Name: Looi Kah Fung

Matric no: P-COM0049/22

Date: 31st Oct 2022

Course: Lab 1 CDS502 Principles & Practices of Data Science and Analytics

```
> phrase <- c("I","don't","know","I","know")
> phrase[1:3]
[1] "I"     "don't" "know"
> phrase[3:5]
[1] "know" "I"    "know"
```

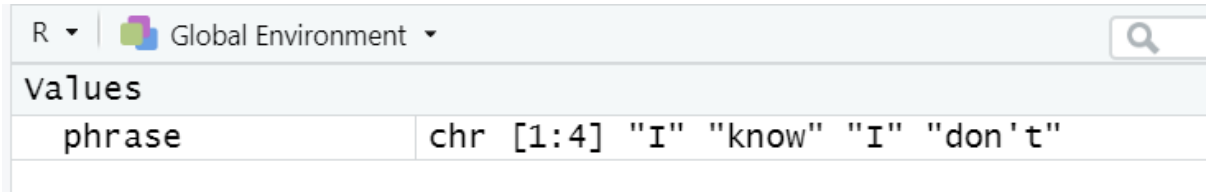
1. What is phrase[-2]? What is phrase[-5]? Given those answers, explain what phrase[-1:- 3] does.

Phrase[-1:-3] uses a slicing method to remove the 1st 3 elements in the vector.

```
> phrase[-2]
[1] "I"    "know" "I"    "know"
> phrase[-5]
[1] "I"     "don't" "know"  "I"
> phrase[-1:-3]
[1] "I"    "know"
```

2. Use indexing of phrase to create a new character vector that forms the phrase "I know I don't", i.e. c("I", "know", "I", "don't").

```
> phrase <- c("I", "know","I","don't")
```




3. Use sum to calculate the summation from 1 to 5 (ie 1+2+3+4+5).

```
> sum(1:5)
[1] 15
```

4. Use sum to calculate the summation from 1 to 10,000.

```
> sum(1:10000)
[1] 50005000
```

5. What does seq do?

```
> help("seq")
```

Descriptive information at the R documentation by using the help function to search out the usage of the seq{base}. Generate regular sequences. seq is a standard generic with a default method. seq.int is a primitive which can be much faster but has a few restrictions. seq_along and seq_len are very fast primitives for two common cases.

seq(from, to)

seq(from, to, by= )

seq(from, to, length.out= )

seq(along.with= )

seq(from)

seq(length.out= )

```
> seq(1,10)
 [1]  1  2  3  4  5  6  7  8  9 10
> seq(1,10, by=2)
[1] 1 3 5 7 9
> seq(1,10,length.out=7)
[1]  1.0  2.5  4.0  5.5  7.0  8.5 10.0
> seq(along.with=10)
[1] 1
> seq(10)
 [1]  1  2  3  4  5  6  7  8  9 10
> seq(length.out=10)
 [1]  1  2  3  4  5  6  7  8  9 10
```