Martin Kolnes, Karin Täht

Table of Contents

# Praktikumi eesmärgid

* RStudio tutvumine
* Andmetega manipuleerimine
* Üldine kirjeldav statistika: keskmine, mediaan, standardhälve

Laadige alla [praktikumi andmed](https://drive.google.com/open?id=1oOK59Dp1jMNUKzuGvKb1OxnePYFx0PPr).

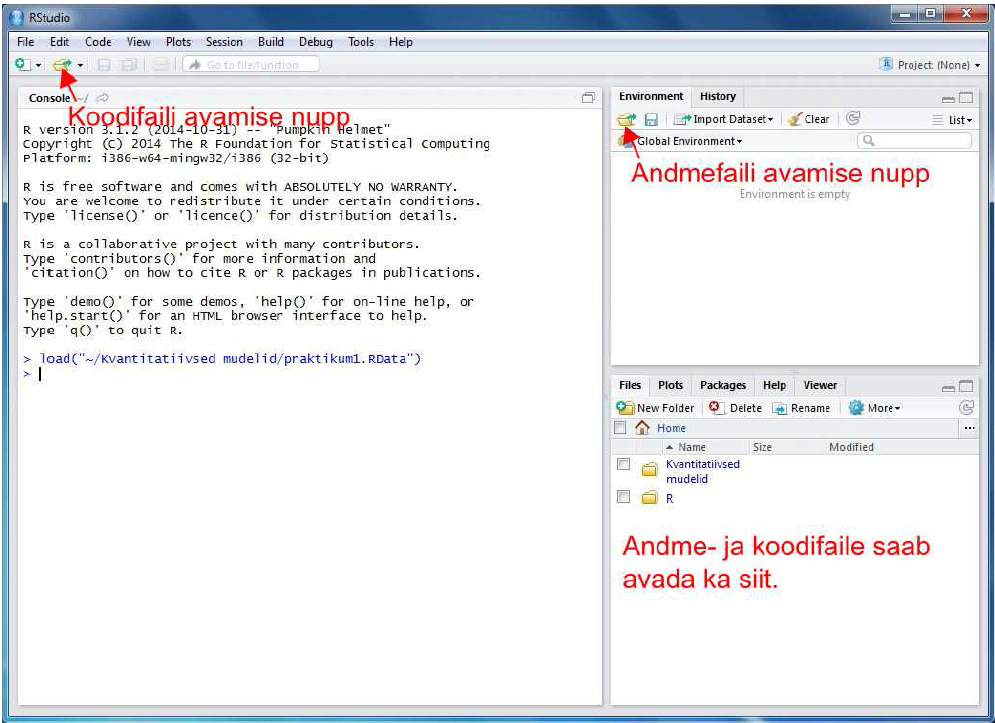
# RStudio paigaldamine

Arvutiklassi arvutites on vajalikud programmid juba olemas, aga kui on soovi paigaldada RStudio ka enda arvutisse, siis kasutage järgnevaid linke:

* programmeerimiskeel R - <http://ftp.eenet.ee/pub/cran>
* RStudio - <http://www.rstudio.com/products/rstudio/download>

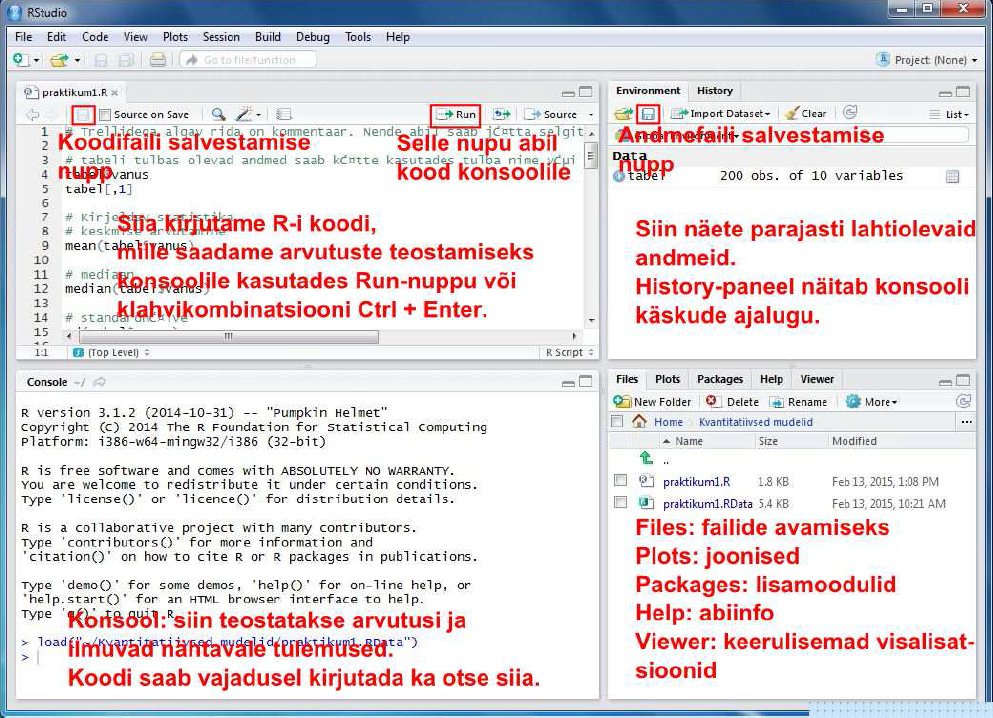
# 1. RStudio kasutamine

Käivitame RStudio. Avanema peaks allolev pilt.



Joonis 1. RStudio ülevaade

Avame andmefaili “1praktikumAndmed.RData” kasutades ülaloleval joonisel osutatud nuppu või RStudio aknas paremal allosas paiknevat paneeli *Files*. Soovitan avada ka tühi koodifail (RScript), valides RStudio menüüribalt *File* -> *New File* -> *R Script*. Koodifaili saate salvestada enda kirjutatud andmeanalüüsi.

Nüüd peaks meil RStudios olema lahti neli akent. Erinevate akende funktsioonid on toodud allpool. 

**Üleval vasakul** olevasse aknasse kirjutame koodi, mille abil ütleme R-ile, mida andmetega teha. Kuidas käib koodi saatmine R-ile? Kui soovime saata ainult ühte rida korraga, siis tuleb viia kursor sellel koodireal ja seejärel vajutada koodiredaktori paneeli paremal ülaosas paiknevat nuppu *Run*. Nupu asemel võib kasutada ka klahvikombinatsiooni *Ctrl + Enter* (Maci arvutitel *Cmd + Enter*). Kui soovime saata mitut koodirida korraga, siis tuleb need koodiredaktoris valida (st teha hiirega siniseks) ja seejärel vajutada *Run*-nuppu või klahvikombinatsiooni.

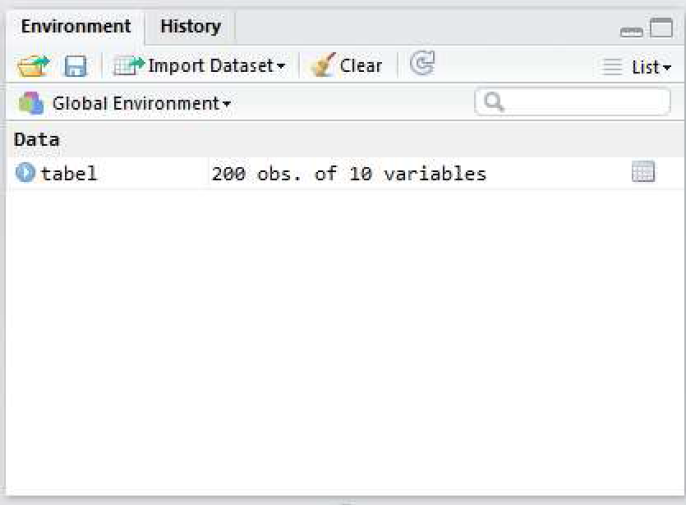
**All vasakul** on konsool (*Console*). Seal asubki arvutusi teostav R ise. Kui laseme R-il midagi arvutada, siis ilmuvad tulemused nähtavale just selles osas. Koodi võib põhimõtteliselt kirjutada ka otse siia, aga enamasti on mugavam kirjutada kood koodifaili. Nii on lihtsam koodi parandusi teha ja koodi korduvkasutada.

**All paremal** asub 5 erinevat paneeli:

* *Files*: R-i koodi- ja andmefailide avamiseks.  
  **NB!** Selles aknas saate sättida enda töökeskkonda. See annab R’ile teada, missuguse kaustaga parasjagu tööd tehakse. Kausta otsimiseks vajutage sümbolile “…”, seejärel otsige üles sobiv kaust ja vajutage “OK”. Seejärel vajutage nupule *More* ja valige *Set As Working Directory*. Soovitatav on kasutada töökeskkonnana kausta, kus on teie analüüsitavad andmed.
* *Plots*: siia kuvatakse tehtud joonised.
* *Packages*: selle osa abil saab installeerida ja laadida lisamooduleid, mis lisavad R-ile täiendavat funktsionaalsust. Lisamoodulid tuleb kõigepealt arvutisse installeerida. Vajutage nuppu *Install* ja kirjutage avenanud aknasse soovitud lisamooduli nimi. Mooduli funktsioonide kasutamiseks tuleb vastav moodul R-is aktiveerida. Aktiveerimseks tuleb moodul *Packages*-paneeli nimekirjast üles otsida ja selle nime eesasuvasse kasti linnukene teha. Lisamooduleid saab installida ja laadida ka R-i koodi abil: *install.packages(“lisamooduli nimi”)* ja *library(“lisamooduli nimi”)*.
* *Help*: Abiinfo R-i funktsioonide kohta. Siin paneelis saate otsida R-is olevate funktsioonide kohta informatsiooni.
* *Viewer*: Selle osa abil saab teha keerulisemat tüüpi andmevisualisatsioone, millel on lisaks joonisele ka kontrollelemendid, mis võimaldavad joonise parameetreid muuta. Seda osa me antud kursusel ei kasuta.

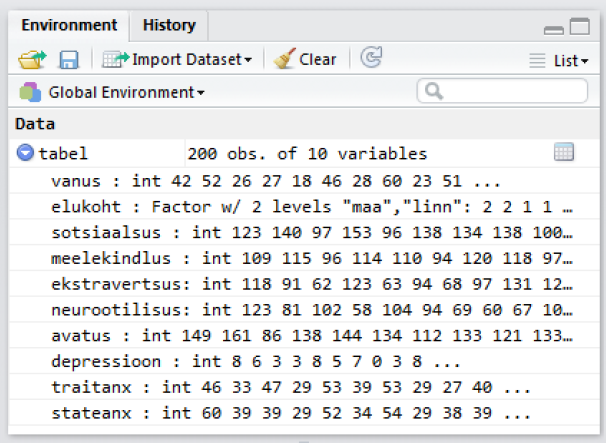
**Üleval paremal** asub 2 paneeli:

* *Environment*: selles osas näidatakse kasutusel olevaid andmeid.
* *History*: R-i konsoolile saadetud käskude ülevaade.  
  Natuke lähemalt *Environment* paneelist. Kui avasite alguses andmefaili “1praktikumiAndmed.RData”, siis peaks paneel välja nägema nagu alloleval pildil (Joonis 3). Objekt nimega “tabel” viitab konkreetse andmetabeli nimele. 200 *obs* of 10 *variables* näitab, et andmetabelis on 200 rida ja 10 tulpa. Erinevalt näiteks SPSS-ist võivad R-i andmefailid sisaldada rohkem kui ühte andmetabelit (ja lisaks andmetabelitele ka teistsuguseid andmestruktuure). Igal andmetabelil on oma nimi ja andmestikus olevad andmed saamegi hiljem kätte selle nime abil.



Joonis 3. RStudio Global Environment

Klõpsates andmestiku nime ees oleval sinisel nupul peaks avanema allolev pilt (Joonis 4).



Joonis 4. RStudio ülevaade

Meile näidatakse tabelis olevate veergude nimesid, andmetüüpe ja esimesi andmepunkte. Esimene veerg/muutuja kannab nime “vanus”. Andmetüüp on *int* , mis tuleb inglisekeelsest sõnast *integer* ehk täisarv, st. tegemist on veeruga, milles sisalduvad andmed on täisarvud. (Teine R-is levinud numbriliste andmete tüüp on *num* ehk *numeric*, mis võib sisaldada ka komakohaga arve.) Teine veerg kannab nime “elukoht” ja selle tüübiks on märgitud *Factor* w/ 2 levels „maa“ , „linn“. *Factor* tähistab R-i kontekstis kategooriaid sisaldavat muutujat. Antud juhul on neid kategooriaid 2 tükki: „maa“ ja „linn“. Kõik ülejäänud tabelis olevad muutujad on täisarvulised (*int*).

# 2. R-i süntaks

Enne andmete juurde asumist teeme väikse kõrvalpõike R-i süntaksiga tutvumiseks. Nagu juba eelnevalt mainisin, siis üldiselt on kasulik kirjutada koodi koodifaili (*Rscript*). Selle saate salvestada enda arvutisse ja avada järgmisel kasutamisel uuesti. Konsoolile kirjutatud koodiread lähevad pärast R-i sulgemist kaduma.  
Paar nõuannet ka koodi kirjutamise kohta. Esiteks, R-i koodi kirjutamisel te ei pea jätma koodi tühikuid. Koodi loetavuse mõttes on see siiski soovitatav. Teiseks, punkte ja komasid kasutatakse R-is erinevatel juhtudel. Punkte kasutatakse andmete puhul. Komasid kasutatakse aga erinevate elementide eristamiseks (näiteks on üks number, aga on kaks numbrit).

## Aritmeetika

R-i võib kasutada tavalise kalkulaatorina. Proovige teha järgnevaid tehteid R-i konsoolil.

* liitmine: 2 + 2
* lahutamine: 4 - 5
* korrutamine: 5 \* 7
* jagamine: 21 / 3
* astmesse tõstmine: 2^4

## Muutujate loomine

Vahepeal on vajalik luua andmetöötluse käigus uus muutuja, kuhu on salvestatud mingi oluline väärtus. Uue muutuja loomaikseks kasutatakse kas noolt “<-” või tavalist võrdusmärki “=”. Muutujate nimede puhul ei luba R tühikuid kasutada. Seega on pikema nime puhul hea kasutada punkti (“uus.muutuja”), alakriipsu (“uus\_muutuja”) või suurtähti (“uusMuutuja”).

# Need kaks väärtustamist annavad sama tulemuse  
uus\_muutuja <- 2  
uus\_muutuja = 2

Töökeskonnas olevate muutujatega saate teha ka tavalisi artimeetilisi tehteid.

uus\_muutuja \* 4

## [1] 8

## Andmetüübid ja vektorid

Põhilised andmetüübid, millega kursuse jooksul kokku puutume:

* numbrilised (*numeric, integer*): 2.2, 3
* sõned (*character*): “hello”
* tõeväärtused: *TRUE*, *FALSE*

Miks on need olulised? Kõik andmestikud koosnevad nendest andmetüüpidest. Andmestiku ühte veergu võib käsitleda vektorina, mis koosneb teatud tüüpi elementidest. Vaatame kõigepealt, kuidas R-is saab vektoreid luua. Vektorid (erinevalt listidest) võivad sisaldada ainult ühte tüüpi elemente.

# Numbriline vektor  
num\_vektor <- c(3,1,2)  
# Sõnedega vektor  
s\_vektor <- c("SPSS", "R", "Matlab")  
# Tõeväärtustega vektor  
t\_vektor <- c(FALSE, TRUE, FALSE)

Objekti tüübi kontrollimiseks saame kasutada funktsiooni nimega *class*.

class(num\_vektor)

## [1] "numeric"

## Tehted vektoritega

Kõik arvutused, mis me R-is teeme on seotud vektoritega. Kirjeldava statistika puhul võetakse sageli üks andmeveerg (ehk vektor andmetega) ja vaadatakse selle näitajaid.

# Teeme kaks vektorit:  
v1 <- c(1,4,6,7,3)  
v2 <- c(1:5) # koolon näitab, et teeme vektori 1-st kuni 5-ni  
  
# Liitmine  
v1 + v2  
  
# Korrutamine  
v1 \* v2  
  
# Vektori keskmine  
mean(v1)  
  
# Vektori andmete summa  
sum(v1)  
  
# Vektori mediaan  
median(v1)

## Andmete eraldamine vektorist

R-i üks peamisi tugevusi peitub andmete manipuleerimises. Andmete eraldamiseks saab kasutada indekseid või loogikaavaldisi. Indeksiga võtame konkreetse koha, mis meid huvitab. Loogikaavaldisega saame valida väärtused, mis vastavad meie poolt määratud reeglile.

# Loome vektori  
v3 <- seq(1,10,0.5) #teeme vektori 1-st kuni 10-ni sammuga 0,5  
  
# Eraldame esimese elemendi  
v3[1]  
  
# Eraldame kolmanda, neljanda ja viienda elemendi:  
v3[3:5]  
  
# Eraldame elemendid, mis on suuremad kui 7  
v3[v3 > 7]

## Ülesanded - R-i süntaks

1. Tehke kaks erinevat vektorit, kus on vähemalt 10 numbrilist elementi.

vektor1 <- c(5,6,2,7,2,8,11,4,9,12)  
vektor2 <- c(3,11,10,2,4,6,3,6,9,4)

1. Arvutage nende vektori aritmeetiline keskmine ja mediaan.

mean(vektor1)

## [1] 6.6

median(vektor1)

## [1] 6.5

mean(vektor2)

## [1] 5.8

median(vektor2)

## [1] 5

1. Liitke need vektorid ja salvestage tulemus uude muutujasse. Arvutage uue muutuja keskmine, mediaan ja summa.

vektor3 <- vektor1 + vektor2  
mean(vektor3)

## [1] 12.4

median(vektor3)

## [1] 13

sum(vektor3)

## [1] 124

1. Eraldage eelmises ülesandes loodud muutujast väärtused, mis on suuremad kui selle muutuja aritmeetiline keskmine.

vektor3[vektor3 > mean(vektor3)]

## [1] 17 14 14 18 16

# 3. Andmetega manipuleerimine

Enne alustamist veenduge, et Teil oleks RStudio töökeskkonnas (*Global Environment*) andmestik nimega “tabel” (Joonis 3.).

Tehke esialgu läbi järgnevad ülesanded:

## Ülesanded - andmestikuga tutvumine

1. Kui palju muutujaid on andmestikus? Mis on nende muutujate nimed?
2. Kasutage funktsiooni *head()*, argumendiks pange andmestiku nimi. Mida see funktsioon näitab?
3. Kasutage funktsiooni *tail()*, argumendiks pange andmestiku nimi. Mida see funktsioon näitab?
4. Kasutage funktsiooni *str()*, argumendiks pange andmestiku nimi. Mida see funktsioon näitab?

R’i võimaldab andmeid kiirelt kohandada vastavalt enda vajadustele. Esialgu peame õppima konkreetsete andmete eraldamise suuremast andmestikust.  
Siin tutvustame kolme võimalust:

* Dollari märgi abil viitamine
* Nurgeliste sulgude kasutamine - R’is annavad nurgelised sulud märku, et mingist objektist/andmestikust tahetakse mingit konkreetset muutujat kätte saada.
* Funktsioon *subset* - see funktsioon võimaldab teha sama, mida nurgelised sulud.

## Veergude eraldamine

Dollari märgi abil (peame teadma veeru nime):

tabel$vanus # võtab andmestikus veeru nimega "vanus"

## [1] 42 52 26 27 18 46 28 60 23 51 51 66 41 18 18 23 64 36 59 25 44 36 53  
## [24] 48 42 27 50 37 64 42 47 34 55 21 48 49 43 45 51 52 34 42 65 18 38 48  
## [47] 48 41 51 46 37 46 65 32 60 43 31 60 55 61 24 30 64 58 32 50 62 24 23  
## [70] 56 37 36 38 62 32 34 69 20 52 69 47 30 44 69 28 57 24 20 19 60 33 50  
## [93] 39 26 59 57 24 57 62 51 23 42 62 28 24 49 22 69 37 42 51 18 39 52 65  
## [116] 49 46 39 30 22 69 41 26 49 42 29 65 51 63 59 58 37 51 51 26 48 26 61  
## [139] 50 52 46 34 26 54 35 25 41 36 42 34 24 20 29 23 65 36 34 59 35 35 35  
## [162] 49 59 54 25 50 37 58 31 69 49 70 56 64 29 67 58 56 42 47 68 40 33 47  
## [185] 59 69 31 31 50 66 63 47 41 65 51 53 70 19 63 60

Nurgeliste sulgudega (peame teadma veeru asukohta andmestikus):

#[rida, veerg]  
tabel[,1] # võtab andmestikust esimese veeru

Funktsiooniga *subset* (peame teadma veeru nime):

subset(tabel, select = vanus) # võtab andmestikust veeru nimega vanus   
subset(tabel, select = c(vanus, elukoht)) # kaks veergu. NB! lisasime c() - see näitab, et anname argumendiks vektori.

Kuidas eemaldada veerge?  
Kasutage miinusmärki:

subset(tabel, select = -c(vanus, elukoht)) # eemaldab veerud vanus ja elukoht  
tabel[,1] # eemaldab esimese veeru  
tabel[-1,] # eemaldab esimese rea

## Ridade eraldamine

Järjekorra numbri abil saab ridasid samamoodi eraldada nagu veerge:

tabel[1,] #esimene rida

tabel[1:5,] #esimesed viis rida

## Tingimuste kasutamine

Kuidas valida ridu, mis vastavad teatud tingimustele? Proovime näiteks võtta andmestikust need read, kus vaadeldava isiku vanus on alla 30.

Nurgeliste sulgudega:

tabel[tabel$vanus<30,]

Funktsiooniga *subset*:

subset(tabel, vanus < 30)

Vahepeal on vaja võtta andmetest välja read, mis on sarnase väärtusega. Näiteks püüame eraldada andmetest kõik read, kus elukoha väärtus on “maa”. Saame kasutada juba tuttavat ridade ja veergude määratlust:

tabel[tabel$elukoht == "maa",] # võtab andmetest ainult need read, kus elukoht on võrdne väärtusega "maa"

Samasuguse tulemuse saame ka funktsiooniga *subset*

subset(tabel, elukoht == "maa")

## Ülesanded - andmetega manipuleerimine

1. Eraldage andmetest veerg “vanus”. Looge uus muutuja, kus on ainult see veerg.

vanus <- tabel$vanus

1. Tehke kaks uut andmestikku. Esimesse salvestage maal elavate katseisikute tulemused ja teise salvestage linnas elavate katseisikute tulemused.

tabel\_maa <- subset(tabel, elukoht == "maa")  
tabel\_linn <- subset(tabel, elukoht == "linn")

# 4. Kirjeldav Statistika

Nüüd, kui teame, kuidas anda R’ile edasi ainult üks muutuja andmestikust, proovime saada selgemat ülevaadet muutujast “vanus”.  
**Keskmise** vanuse saame, kui anname funktsioonile *mean* argumendiks vastava muutuja nime:

mean(tabel$vanus)

## [1] 44.19

**Mediaani** saame funktsiooni *median* abil:

median(tabel$vanus)

## [1] 46

**Standardhälve**

sd(tabel$vanus)

## [1] 14.61478

**Miinimum** ja **maksimum**

min(tabel$vanus)

## [1] 18

max(tabel$vanus)

## [1] 70

**Puuduvad väärtused**  
Kui veerus esineb puuduvaid väärtusi, siis annab R meile statistiku väärtuseks samuti puuduva väärtuse ehk NA (*not available*):

mean(tabel$sotsiaalsus)

## [1] NA

Selleks, et puuduvad väärtused arvutustest välja jätta tuleb kirjeldava statistika funktsioonidele ette anda täiendav argument “na.rm=TRUE”.

mean(tabel$sotsiaalsus, na.rm=TRUE)

## [1] 126.0201

median(tabel$sotsiaalsus, na.rm=TRUE)

## [1] 126

Kategooriaid sisaldava tunnuse kirjeldamisel on abiks **sagedustabel**:

table(tabel$elukoht)

##   
## maa linn   
## 87 113

Protsentuaalse jaotuse saame, kui lisame sagedustabeli ümber funktsiooni *prop.table*:

prop.table(table(tabel$elukoht))

##   
## maa linn   
## 0.435 0.565

## Ülesanded - kirjeldav statistika

1. Leidke sotsiaalsuse, meelekindluse, ekstravertsuse, neurootilisuse ja avatuse keskmised tulemused.

mean(tabel$sotsiaalsus, na.rm=TRUE)

## [1] 126.0201

mean(tabel$meelekindlus, na.rm=TRUE)

## [1] 113.5707

mean(tabel$ekstravertsus)

## [1] 102.88

mean(tabel$neurootilisus)

## [1] 87.89

1. Kasutage funktsiooni *summary()*. Andke argumendiks üks veerg. Andke argumendiks terve tabel. Mida see funktsioon väljastab?

summary(tabel$meelekindlus)

## Min. 1st Qu. Median Mean 3rd Qu. Max. NA's   
## 53.00 99.25 114.00 113.57 127.75 178.00 2

summary(tabel)

## vanus elukoht sotsiaalsus meelekindlus   
## Min. :18.00 maa : 87 Min. : 74.0 Min. : 53.00   
## 1st Qu.:32.00 linn:113 1st Qu.:114.5 1st Qu.: 99.25   
## Median :46.00 Median :126.0 Median :114.00   
## Mean :44.19 Mean :126.0 Mean :113.57   
## 3rd Qu.:56.25 3rd Qu.:136.0 3rd Qu.:127.75   
## Max. :70.00 Max. :167.0 Max. :178.00   
## NA's :1 NA's :2   
## ekstravertsus neurootilisus avatus depressioon   
## Min. : 8.00 Min. : 34.00 Min. : 73.0 Min. : 0.000   
## 1st Qu.: 88.75 1st Qu.: 70.00 1st Qu.:110.0 1st Qu.: 3.000   
## Median :105.00 Median : 89.50 Median :124.5 Median : 6.000   
## Mean :102.88 Mean : 87.89 Mean :123.1 Mean : 7.015   
## 3rd Qu.:118.25 3rd Qu.:104.00 3rd Qu.:136.2 3rd Qu.: 9.000   
## Max. :168.00 Max. :152.00 Max. :173.0 Max. :27.000   
##   
## traitanx stateanx   
## Min. :22.00 Min. :21   
## 1st Qu.:31.00 1st Qu.:32   
## Median :38.00 Median :38   
## Mean :39.08 Mean :40   
## 3rd Qu.:45.00 3rd Qu.:48   
## Max. :71.00 Max. :79   
## NA's :1

1. Millises vahemikus varieeruvad ekstravertsuse skoorid?

range(tabel$ekstravertsus)

## [1] 8 168

#või:  
min(tabel$ekstravertsus)

## [1] 8

max(tabel$ekstravertsus)

## [1] 168