

Operators in R

adding 2 vectors

```
v1<-c(50,60)
v2<-c(60,23)
print(v1+v2)
```

```
## [1] 110 83
```

multiplying 2 vectors

```
print(v1*v2)
```

```
## [1] 3000 1380
```

remainder of 2 vectors

```
print(v1%%v2)
```

```
## [1] 50 14
```

dividing 2 vectors

```
print(v1/v2)
```

```
## [1] 0.8333333 2.6086957
```

quotient of 2 vectors

```
print(v1%/%v2)
```

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

creating sequential series

```
series<- 200:300
print(series)
```

```
## [1] 200 201 202 203 204 205 206 207 208 209 210 211 212 213 214 215 216 217
## [19] 218 219 220 221 222 223 224 225 226 227 228 229 230 231 232 233 234 235
## [37] 236 237 238 239 240 241 242 243 244 245 246 247 248 249 250 251 252 253
## [55] 254 255 256 257 258 259 260 261 262 263 264 265 266 267 268 269 270 271
## [73] 272 273 274 275 276 277 278 279 280 281 282 283 284 285 286 287 288 289
## [91] 290 291 292 293 294 295 296 297 298 299 300
```

identifying the value if it present in the series

```
print(156 %in% series)
```

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

```
print(v1 %in% series)
```

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

```
print(289 %in% series)
```

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

```
print(v2+200 %in% series)
```

```
## [1] 61 24
```

$A \cdot A^T$ (matrix A * (matrix A)^t)

```
M = matrix( c(734,456,2345,657), nrow=2,ncol=2,byrow = TRUE)
print(M)
```

```
##      [,1] [,2]
## [1,]  734  456
## [2,] 2345  657
```

```
t = M %*% t(M)
print(t)
```

```
##      [,1] [,2]
## [1,] 746692 2020822
## [2,] 2020822 5930674
```

Control Statements

If Statements

```
if(250%in%series)
{
print("Yes it is present in the given series")
}else
{
print("NA")
}
```

```
## [1] "Yes it is present in the given series"
```

LOOPS

While Loop

```
i<-2
while(i)
{
  print(i+1)
  i<-i+1
  if(i==20)
  {
    break
  }
}
```

```
## [1] 3
## [1] 4
## [1] 5
## [1] 6
## [1] 7
## [1] 8
## [1] 9
## [1] 10
## [1] 11
## [1] 12
## [1] 13
## [1] 14
## [1] 15
## [1] 16
## [1] 17
## [1] 18
## [1] 19
## [1] 20
```

Repeat Loops

```
v <- c(999)
count <- 2
repeat{
  print(v)
  count <- count+1
  if(count > 5)
  { break }
}
```

```
## [1] 999
## [1] 999
## [1] 999
## [1] 999
```

FOR loop

```
series<- 1:10
for(k in series)
{
  print(k)
}
```

```
## [1] 1
## [1] 2
## [1] 3
## [1] 4
## [1] 5
## [1] 6
## [1] 7
## [1] 8
## [1] 9
## [1] 10
```

Loop Control Statements

Break

```
v <- letters[1:7]
for ( i in v)
{
  if(i == "c")
  {
    break ## breaks at the point letter C
  }
  print(i)
}
```

```
## [1] "a"
## [1] "b"
```

Next

```
v <- letters[1:7]
for ( i in v)
{
  if(i == "c")
  {
    next ## skips the letter C
  }
  print(i)
}
```

```
## [1] "a"
## [1] "b"
## [1] "d"
## [1] "e"
## [1] "f"
## [1] "g"
```

Importing Data

```
data <- mtcars
print(data)
```

```
##           mpg  cyl  disp  hp drat    wt  qsec vs  am gear carb
## Mazda RX4      21.0    6 160.0 110 3.90 2.620 16.46 0  1    4    4
## Mazda RX4 Wag  21.0    6 160.0 110 3.90 2.875 17.02 0  1    4    4
## Datsun 710     22.8    4 108.0  93 3.85 2.320 18.61 1  1    4    1
## Hornet 4 Drive  21.4    6 258.0 110 3.08 3.215 19.44 1  0    3    1
## Hornet Sportabout 18.7    8 360.0 175 3.15 3.440 17.02 0  0    3    2
## Valiant        18.1    6 225.0 105 2.76 3.460 20.22 1  0    3    1
## Duster 360     14.3    8 360.0 245 3.21 3.570 15.84 0  0    3    4
## Merc 240D      24.4    4 146.7  62 3.69 3.190 20.00 1  0    4    2
## Merc 230       22.8    4 140.8  95 3.92 3.150 22.90 1  0    4    2
## Merc 280       19.2    6 167.6 123 3.92 3.440 18.30 1  0    4    4
## Merc 280C      17.8    6 167.6 123 3.92 3.440 18.90 1  0    4    4
## Merc 450SE     16.4    8 275.8 180 3.07 4.070 17.40 0  0    3    3
## Merc 450SL     17.3    8 275.8 180 3.07 3.730 17.60 0  0    3    3
## Merc 450SLC    15.2    8 275.8 180 3.07 3.780 18.00 0  0    3    3
## Cadillac Fleetwood 10.4    8 472.0 205 2.93 5.250 17.98 0  0    3    4
## Lincoln Continental 10.4    8 460.0 215 3.00 5.424 17.82 0  0    3    4
## Chrysler Imperial 14.7    8 440.0 230 3.23 5.345 17.42 0  0    3    4
## Fiat 128       32.4    4  78.7  66 4.08 2.200 19.47 1  1    4    1
## Honda Civic    30.4    4  75.7  52 4.93 1.615 18.52 1  1    4    2
## Toyota Corolla 33.9    4  71.1  65 4.22 1.835 19.90 1  1    4    1
## Toyota Corona  21.5    4 120.1  97 3.70 2.465 20.01 1  0    3    1
## Dodge Challenger 15.5    8 318.0 150 2.76 3.520 16.87 0  0    3    2
## AMC Javelin    15.2    8 304.0 150 3.15 3.435 17.30 0  0    3    2
## Camaro Z28     13.3    8 350.0 245 3.73 3.840 15.41 0  0    3    4
## Pontiac Firebird 19.2    8 400.0 175 3.08 3.845 17.05 0  0    3    2
## Fiat X1-9      27.3    4  79.0  66 4.08 1.935 18.90 1  1    4    1
## Porsche 914-2  26.0    4 120.3  91 4.43 2.140 16.70 0  1    5    2
## Lotus Europa   30.4    4  95.1 113 3.77 1.513 16.90 1  1    5    2
## Ford Pantera L 15.8    8 351.0 264 4.22 3.170 14.50 0  1    5    4
## Ferrari Dino   19.7    6 145.0 175 3.62 2.770 15.50 0  1    5    6
## Maserati Bora  15.0    8 301.0 335 3.54 3.570 14.60 0  1    5    8
## Volvo 142E     21.4    4 121.0 109 4.11 2.780 18.60 1  1    4    2
```

Summary

```
summary(data)
```

```
##      mpg      cyl      disp      hp
## Min.   :10.40  Min.   :4.000  Min.   : 71.1  Min.   : 52.0
## 1st Qu.:15.43  1st Qu.:4.000  1st Qu.:120.8  1st Qu.: 96.5
## Median :19.20  Median :6.000  Median :196.3  Median :123.0
## Mean   :20.09  Mean   :6.188  Mean   :230.7  Mean   :146.7
## 3rd Qu.:22.80  3rd Qu.:8.000  3rd Qu.:326.0  3rd Qu.:180.0
## Max.   :33.90  Max.   :8.000  Max.   :472.0  Max.   :335.0
##      drat      wt      qsec      vs
## Min.   :2.760  Min.   :1.513  Min.   :14.50  Min.   :0.0000
## 1st Qu.:3.080  1st Qu.:2.581  1st Qu.:16.89  1st Qu.:0.0000
## Median :3.695  Median :3.325  Median :17.71  Median :0.0000
## Mean   :3.597  Mean   :3.217  Mean   :17.85  Mean   :0.4375
## 3rd Qu.:3.920  3rd Qu.:3.610  3rd Qu.:18.90  3rd Qu.:1.0000
## Max.   :4.930  Max.   :5.424  Max.   :22.90  Max.   :1.0000
##      am      gear      carb
## Min.   :0.0000  Min.   :3.000  Min.   :1.000
## 1st Qu.:0.0000  1st Qu.:3.000  1st Qu.:2.000
## Median :0.0000  Median :4.000  Median :2.000
## Mean   :0.4062  Mean   :3.688  Mean   :2.812
## 3rd Qu.:1.0000  3rd Qu.:4.000  3rd Qu.:4.000
## Max.   :1.0000  Max.   :5.000  Max.   :8.000
```