina: num  179 178 174 174 174 ...
##  $ Teža  : int  70 68 64 63 61 60 57 57 56 78 ...
```

Iz strukture tabele lahko ugotovimo tudi tip uporabljenih spremenljivk. Najpogosteje uporabljeni tipi so:

- `num`: poljubna številska spremenljivka, imenovana tudi `double`,
- `int`: številska spremenljivka s celimi vrednostmi (diskretna spremenljivka), `integer`,
- `chr`: opisna spremenljivka, `character`,
- `factor`: kategorialne spremenljivke pogosto spremenimo v faktorje,
- `logical`: logistični test z vrednostmi `TRUE` in `FALSE`.

Spremenljivko `Višina`, ki je izražena v cm, spremenimo v novo spremenljivko `Višina_m`, ki naj bo izražena v m. Opozorimo, da moramo spremenljivko poimenovati z eno besedo. Nato izračunamo indeks telesne mase (`BMI`) ter prikažemo prvih 6 opazovanj.

```
naloga1$Višina_m <- naloga1$Višina / 100 #Izračunamo novo spremenljivko

naloga1$BMI <- naloga1$Teža / naloga1$Višina_m^2 #Izračunamo BMI

head(naloga1) #Izpišemo prvih 6 opazovanj
##   Višina Teža Višina_m      BMI
## 1  179.0   70    1.790 21.84701
## 2  178.0   68    1.780 21.46194
## 3  174.0   64    1.740 21.13886
## 4  174.0   63    1.740 20.80856
## 5  173.5   61    1.735 20.26427
## 6  173.0   60    1.730 20.04745
```

Za spremenljivko `BMI` izračunamo opisno statistiko in narišemo grafikon kvantilov.

```
summary(naloga1$BMI) #Opisna statistika za spremenljivko BMI
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   19.72   20.06   21.10   21.20   22.35   23.25

boxplot(naloga1$BMI) #Grafikon kvantilov za spremenljivko BMI
```