I don’t have MATLAB installed, and I’m better with R, so I’m using R.

library("tidyverse")

## 1

B <-  
 matrix(  
 data = c(  
 seq(1L, 7L, 1L),  
 seq(9L, -3L, -2L),  
 2L ^ seq(2L, 8L, 1L)  
 ),  
 nrow = 3, ncol = 7, byrow = TRUE  
 )  
B

## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 1 2 3 4 5 6 7  
## [2,] 9 7 5 3 1 -1 -3  
## [3,] 4 8 16 32 64 128 256

## 2

str <- "Need-to-break-this-string"  
str %>% str\_split\_fixed("-", n = 5)

## [,1] [,2] [,3] [,4] [,5]   
## [1,] "Need" "to" "break" "this" "string"

## 3

With R, (assuming "data.csv" exists in the working directory)

# R  
readr::read\_csv("data.csv", col\_names = TRUE)

With MATLAB,

dlmread("data.csv", ",")

## 4

a <- t(c(1, 2, 3))  
b <- t(c(2, 4, 7))  
a / b

## [,1] [,2] [,3]  
## [1,] 0.5 0.5 0.4285714

b / a

## [,1] [,2] [,3]  
## [1,] 2 2 2.333333

## 5

I’ll assume “matrix A” is matrix B from #1.

B[1, 3] <- 0

## 6

With R,

exists("a.b.c")  
# Or...

With MATLAB,

ismember(a,b,c)

## 7

a <- matrix(  
 data =  
 c(  
 c(1, 4, 3, 2),  
 c(-3, 4, -5, 1),  
 c(5, 6, 1, 2),  
 c(-2, -3, 5, 6)  
 ),  
 nrow = 4, ncol = 4, byrow = TRUE  
)  
c <- a[c(1:3), c(2:3)]  
c

## [,1] [,2]  
## [1,] 4 3  
## [2,] 4 -5  
## [3,] 6 1

s <- sum(a)  
s

## [1] 27

d <- a \* (a \* diag(a))  
d

## [,1] [,2] [,3] [,4]  
## [1,] 1 16 9 4  
## [2,] 36 64 100 4  
## [3,] 25 36 1 4  
## [4,] 24 54 150 216

e <- a[c(3:4), c(3:4)] \* t(a[c(3:4), c(3:4)])  
e

## [,1] [,2]  
## [1,] 1 10  
## [2,] 10 36

## 8

TRUE (because NOT(4 == 5 & 4 == 4) -> NOT(FALSE & TRUE) -> NOT(FALSE) -> TRUE)

## 9

# v <- seq(2, 2 \* 30, by = 2)  
# v %\*% v  
(2 ^ 30) \* factorial(30)

## [1] 2.848131e+41

## 10

z = 4

## 11

twice

## 12

1 'men'  
2 'mens'  
3 'mensa'  
4  
5  
6  
7  
8  
9  
10

## 13

## 14

5

## 15

m <- matrix(0, 5, 5)  
for (i in (1:5)) {  
 for (j in (1:5)) {  
 m[i, j] <- i + j  
 m[j, i] <- m[i, j]  
 }  
}  
m[4, 3]

## [1] 7

m[4, 3] = 7

## 16

k <- 0  
while((k ^ 0.5) < k) {  
 k <- k + 1  
}  
k

## [1] 0

k = 0

## 17

## 18

mask1 <- matrix(0, 3, 7)  
mask1[, 3] <- 1  
mask1[, 5] <- 1  
mask1

## [,1] [,2] [,3] [,4] [,5] [,6] [,7]  
## [1,] 0 0 1 0 1 0 0  
## [2,] 0 0 1 0 1 0 0  
## [3,] 0 0 1 0 1 0 0

sum((mask1 %\*% t(B))[1, ])

## [1] 94

mask2 <- matrix(0, 7, 3)  
mask2[, 3] <- 1  
mask2

## [,1] [,2] [,3]  
## [1,] 0 0 1  
## [2,] 0 0 1  
## [3,] 0 0 1  
## [4,] 0 0 1  
## [5,] 0 0 1  
## [6,] 0 0 1  
## [7,] 0 0 1

sum(t(B) \* mask2)

## [1] 508

# ...

## 19

## 20

# Part II

## 1

select \*   
from VetVisit a  
left join VetVisit b  
on (select PetName, max(VisitNumber) mx   
from VetVisit group by PetName) b  
where a.VisitNumber = b.mx

## 2

Doing this with row\_number() (which could be used for the previous question as well),

with temp as  
(select PetName,  
row\_number() over (partition by PetName order by VisitNumber desc) idx  
from VetVisit)  
  
select \*  
where idx = 2