Handy Vector Functions

Introduction to R

Here we define some vectors:

```
set.seed(1201)
u <- sample(100,100,replace=TRUE)
v <- sample(100,100,replace=TRUE)
l <- list("x"=sample(1:10,5),"y"=sample(11:20,5))
df <- data.frame("x"=sample(1:10,5),"y"=sample(11:20,5))
x <- c(1,2,3,4)
y <- c(-2,2,-3,3)
z <- c(-5,1,2,-4,3,4,-3,6)
```

Question 1

Display the list `l` as a numerical vector, with names associated with each element.

```
unlist(l)
```

```
## x1 x2 x3 x4 x5 y1 y2 y3 y4 y5
##  6  8  7  3  9 15 16 17 18 12
```

Question 2

Display the list `l` as a numerical vector, while stripping away the names seen in Q1.

```
as.vector(unlist(l))
```

```
##  [1]  6  8  7  3  9 15 16 17 18 12
```

Question 3

Repeat Q2, but display the vector in *descending* order.

```
sort(as.vector(unlist(l)),decreasing=TRUE)
```

```
##  [1] 18 17 16 15 12  9  8  7  6  3
```

```
# or
rev(sort(as.vector(unlist(l))))
```

```
##  [1] 18 17 16 15 12  9  8  7  6  3
```

Question 4

Here are the contents of the data frame `df`:

```
df
```

```
##   x  y
## 1 8 12
## 2 4 16
## 3 3 19
## 4 6 17
## 5 9 15
```

Reorder the rows so that the entries of the `x` column are in numerical order and the association between the ith entry of `x` and the ith entry of `y` is not lost. Display the result.

```
o <- order(df$x)
df[o,]
```

```
##   x  y
## 3 3 19
## 2 4 16
## 4 6 17
## 1 8 12
## 5 9 15
```

Question 5

Display the proportion of the total number of unique values in `u` to the number of values in `u`.

```
length(unique(u))/length(u)
```

```
## [1] 0.62
```

Question 6

Display a table that shows how often each value of `v` appears.

```
table(v)
```

```
## v
##  2  4  5  7 11 12 13 14 15 16 17 22 23 26 27 28 29 30 33 35 36 37 40 41 42 43
##  5  1  1  1  2  1  3  1  3  1  2  1  1  3  5  1  1  2  3  1  1  1  1  1  5  1
## 45 46 47 49 50 52 54 57 58 61 62 64 65 66 67 73 74 77 78 79 80 81 84 87 88 92
##  1  2  1  3  2  2  1  3  1  1  1  2  2  1  3  2  2  1  1  2  2  1  3  1  1  2
## 93 94 95 96 98
##  1  1  2  1  2
```

Question 7

How many unique values do `u` and `v` have in common?

```
length(intersect(u,v))
```

```
## [1] 39
```

Question 8

Write down an expression that returns `TRUE` if the union of `u` and `v` has 100 elements and `FALSE` otherwise.

```
length(union(u,v)) == 100
```

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

Question 9

Display the (sorted!) values of `u` that do not appear in `v` .

```
sort(setdiff(u,v))
```

```
##  [1]   1   3   9  10  19  20  24  34  38  39  44  55  60  63  70  72  75  82  85
## [20]  89  97  99 100
```

Question 10

Display a table showing how many values that are in `v` but not in `u` fall into the bins [1,50] and [51,100].

```
table(findInterval(setdiff(v,u),c(51)))
```

```
## 
## 0 1 
## 9 9
```