

Programming in Base R

Task 1: Basic Vector Practice

Question 1

```
# pre vector
pre <- c(130, 128, 116, 124, 133, 134, 118, 126, 114, 127,
        141, 138, 128, 140, 137, 131, 120, 128, 139, 135)

# post vector
post<- c(114, 98, 113, 99, 107, 116, 113, 111, 119, 117,
        101, 119, 130, 122, 106, 106, 124, 102, 117, 113)
```

Question 2

```
names <- paste("Subject", 1:20, sep = "_")

names(pre) <- names
pre
```

Subject_1	Subject_2	Subject_3	Subject_4	Subject_5	Subject_6	Subject_7
130	128	116	124	133	134	118
Subject_8	Subject_9	Subject_10	Subject_11	Subject_12	Subject_13	Subject_14
126	114	127	141	138	128	140
Subject_15	Subject_16	Subject_17	Subject_18	Subject_19	Subject_20	
137	131	120	128	139	135	

```
names(post) <- names
post
```

Subject_1	Subject_2	Subject_3	Subject_4	Subject_5	Subject_6	Subject_7
114	98	113	99	107	116	113
Subject_8	Subject_9	Subject_10	Subject_11	Subject_12	Subject_13	Subject_14
111	119	117	101	119	130	122
Subject_15	Subject_16	Subject_17	Subject_18	Subject_19	Subject_20	
106	106	124	102	117	113	

Question 3

```
diff_op <- (pre - post)

diff_op
```

Subject_1	Subject_2	Subject_3	Subject_4	Subject_5	Subject_6	Subject_7
16	30	3	25	26	18	5
Subject_8	Subject_9	Subject_10	Subject_11	Subject_12	Subject_13	Subject_14
15	-5	10	40	19	-2	18
Subject_15	Subject_16	Subject_17	Subject_18	Subject_19	Subject_20	
31	25	-4	26	22	22	

Question 4

```
mean_bp_dec <- mean(diff_op)
mean_bp_dec
```

[1] 17

Question 5

```
pos_change <- which(diff_op > 0)
pos_change
```

Subject_1	Subject_2	Subject_3	Subject_4	Subject_5	Subject_6	Subject_7
1	2	3	4	5	6	7
Subject_8	Subject_10	Subject_11	Subject_12	Subject_14	Subject_15	Subject_16
8	10	11	12	14	15	16
Subject_18	Subject_19	Subject_20				
18	19	20				

Question 6

```
diff_op1 <- diff_op[diff_op > 0]
diff_op1
```

Subject_1	Subject_2	Subject_3	Subject_4	Subject_5	Subject_6	Subject_7
16	30	3	25	26	18	5
Subject_8	Subject_10	Subject_11	Subject_12	Subject_14	Subject_15	Subject_16
15	10	40	19	18	31	25
Subject_18	Subject_19	Subject_20				
26	22	22				

Question 7

```
mean_pos_change <- mean(diff_op1)
mean_pos_change
```

[1] 20.64706

Task 2: Basic data Frame Practice

Question 1

```
bp_data <- data.frame(patient = names, pre_bp = pre, post_bp = post, diff_bp = diff_op)
```

Question 2

```
bp_data[bp_data$diff_bp<0, ]
```

	patient	pre_bp	post_bp	diff_bp
Subject_9	Subject_9	114	119	-5
Subject_13	Subject_13	128	130	-2
Subject_17	Subject_17	120	124	-4

Question 3

```
bp_data$post_less120 <- bp_data$post_bp<120  
bp_data
```

	patient	pre_bp	post_bp	diff_bp	post_less120
Subject_1	Subject_1	130	114	16	TRUE
Subject_2	Subject_2	128	98	30	TRUE
Subject_3	Subject_3	116	113	3	TRUE
Subject_4	Subject_4	124	99	25	TRUE
Subject_5	Subject_5	133	107	26	TRUE
Subject_6	Subject_6	134	116	18	TRUE
Subject_7	Subject_7	118	113	5	TRUE
Subject_8	Subject_8	126	111	15	TRUE
Subject_9	Subject_9	114	119	-5	TRUE
Subject_10	Subject_10	127	117	10	TRUE
Subject_11	Subject_11	141	101	40	TRUE
Subject_12	Subject_12	138	119	19	TRUE
Subject_13	Subject_13	128	130	-2	FALSE
Subject_14	Subject_14	140	122	18	FALSE
Subject_15	Subject_15	137	106	31	TRUE
Subject_16	Subject_16	131	106	25	TRUE
Subject_17	Subject_17	120	124	-4	FALSE
Subject_18	Subject_18	128	102	26	TRUE
Subject_19	Subject_19	139	117	22	TRUE
Subject_20	Subject_20	135	113	22	TRUE

Question 4

```
knitr::kable(bp_data)
```

Warning in attr(x, "align"): 'xfun::attr()' is deprecated.
Use 'xfun::attr2()' instead.
See help("Deprecated")

Warning in attr(x, "format"): 'xfun::attr()' is deprecated.
Use 'xfun::attr2()' instead.
See help("Deprecated")

	patient	pre_bp	post_bp	diff_bp	post_less120
Subject_1	Subject_1	130	114	16	TRUE
Subject_2	Subject_2	128	98	30	TRUE
Subject_3	Subject_3	116	113	3	TRUE
Subject_4	Subject_4	124	99	25	TRUE
Subject_5	Subject_5	133	107	26	TRUE
Subject_6	Subject_6	134	116	18	TRUE
Subject_7	Subject_7	118	113	5	TRUE
Subject_8	Subject_8	126	111	15	TRUE
Subject_9	Subject_9	114	119	-5	TRUE
Subject_10	Subject_10	127	117	10	TRUE
Subject_11	Subject_11	141	101	40	TRUE
Subject_12	Subject_12	138	119	19	TRUE
Subject_13	Subject_13	128	130	-2	FALSE
Subject_14	Subject_14	140	122	18	FALSE
Subject_15	Subject_15	137	106	31	TRUE
Subject_16	Subject_16	131	106	25	TRUE
Subject_17	Subject_17	120	124	-4	FALSE
Subject_18	Subject_18	128	102	26	TRUE
Subject_19	Subject_19	139	117	22	TRUE
Subject_20	Subject_20	135	113	22	TRUE

Task 3: List Practice

Question 1

```
subject_IDs <- paste("Subject", 1:10, sep = "_")
pre_placebo <- c(138, 135, 147, 117, 152, 134, 114, 121, 131, 130)
post_placebo <- c(105, 136, 123, 130, 134, 143, 135, 139, 120, 124)
diff <- pre_placebo - post_placebo
```

```
bp_df_placebo <- data.frame(subject_ID = subject_IDs, pre = pre_placebo, post = post_pla
bp_df_placebo
```

	subject_ID	pre	post	diff
1	Subject_1	138	105	33
2	Subject_2	135	136	-1
3	Subject_3	147	123	24
4	Subject_4	117	130	-13
5	Subject_5	152	134	18
6	Subject_6	134	143	-9
7	Subject_7	114	135	-21
8	Subject_8	121	139	-18
9	Subject_9	131	120	11
10	Subject_10	130	124	6

```
bp_df_placebo$post_below_120 <- bp_df_placebo$post<120
```

Question 2

```
bp_list <- list(treatment = bp_data, placebo = bp_df_placebo)
```

Question 3

```
#Access trt using 3 ways
#1
bp_list[[1]]
```

	patient	pre_bp	post_bp	diff_bp	post_less120
Subject_1	Subject_1	130	114	16	TRUE
Subject_2	Subject_2	128	98	30	TRUE
Subject_3	Subject_3	116	113	3	TRUE
Subject_4	Subject_4	124	99	25	TRUE
Subject_5	Subject_5	133	107	26	TRUE
Subject_6	Subject_6	134	116	18	TRUE
Subject_7	Subject_7	118	113	5	TRUE
Subject_8	Subject_8	126	111	15	TRUE
Subject_9	Subject_9	114	119	-5	TRUE
Subject_10	Subject_10	127	117	10	TRUE
Subject_11	Subject_11	141	101	40	TRUE
Subject_12	Subject_12	138	119	19	TRUE
Subject_13	Subject_13	128	130	-2	FALSE
Subject_14	Subject_14	140	122	18	FALSE
Subject_15	Subject_15	137	106	31	TRUE
Subject_16	Subject_16	131	106	25	TRUE
Subject_17	Subject_17	120	124	-4	FALSE
Subject_18	Subject_18	128	102	26	TRUE

Subject_19	Subject_19	139	117	22	TRUE
Subject_20	Subject_20	135	113	22	TRUE

#2

`bp_list$treatment`

	patient	pre_bp	post_bp	diff_bp	post_less120
Subject_1	Subject_1	130	114	16	TRUE
Subject_2	Subject_2	128	98	30	TRUE
Subject_3	Subject_3	116	113	3	TRUE
Subject_4	Subject_4	124	99	25	TRUE
Subject_5	Subject_5	133	107	26	TRUE
Subject_6	Subject_6	134	116	18	TRUE
Subject_7	Subject_7	118	113	5	TRUE
Subject_8	Subject_8	126	111	15	TRUE
Subject_9	Subject_9	114	119	-5	TRUE
Subject_10	Subject_10	127	117	10	TRUE
Subject_11	Subject_11	141	101	40	TRUE
Subject_12	Subject_12	138	119	19	TRUE
Subject_13	Subject_13	128	130	-2	FALSE
Subject_14	Subject_14	140	122	18	FALSE
Subject_15	Subject_15	137	106	31	TRUE
Subject_16	Subject_16	131	106	25	TRUE
Subject_17	Subject_17	120	124	-4	FALSE
Subject_18	Subject_18	128	102	26	TRUE
Subject_19	Subject_19	139	117	22	TRUE
Subject_20	Subject_20	135	113	22	TRUE

#3

`bp_list[["treatment"]]`

	patient	pre_bp	post_bp	diff_bp	post_less120
Subject_1	Subject_1	130	114	16	TRUE
Subject_2	Subject_2	128	98	30	TRUE
Subject_3	Subject_3	116	113	3	TRUE
Subject_4	Subject_4	124	99	25	TRUE
Subject_5	Subject_5	133	107	26	TRUE
Subject_6	Subject_6	134	116	18	TRUE
Subject_7	Subject_7	118	113	5	TRUE
Subject_8	Subject_8	126	111	15	TRUE
Subject_9	Subject_9	114	119	-5	TRUE
Subject_10	Subject_10	127	117	10	TRUE
Subject_11	Subject_11	141	101	40	TRUE
Subject_12	Subject_12	138	119	19	TRUE
Subject_13	Subject_13	128	130	-2	FALSE
Subject_14	Subject_14	140	122	18	FALSE
Subject_15	Subject_15	137	106	31	TRUE
Subject_16	Subject_16	131	106	25	TRUE
Subject_17	Subject_17	120	124	-4	FALSE

Subject_18	Subject_18	128	102	26	TRUE
Subject_19	Subject_19	139	117	22	TRUE
Subject_20	Subject_20	135	113	22	TRUE

Question 4

```
bp_list$placebo$pre
```

```
[1] 138 135 147 117 152 134 114 121 131 130
```

Task 4: Control Flow Practice

Question 1

```
bp_list$treatment$status <- character(20)
bp_list$placebo$status <- character(10)
```

Question 2

```
for (i in 1:20) { bp<- bp_list$treatment$post_bp[i]

  if (bp <= 120) {
    bp_list$treatment$status[i] <- "optimal"
  } else if (bp <= 130) {
    bp_list$treatment$status[i] <- "borderline"

  } else { bp_list$treatment$status[i] <- "high"}

}
```

Question 3

```
for (i in 1:10) { bp<- bp_list$placebo$post[i]

  if (bp <= 120) {
    bp_list$placebo$status[i] <- "optimal"
  } else if (bp <= 130) {
    bp_list$placebo$status[i] <- "borderline"

  } else { bp_list$placebo$status[i] <- "high"}

}
```

Task 5: Function Writing

Question 1

```
bp_function <- function( bp_list, stat ="mean"){  
  #the function gets a statistic, mean by default  
  my_fun <- get(stat)  
  
  #make a vector of all the stats it should get for each column we want in each df  
  stats<- c(my_fun(bp_list$treatment$pre_bp),  
    my_fun(bp_list$treatment$post_bp),  
    my_fun(bp_list$treatment$diff_bp),  
    my_fun(bp_list$placebo$pre),  
    my_fun(bp_list$placebo$post),  
    my_fun(bp_list$placebo$post_below_120))  
  
  }  
  
  #mean, var, sd, min and max  
  bp_function(bp_list)  
  bp_function(bp_list, "var")  
  bp_function(bp_list, "sd")  
  bp_function(bp_list, "min")  
  bp_function(bp_list, "max")
```