Exercises, Chapter 2

Exercise 2.1

```
100 * ( 1 + (0.05 / 12) )^24
```

```
## [1] 110.4941
```

Exercise 2.2

```
5 %% 2
```

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

Exercise 2.3

```
3333 %% 222
```

```
## [1] 3
```

Exercise 2.4

```
domainValues<-10^(c(1:10))
# avoid scientific notation
options(scipen=1000)
# increase significant digits
options(digits=22)
# applying formula
rangeValues<- ( 1 + 1/domainValues)^domainValues
# force output to a single column of values
options(width=40)
rangeValues
```

```
##  [1] 2.5937424601000023116630
##  [2] 2.7048138294215284815890
##  [3] 2.7169239322355935861710
##  [4] 2.7181459268249255067930
##  [5] 2.7182682371922974873260
##  [6] 2.7182804690957533821920
##  [7] 2.7182816941320817605290
##  [8] 2.7182817983473577250210
##  [9] 2.7182820520115602569430
## [10] 2.7182820532347875541750
```

Exercises, Chapter 3

Exercise 3.1

```
D<-1000
K<-5
h<-0.25
# implementing square root as an exponent: raised to 1/2
Q<-((2*D*K)/h)^(0.5)
Q
```

```
## [1] 200
```

```
ls()
```

```
## [1] "D"            "domainValues"
## [3] "h"            "K"
## [5] "Q"            "rangeValues"
```

Exercise 3.2

```
P<-100; r<-0.08; n<-12; t<-3
F<-P*(1+r/n)^(n*t)
F
```

```
## [1] 127.0237051620650703398
```

Exercises, Chapter 4

Exercise 4.1

```
desiredVector<-rep(c(2.7,8,3),2)
desiredVector
```

```
## [1] 2.700000000000000177636
## [2] 8.000000000000000000000
## [3] 3.000000000000000000000
## [4] 2.700000000000000177636
## [5] 8.000000000000000000000
## [6] 3.000000000000000000000
```

Exercise 4.2

```
desiredVector<-seq(0,2,by=.4)
desiredVector
```

```
## [1] 0.000000000000000000000
## [2] 0.400000000000000222045
## [3] 0.800000000000000444089
## [4] 1.200000000000001776357
```

```
## [5] 1.600000000000000888178
## [6] 2.000000000000000000000
```

Exercise 4.3

```
primes      <-  c(2,3,5,7,11,13,17,19,23,29)
composites  <-  c(4,6,8,9,10)
primes[composites]
```

```
## [1]  7 13 19 23 29
```

Exercise 4.4

```
seq(3, 28, by=11) %/% 4
```

```
## [1] 0 3 6
```

Exercise 4.5

```
seq(0, 2, length.out=5)
```

```
## [1] 0.0 0.5 1.0 1.5 2.0
```

Exercise 4.6

```
x<-c(2,0,-5,-7)
x
```

```
## [1]  2  0 -5 -7
```

```
# negative index means show all in x except for index
# here -2.8, which is truncated to -2,
# therefore, show all of x except for the second element
x[-2.8]
```

```
## [1]  2 -5 -7
```

Exercise 4.7

```
rep(0:2,1:3)
```

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

```
3 ^ rep(0:2,1:3)
```

```
## [1] 1 3 3 9 9 9
```

Exercise 4.8

```
seed <- rep(1:4)
desiredVector <- c(seed,1+seed,2+seed,3+seed)
desiredVector
```

```
##  [1] 1 2 3 4 2 3 4 5 3 4 5 6 4 5 6 7
```

```
# also
rep(1:4,4)+c(rep(0,4),rep(1,4),rep(2,4),rep(3,4))
```

```
##  [1] 1 2 3 4 2 3 4 5 3 4 5 6 4 5 6 7
```

Exercise 4.9

```
0:(4/.05)*.05
```

```
##  [1] 0.00000000000000000000000
##  [2] 0.05000000000000000277556
##  [3] 0.10000000000000000555112
##  [4] 0.15000000000000002220446
##  [5] 0.20000000000000001110223
##  [6] 0.25000000000000000000000
##  [7] 0.30000000000000004440892
##  [8] 0.35000000000000003330669
##  [9] 0.40000000000000002220446
## [10] 0.45000000000000001110223
## [11] 0.50000000000000000000000
## [12] 0.55000000000000004440892
## [13] 0.60000000000000008881784
## [14] 0.65000000000000002220446
## [15] 0.70000000000000006661338
## [16] 0.75000000000000000000000
## [17] 0.80000000000000004440892
## [18] 0.85000000000000008881784
## [19] 0.90000000000000002220446
## [20] 0.95000000000000006661338
## [21] 1.00000000000000000000000
## [22] 1.05000000000000004440892
## [23] 1.10000000000000008881784
## [24] 1.15000000000000013322676
## [25] 1.20000000000000017763568
## [26] 1.25000000000000000000000
## [27] 1.30000000000000004440892
## [28] 1.35000000000000008881784
## [29] 1.40000000000000013322676
## [30] 1.45000000000000017763568
## [31] 1.50000000000000000000000
## [32] 1.55000000000000004440892
## [33] 1.60000000000000008881784
## [34] 1.65000000000000013322676
## [35] 1.70000000000000017763568
## [36] 1.75000000000000000000000
## [37] 1.80000000000000004440892
```

```
## [38] 1.850000000000000008881784
## [39] 1.900000000000000013322676
## [40] 1.950000000000000017763568
## [41] 2.000000000000000000000000
## [42] 2.050000000000000026645353
## [43] 2.100000000000000008881784
## [44] 2.149999999999999118216
## [45] 2.200000000000000017763568
## [46] 2.250000000000000000000000
## [47] 2.300000000000000026645353
## [48] 2.350000000000000008881784
## [49] 2.400000000000000035527137
## [50] 2.450000000000000017763568
## [51] 2.500000000000000000000000
## [52] 2.550000000000000026645353
## [53] 2.600000000000000008881784
## [54] 2.650000000000000035527137
## [55] 2.700000000000000017763568
## [56] 2.750000000000000000000000
## [57] 2.800000000000000026645353
## [58] 2.850000000000000008881784
## [59] 2.900000000000000035527137
## [60] 2.950000000000000017763568
## [61] 3.000000000000000000000000
## [62] 3.050000000000000026645353
## [63] 3.100000000000000008881784
## [64] 3.150000000000000035527137
## [65] 3.200000000000000017763568
## [66] 3.250000000000000000000000
## [67] 3.300000000000000026645353
## [68] 3.350000000000000008881784
## [69] 3.400000000000000035527137
## [70] 3.450000000000000017763568
## [71] 3.500000000000000000000000
## [72] 3.550000000000000026645353
## [73] 3.600000000000000008881784
## [74] 3.650000000000000035527137
## [75] 3.700000000000000017763568
## [76] 3.750000000000000000000000
## [77] 3.800000000000000026645353
## [78] 3.850000000000000008881784
## [79] 3.900000000000000035527137
## [80] 3.950000000000000017763568
## [81] 4.000000000000000000000000
```

Exercise 4.10

```
x <- seq(1:8)
x
```

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

```
x[6:8]
```

```
## [1] 6 7 8
```

```
x[c(6:8)]
```

```
## [1] 6 7 8
```

```
x[-c(-6:-8)]
```

```
## [1] 6 7 8
```

Exercise 4.11

```
(1:10)^(1:10)
```

```
##  [1]           1           4          27
##  [4]         256        3125       46656
##  [7]      823543    16777216   387420489
## [10] 10000000000
```

Exercises taken from Chapters 2, 3, and 4 of "Learning Base R", by Lawrence M Leemis, ISBN 978-0-9829174-8-0