Introduction to FUNCTIONS

Part 2

```
ggplot(df, aes(x = time,
               y = conc,
               group = ID))
```

```
ID <- 20

ggplot(df, aes(x = “time”,
               y = “conc”,
                   group = “ID”))
```

```
ID <- 20

ggplot(df, aes(x = time,
               y = conc,
               group = ID))
```

```
Theoph %>%
    ggplot(aes(x = Time,
               y = conc,
               group = Subject)) +
    geom_line() + geom_point()
```

```
gg_conc_time <- function() {
   Theoph %>%
       ggplot(aes(x = Time,
                  y = conc,
                  group = Subject)) +
       geom_line() + geom_point()
   }
```

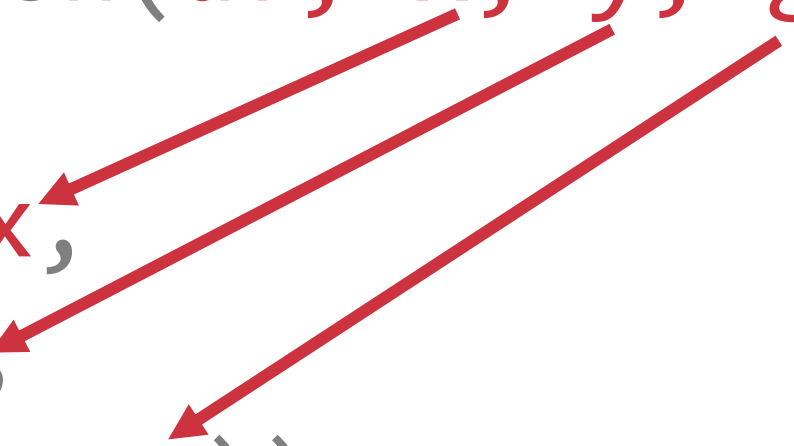
```
gg_conc_time <- function() {
   Theoph %>%
        ggplot(aes(x = Time,
                   y = conc,
                   group = Subject)) +
        geom_line() + geom_point()
    }
```

```
gg_conc_time <- function(df) {
   Theoph %>%
        ggplot(aes(x = Time,
                   y = conc,
                   group = Subject)) +
        geom_line() + geom_point()
    }
```

```
gg_conc_time <- function(df) {
        df %>%
        ggplot(aes(x = Time,
                   y = conc,
                   group = Subject)) +
        geom_line() + geom_point()
    }
```

```
gg_conc_time <- function(df) {
        df %>%
        ggplot(aes(x = Time,
                   y = conc,
                   group = Subject)) +
        geom_line() + geom_point()
    }
```

```
gg_conc_time <- function(df, x, y, g){
        df %>%
        ggplot(aes(x = x,
                   y = y,
                   group = g)) +
        geom_line() + geom_point()
    }
```



Will this work?

```
gg_conc_time(Theoph,
       “Time”,
       “conc”,
       “Subject”)
```

Introduction to

Non-Standard Evaluation (NSE)

function	purpose
rlang::quo()	take in a code with ‘template’ variables, and the reassign the template variables
rlang::eval_tidy()	evaluate the generated code expression from the quo()

```
tvar1 <- “hello”
tvar2 <- “world”
rlang::quo(paste(!!tvar1, !!tvar2))
```

~paste(“hello”, “world”)

```
tvar1 <- “hello”
tvar2 <- “world”
quo_exp <- rlang::quo(paste(!!tvar1, !!tvar2))

rlang::eval_tidy(quo_exp)
```

“hello world”

Is this how you write code?

```
gg_conc_time(Theoph,
       “Time”,
       “conc”,
       “Subject”)
```

Or this…

```
gg_conc_time(Theoph,
       Time,
       conc,
       Subject)
```

```
my_nse_func <- function(param_1, param_2) {
    template_var1 = enexpr(param_1),
    template_var2 = enexpr(param_2)
    ptemplate <- rlang::quo(<code w/ template_vars>)
    return(rlang::eval_tidy(p_template))
}
```

```
gg_conc_time <- function(df, x, y, g){
    x <- enexpr(x)
    y <- enexpr(y)
    g <- enexpr(g)
    res <- rlang::quo(df %>%
        ggplot(aes(x = x, y = y, group = g)) +
        geom_line() + geom_point()
     )
     return(rlang::eval_tidy(res))
     }
```

```
my_nse_func <- function(param_1, param_2) {
    template_var1 = enexpr(param_1),
    template_var2 = enexpr(param_2)
    ptemplate <- rlang::quo(<code w/ template_vars>)
    return(rlang::eval_tidy(p_template))
}

my_se_func <- function(param_1, param_2) {
    template_var1 = sym(param_1),
    template_var2 = sym(param_2)
    ptemplate <- rlang::quo(<code w/ template_vars>)
    return(rlang::eval_tidy(p_template))
}
```