In [1]:

```
library(tidyverse)
```

```
── Attaching packages ─────────────────────────────────────── tidyverse 1.3.0 ──
✔ ggplot2 3.3.3     ✔ purrr   0.3.4
✔ tibble  3.0.6     ✔ dplyr   1.0.4
✔ tidyr   1.1.2     ✔ stringr 1.4.0
✔ readr   1.4.0     ✔ forcats 0.5.1
── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
✖ dplyr::filter() masks stats::filter()
✖ dplyr::lag()    masks stats::lag()
```

Conditional Statements

- if (if..then)
- if...else
- if...else if...else if...else

In [3]:

```
x <- 0
if (x < 0) {
    print("Number is negative")
} else if (x >= 50) {
   print("Number is 50 or greater")
} else if (x == 0) {
   print("Number is 0")
} else {
   print("Number is between 0 and 50")
}
```

```
[1] "Number is 0"
```

In [7]:

```
zero_and_fifty <- function(x) {
    if (x < 0) {
        print("Number is negative")
    } else if (x >= 50) {
        print("Number is 50 or greater")
    } else if (x == 0) {
        print("Number is 0")
    } else {
        print("Number is between 0 and 50")
    }
}

zero_and_fifty(10)
zero_and_fifty(-5)
zero_and_fifty(50)
```

```
[1] "Number is between 0 and 50"
[1] "Number is negative"
[1] "Number is 50 or greater"
```

Attempt:

- Write a function that outputs the square if the input is a positive number, the absolute value if the input is a negative number, and prints "ZERO" if the input is zero.
- What is the output of the following block of code?

In [13]:

```
x <- 0
if (x > 0) {
    x**2
} else if (x < 0) {
    abs(x)
} else if (x == 0){
    print("ZERO")
}
```

```
[1] "ZERO"
```

In [19]:

```
#warm_up_function <- function(x) {
#    if (x > 0) {
#        x**2
#    } else if (x < 0) {
#        abs(x)
#    } else if (x == 0){
#        print("ZERO")
#    }
#}

warm_up_function <- function(x) {
    if (x > 0) {
        x**2
    } else if (x < 0) {
        abs(x)
    } else {
        print("ZERO")
    }
}

warm_up_function(10)
warm_up_function(-10)
warm_up_function(0)
```

100

10

```
[1] "ZERO"
```

In [20]:

```
x <- 0
if (x == 0) {
   print("THIS")
} else if (x < 100) {
    print("THAT")
}
```

```
[1] "THIS"
```

Be careful with the position of the curly braces

In [23]:

```
x <- 0
if (x == 0) {
   print("THIS")
} else if (x < 100) {
    print("THAT")
}
```

```
[1] "THIS"
```

Be careful with the output of your condition statement. It must be either `TRUE` or `FALSE`.

In [27]:

```
c(1,2) == 1
```

TRUE · FALSE

In [26]:

```
if (c(1,2) == 1) {print("THIS")}
```

```
Warning message in if (c(1, 2) == 1) {:
"the condition has length > 1 and only the first element will be used"
[1] "THIS"
```

The following commands condense Boolean vectors:

- `any()` returns `TRUE` if any value is `TRUE`, `FALSE` otherwise
- `all()` returns `TRUE` if all values are `TRUE`, `FALSE` otherwise

In [28]:

```
any(c(T, F))
any(c(F, F))

all(c(T, F))
all(c(T, T))
```

TRUE

FALSE

FALSE

TRUE

You can use logical operators to combine conditions:

- `&&` for "and"
- `||` for "or"
- `identical()` for `==` These are *not* vectorized meaning that they will always output a single `TRUE` / `FALSE` value

In [31]:

```
pos_and_not_five <- function(x) {(x > 0) && !(identical(x,5))}

pos_and_not_five(5)
pos_and_not_five(1)
pos_and_not_five(-1)
```

FALSE

TRUE

FALSE

The command `identical()` returns `TRUE` if all arguments are **exactly** the same, `FALSE` otherwise

In [32]:

```
identical(0, 0, 0, 0, 0)
identical(1L, 1)
```

TRUE

FALSE

Loops

- Useful for when you want to do the same thing to many different inputs

In [34]:

```
seq(1:5)
```

1 · 2 · 3 · 4 · 5

In [33]:

```
# print the first five positive integers
for (i in seq(1:5)) {
    print(i)
}
```

```
[1] 1
[1] 2
[1] 3
[1] 4
[1] 5
```

The above loop is equivalent to:

`print(1)`

`print(2)`

`print(3)`

`print(4)`

`print(5)`

The variable `i` takes on the values in `seq(1:5) = c(1, 2, 3, 4, 5)` one by one.

- We can iterate over vectors and lists

In [39]:

```
x <- list("hello","everyone","it's","wednesday", 2)

for (element in x) {
    print(element)
}
```

```
[1] "hello"
[1] "everyone"
[1] "it's"
[1] "wednesday"
[1] 2
```

In [43]:

```
diamonds_number <- diamonds %>%
    select(depth, table, price, x, y, z) %>%
    head() %>%
    print()
```

```
# A tibble: 6 x 6
  depth table price     x     y     z
  <dbl> <dbl> <int> <dbl> <dbl> <dbl>
1  61.5    55   326  3.95  3.98  2.43
2  59.8    61   326  3.89  3.84  2.31
3  56.9    65   327  4.05  4.07  2.31
4  62.4    58   334  4.2   4.23  2.63
5  63.3    58   335  4.34  4.35  2.75
6  62.8    57   336  3.94  3.96  2.48
```

In [44]:

```
for (column in diamonds_number) {
    column <- column - mean(column)
}
diamonds_number
```

A tibble: 6 × 6

depth	table	price	x	y	z
<dbl>	<dbl>	<int>	<dbl>	<dbl>	<dbl>
61.5	55	326	3.95	3.98	2.43
59.8	61	326	3.89	3.84	2.31
56.9	65	327	4.05	4.07	2.31
62.4	58	334	4.20	4.23	2.63
63.3	58	335	4.34	4.35	2.75
62.8	57	336	3.94	3.96	2.48

In [42]:

```
# to modify a dataframe we can use seq_along()
# here i represents an **index** rather than the element itself
for (i in seq_along(diamonds_number)) {
    diamonds_number[[i]] <- diamonds_number[[i]] - mean(diamonds_number[[i]])
}
diamonds_number
```

A tibble: 6 × 6

depth	table	price	x	y	z
<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
0.3833333	-4	-4.666667	-0.11166667	-0.091666667	-0.055
-1.3166667	2	-4.666667	-0.17166667	-0.231666667	-0.175
-4.2166667	6	-3.666667	-0.01166667	-0.001666667	-0.175
1.2833333	-1	3.333333	0.13833333	0.158333333	0.145
2.1833333	-1	4.333333	0.27833333	0.278333333	0.265
1.6833333	-2	5.333333	-0.12166667	-0.111666667	-0.005

In [49]:

```
x <- c(1,2,3)

for (i in seq_along(x)) {
    x[[i]] <- x[[i]]**2
    print(x[[i]])
}
x
```

```
[1] 1
[1] 4
[1] 9
```

1 · 4 · 9

Attempt

- Write a function that adjusts a column of a tibble by the mean if its an integer or double type column

In [66]:

```
is.numeric(diamonds$carat)
```

TRUE

In [67]:

```
is.numeric(diamonds$price)
```

TRUE

In [68]:

```
is.numeric(diamonds$cut)
```

FALSE

In [70]:

```
column <- diamonds$price
if (is.numeric(column)){
    column <- column - mean(column)
}
```

```
[1] "something"
```

In [72]:

```
new_diamonds <- diamonds
```

In [73]:

```
tib <- new_diamonds
for (i in seq_along(tib)) {
    if (is.numeric(tib[[i]])){
        tib[[i]] <- tib[[i]] - mean(tib[[i]])
    }
}
tib
```

A tibble: 53940 × 10

carat	cut	color	clarity	depth	table	price	x	y	z
<dbl>	<ord>	<ord>	<ord>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
-0.5679397	Ideal	E	SI2	-0.2494049	-2.4571839	-3606.8	-1.781157	-1.754526	-1.1087338
-0.5879397	Premium	E	SI1	-1.9494049	3.5428161	-3606.8	-1.841157	-1.894526	-1.2287338
-0.5679397	Good	E	VS1	-4.8494049	7.5428161	-3605.8	-1.681157	-1.664526	-1.2287338
-0.5079397	Premium	I	VS2	0.6505951	0.5428161	-3598.8	-1.531157	-1.504526	-0.9087338
-0.4879397	Good	J	SI2	1.5505951	0.5428161	-3597.8	-1.391157	-1.384526	-0.7887338
-0.5579397	Very Good	J	VVS2	1.0505951	-0.4571839	-3596.8	-1.791157	-1.774526	-1.0587338
-0.5579397	Very Good	I	VVS1	0.5505951	0.5428161	-3596.8	-1.781157	-1.754526	-1.0687338
-0.5379397	Very Good	H	SI1	0.1505951	-2.4571839	-3595.8	-1.661157	-1.624526	-1.0087338
-0.5779397	Fair	E	VS2	3.3505951	3.5428161	-3595.8	-1.861157	-1.954526	-1.0487338
-0.5679397	Very Good	H	VS1	-2.3494049	3.5428161	-3594.8	-1.731157	-1.684526	-1.1487338
-0.4979397	Good	J	SI1	2.2505951	-2.4571839	-3593.8	-1.481157	-1.454526	-0.8087338
-0.5679397	Ideal	J	VS1	1.0505951	-1.4571839	-3592.8	-1.801157	-1.834526	-1.0787338
-0.5779397	Premium	F	SI1	-1.3494049	3.5428161	-3590.8	-1.851157	-1.894526	-1.2087338
-0.4879397	Ideal	J	SI2	0.4505951	-3.4571839	-3588.8	-1.381157	-1.364526	-0.8287338
-0.5979397	Premium	E	SI2	-1.5494049	4.5428161	-3587.8	-1.941157	-1.984526	-1.2687338
-0.4779397	Premium	E	I1	-0.8494049	0.5428161	-3587.8	-1.351157	-1.314526	-0.8587338
-0.4979397	Ideal	I	SI2	0.2505951	-3.4571839	-3584.8	-1.421157	-1.394526	-0.8587338
-0.4979397	Good	J	SI1	1.6505951	-3.4571839	-3581.8	-1.501157	-1.444526	-0.8387338
-0.4979397	Good	J	SI1	2.0505951	-3.4571839	-3581.8	-1.501157	-1.474526	-0.8287338
-0.4979397	Very Good	J	SI1	0.9505951	1.5428161	-3581.8	-1.521157	-1.464526	-0.8787338
-0.4979397	Good	I	SI2	1.5505951	-1.4571839	-3581.8	-1.471157	-1.434526	-0.8287338
-0.5679397	Very Good	E	VS2	2.0505951	-2.4571839	-3580.8	-1.881157	-1.814526	-1.0587338
-0.5679397	Very Good	H	VS1	-0.7494049	-0.4571839	-3579.8	-1.791157	-1.774526	-1.1287338
-0.4879397	Very Good	J	SI1	-2.3494049	4.5428161	-3579.8	-1.341157	-1.304526	-0.9187338
-0.4879397	Very Good	J	SI1	-3.6494049	4.5428161	-3579.8	-1.291157	-1.264526	-0.9487338
-0.5679397	Very Good	G	VVS2	-1.3494049	0.5428161	-3578.8	-1.761157	-1.724526	-1.1287338
-0.5579397	Premium	I	VS1	0.7505951	-0.4571839	-3577.8	-1.761157	-1.794526	-1.0687338
-0.4979397	Very Good	J	VS2	0.4505951	-1.4571839	-3575.8	-1.451157	-1.434526	-0.8687338
-0.5679397	Very Good	D	VS2	-1.2494049	3.5428161	-3575.8	-1.771157	-1.764526	-1.1387338
-0.5679397	Very Good	F	VS1	-0.8494049	-0.4571839	-3575.8	-1.771157	-1.744526	-1.1187338
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
-0.097939748	Premium	E	SI1	-1.24940489	0.5428161	-1179.8	0.008842788	0.035474045	-0.058733778
-0.227939748	Premium	E	IF	-1.94940489	2.5428161	-1179.8	-0.301157212	-0.354525955	-0.308733778
-0.187939748	Premium	F	VVS1	0.05059511	1.5428161	-1179.8	-0.251157212	-0.334525955	-0.178733778
0.002060252	Good	G	VS2	2.45059511	0.5428161	-1179.8	0.108842788	0.075474045	0.201266222
0.042060252	Good	I	VS1	1.95059511	1.5428161	-1179.8	0.208842788	0.165474045	0.231266222
-0.027939748	Ideal	E	SI2	0.35059511	-1.4571839	-1179.8	0.108842788	0.125474045	0.091266222
-0.057939748	Good	D	SI1	1.35059511	1.5428161	-1179.8	-0.021157212	0.005474045	0.071266222
0.102060252	Very Good	J	SI1	1.45059511	2.5428161	-1179.8	0.388842788	0.355474045	0.321266222
-0.037939748	Premium	I	VS1	-2.44940489	4.5428161	-1179.8	0.198842788	0.115474045	-0.048733778
-0.037939748	Ideal	I	VVS1	0.45059511	-2.4571839	-1179.8	0.158842788	0.135474045	0.121266222
-0.097939748	Very Good	E	VS2	0.65059511	2.5428161	-1177.8	-0.161157212	-0.124525955	-0.048733778
-0.097939748	Very Good	E	VS2	1.05059511	2.5428161	-1177.8	-0.141157212	-0.084525955	-0.008733778
-0.097939748	Very Good	D	VS1	1.35059511	1.5428161	-1177.8	-0.061157212	-0.154525955	0.011266222
-0.067939748	Ideal	I	VS2	-0.44940489	-1.4571839	-1176.8	0.068842788	0.105474045	0.031266222
-0.067939748	Ideal	I	VS2	-0.14940489	-2.4571839	-1176.8	0.088842788	0.105474045	0.051266222
-0.007939748	Ideal	I	SI1	-0.14940489	-1.4571839	-1176.8	0.218842788	0.235474045	0.131266222
-0.087939748	Ideal	E	SI1	0.15059511	-1.4571839	-1176.8	-0.021157212	-0.004525955	0.001266222
-0.007939748	Good	F	SI1	-3.64940489	1.5428161	-1176.8	0.328842788	0.395474045	0.001266222
-0.007939748	Premium	E	SI2	-0.34940489	0.5428161	-1176.8	0.298842788	0.225474045	0.141266222
-0.087939748	Ideal	G	VS1	-0.34940489	-1.4571839	-1176.8	0.028842788	-0.004525955	-0.008733778
-0.087939748	Premium	E	SI1	-1.24940489	-2.4571839	-1176.8	0.058842788	0.005474045	-0.048733778
-0.087939748	Premium	F	SI1	-1.94940489	4.5428161	-1176.8	0.008842788	-0.004525955	-0.108733778
-0.097939748	Very Good	E	VS2	-1.24940489	1.5428161	-1175.8	-0.021157212	0.025474045	-0.068733778
-0.097939748	Very Good	E	VS2	-0.54940489	1.5428161	-1175.8	-0.041157212	-0.014525955	-0.048733778
-0.077939748	Premium	D	SI1	0.95059511	1.5428161	-1175.8	-0.041157212	-0.004525955	0.041266222
-0.077939748	Ideal	D	SI1	-0.94940489	-0.4571839	-1175.8	0.018842788	0.025474045	-0.038733778
-0.077939748	Good	D	SI1	1.35059511	-2.4571839	-1175.8	-0.041157212	0.015474045	0.071266222
-0.097939748	Very Good	D	SI1	1.05059511	2.5428161	-1175.8	-0.071157212	-0.054525955	0.021266222
0.062060252	Premium	H	SI2	-0.74940489	0.5428161	-1175.8	0.418842788	0.385474045	0.201266222
-0.047939748	Ideal	D	SI2	0.45059511	-2.4571839	-1175.8	0.098842788	0.135474045	0.101266222

In [83]:

```
subtract_tib_mean <- function(tib) {
    # tib is a copy of the input tibble; the following code will not actually alter the original object
    for (i in seq_along(tib)){
        if (is.numeric(tib[[i]])) {
            tib[[i]] <- tib[[i]] - mean(tib[[i]])
        }
    }
    tib # need to add this line to make sure function *outputs* a tibble
}
```

In [84]:

```
subtract_tib_mean(new_diamonds)
```

A tibble: 53940 × 10

carat	cut	color	clarity	depth	table	price	x	y	z
<dbl>	<ord>	<ord>	<ord>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>	<dbl>
-0.5679397	Ideal	E	SI2	-0.2494049	-2.4571839	-3606.8	-1.781157	-1.754526	-1.1087338
-0.5879397	Premium	E	SI1	-1.9494049	3.5428161	-3606.8	-1.841157	-1.894526	-1.2287338
-0.5679397	Good	E	VS1	-4.8494049	7.5428161	-3605.8	-1.681157	-1.664526	-1.2287338
-0.5079397	Premium	I	VS2	0.6505951	0.5428161	-3598.8	-1.531157	-1.504526	-0.9087338
-0.4879397	Good	J	SI2	1.5505951	0.5428161	-3597.8	-1.391157	-1.384526	-0.7887338
-0.5579397	Very Good	J	VVS2	1.0505951	-0.4571839	-3596.8	-1.791157	-1.774526	-1.0587338
-0.5579397	Very Good	I	VVS1	0.5505951	0.5428161	-3596.8	-1.781157	-1.754526	-1.0687338
-0.5379397	Very Good	H	SI1	0.1505951	-2.4571839	-3595.8	-1.661157	-1.624526	-1.0087338
-0.5779397	Fair	E	VS2	3.3505951	3.5428161	-3595.8	-1.861157	-1.954526	-1.0487338
-0.5679397	Very Good	H	VS1	-2.3494049	3.5428161	-3594.8	-1.731157	-1.684526	-1.1487338
-0.4979397	Good	J	SI1	2.2505951	-2.4571839	-3593.8	-1.481157	-1.454526	-0.8087338
-0.5679397	Ideal	J	VS1	1.0505951	-1.4571839	-3592.8	-1.801157	-1.834526	-1.0787338
-0.5779397	Premium	F	SI1	-1.3494049	3.5428161	-3590.8	-1.851157	-1.894526	-1.2087338
-0.4879397	Ideal	J	SI2	0.4505951	-3.4571839	-3588.8	-1.381157	-1.364526	-0.8287338
-0.5979397	Premium	E	SI2	-1.5494049	4.5428161	-3587.8	-1.941157	-1.984526	-1.2687338
-0.4779397	Premium	E	I1	-0.8494049	0.5428161	-3587.8	-1.351157	-1.314526	-0.8587338
-0.4979397	Ideal	I	SI2	0.2505951	-3.4571839	-3584.8	-1.421157	-1.394526	-0.8587338
-0.4979397	Good	J	SI1	1.6505951	-3.4571839	-3581.8	-1.501157	-1.444526	-0.8387338
-0.4979397	Good	J	SI1	2.0505951	-3.4571839	-3581.8	-1.501157	-1.474526	-0.8287338
-0.4979397	Very Good	J	SI1	0.9505951	1.5428161	-3581.8	-1.521157	-1.464526	-0.8787338
-0.4979397	Good	I	SI2	1.5505951	-1.4571839	-3581.8	-1.471157	-1.434526	-0.8287338
-0.5679397	Very Good	E	VS2	2.0505951	-2.4571839	-3580.8	-1.881157	-1.814526	-1.0587338
-0.5679397	Very Good	H	VS1	-0.7494049	-0.4571839	-3579.8	-1.791157	-1.774526	-1.1287338
-0.4879397	Very Good	J	SI1	-2.3494049	4.5428161	-3579.8	-1.341157	-1.304526	-0.9187338
-0.4879397	Very Good	J	SI1	-3.6494049	4.5428161	-3579.8	-1.291157	-1.264526	-0.9487338
-0.5679397	Very Good	G	VVS2	-1.3494049	0.5428161	-3578.8	-1.761157	-1.724526	-1.1287338
-0.5579397	Premium	I	VS1	0.7505951	-0.4571839	-3577.8	-1.761157	-1.794526	-1.0687338
-0.4979397	Very Good	J	VS2	0.4505951	-1.4571839	-3575.8	-1.451157	-1.434526	-0.8687338
-0.5679397	Very Good	D	VS2	-1.2494049	3.5428161	-3575.8	-1.771157	-1.764526	-1.1387338
-0.5679397	Very Good	F	VS1	-0.8494049	-0.4571839	-3575.8	-1.771157	-1.744526	-1.1187338
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
-0.097939748	Premium	E	SI1	-1.24940489	0.5428161	-1179.8	0.008842788	0.035474045	-0.058733778
-0.227939748	Premium	E	IF	-1.94940489	2.5428161	-1179.8	-0.301157212	-0.354525955	-0.308733778
-0.187939748	Premium	F	VVS1	0.05059511	1.5428161	-1179.8	-0.251157212	-0.334525955	-0.178733778
0.002060252	Good	G	VS2	2.45059511	0.5428161	-1179.8	0.108842788	0.075474045	0.201266222
0.042060252	Good	I	VS1	1.95059511	1.5428161	-1179.8	0.208842788	0.165474045	0.231266222
-0.027939748	Ideal	E	SI2	0.35059511	-1.4571839	-1179.8	0.108842788	0.125474045	0.091266222
-0.057939748	Good	D	SI1	1.35059511	1.5428161	-1179.8	-0.021157212	0.005474045	0.071266222
0.102060252	Very Good	J	SI1	1.45059511	2.5428161	-1179.8	0.388842788	0.355474045	0.321266222
-0.037939748	Premium	I	VS1	-2.44940489	4.5428161	-1179.8	0.198842788	0.115474045	-0.048733778
-0.037939748	Ideal	I	VVS1	0.45059511	-2.4571839	-1179.8	0.158842788	0.135474045	0.121266222
-0.097939748	Very Good	E	VS2	0.65059511	2.5428161	-1177.8	-0.161157212	-0.124525955	-0.048733778
-0.097939748	Very Good	E	VS2	1.05059511	2.5428161	-1177.8	-0.141157212	-0.084525955	-0.008733778
-0.097939748	Very Good	D	VS1	1.35059511	1.5428161	-1177.8	-0.061157212	-0.154525955	0.011266222
-0.067939748	Ideal	I	VS2	-0.44940489	-1.4571839	-1176.8	0.068842788	0.105474045	0.031266222
-0.067939748	Ideal	I	VS2	-0.14940489	-2.4571839	-1176.8	0.088842788	0.105474045	0.051266222
-0.007939748	Ideal	I	SI1	-0.14940489	-1.4571839	-1176.8	0.218842788	0.235474045	0.131266222
-0.087939748	Ideal	E	SI1	0.15059511	-1.4571839	-1176.8	-0.021157212	-0.004525955	0.001266222
-0.007939748	Good	F	SI1	-3.64940489	1.5428161	-1176.8	0.328842788	0.395474045	0.001266222
-0.007939748	Premium	E	SI2	-0.34940489	0.5428161	-1176.8	0.298842788	0.225474045	0.141266222
-0.087939748	Ideal	G	VS1	-0.34940489	-1.4571839	-1176.8	0.028842788	-0.004525955	-0.008733778
-0.087939748	Premium	E	SI1	-1.24940489	-2.4571839	-1176.8	0.058842788	0.005474045	-0.048733778
-0.087939748	Premium	F	SI1	-1.94940489	4.5428161	-1176.8	0.008842788	-0.004525955	-0.108733778
-0.097939748	Very Good	E	VS2	-1.24940489	1.5428161	-1175.8	-0.021157212	0.025474045	-0.068733778
-0.097939748	Very Good	E	VS2	-0.54940489	1.5428161	-1175.8	-0.041157212	-0.014525955	-0.048733778
-0.077939748	Premium	D	SI1	0.95059511	1.5428161	-1175.8	-0.041157212	-0.004525955	0.041266222
-0.077939748	Ideal	D	SI1	-0.94940489	-0.4571839	-1175.8	0.018842788	0.025474045	-0.038733778
-0.077939748	Good	D	SI1	1.35059511	-2.4571839	-1175.8	-0.041157212	0.015474045	0.071266222
-0.097939748	Very Good	D	SI1	1.05059511	2.5428161	-1175.8	-0.071157212	-0.054525955	0.021266222
0.062060252	Premium	H	SI2	-0.74940489	0.5428161	-1175.8	0.418842788	0.385474045	0.201266222
-0.047939748	Ideal	D	SI2	0.45059511	-2.4571839	-1175.8	0.098842788	0.135474045	0.101266222

In [82]:

```
new_diamonds
```

A tibble: 53940 × 10

carat	cut	color	clarity	depth	table	price	x	y	z
<dbl>	<ord>	<ord>	<ord>	<dbl>	<dbl>	<int>	<dbl>	<dbl>	<dbl>
0.23	Ideal	E	SI2	61.5	55	326	3.95	3.98	2.43
0.21	Premium	E	SI1	59.8	61	326	3.89	3.84	2.31
0.23	Good	E	VS1	56.9	65	327	4.05	4.07	2.31
0.29	Premium	I	VS2	62.4	58	334	4.20	4.23	2.63
0.31	Good	J	SI2	63.3	58	335	4.34	4.35	2.75
0.24	Very Good	J	VVS2	62.8	57	336	3.94	3.96	2.48
0.24	Very Good	I	VVS1	62.3	57	336	3.95	3.98	2.47
0.26	Very Good	H	SI1	61.9	55	337	4.07	4.11	2.53
0.22	Fair	E	VS2	65.1	61	337	3.87	3.78	2.49
0.23	Very Good	H	VS1	59.4	61	338	4.00	4.05	2.39
0.30	Good	J	SI1	64.0	55	339	4.25	4.28	2.73
0.23	Ideal	J	VS1	62.8	56	340	3.93	3.90	2.46
0.22	Premium	F	SI1	60.4	61	342	3.88	3.84	2.33
0.31	Ideal	J	SI2	62.2	54	344	4.35	4.37	2.71
0.20	Premium	E	SI2	60.2	62	345	3.79	3.75	2.27
0.32	Premium	E	I1	60.9	58	345	4.38	4.42	2.68
0.30	Ideal	I	SI2	62.0	54	348	4.31	4.34	2.68
0.30	Good	J	SI1	63.4	54	351	4.23	4.29	2.70
0.30	Good	J	SI1	63.8	56	351	4.23	4.26	2.71
0.30	Very Good	J	SI1	62.7	59	351	4.21	4.27	2.66
0.30	Good	I	SI2	63.3	56	351	4.26	4.30	2.71
0.23	Very Good	E	VS2	63.8	55	352	3.85	3.92	2.48
0.23	Very Good	H	VS1	61.0	57	353	3.94	3.96	2.41
0.31	Very Good	J	SI1	59.4	62	353	4.39	4.43	2.62
0.31	Very Good	J	SI1	58.1	62	353	4.44	4.47	2.59
0.23	Very Good	G	VVS2	60.4	58	354	3.97	4.01	2.41
0.24	Premium	I	VS1	62.5	57	355	3.97	3.94	2.47
0.30	Very Good	J	VS2	62.2	57	357	4.28	4.30	2.67
0.23	Very Good	D	VS2	60.5	61	357	3.96	3.97	2.40
0.23	Very Good	F	VS1	60.9	57	357	3.96	3.99	2.42
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
0.70	Premium	E	SI1	60.5	58	2753	5.74	5.77	3.48
0.57	Premium	E	IF	59.8	60	2753	5.43	5.38	3.23
0.61	Premium	F	VVS1	61.8	59	2753	5.48	5.40	3.36
0.80	Good	G	VS2	64.2	58	2753	5.84	5.81	3.74
0.84	Good	I	VS1	63.7	59	2753	5.94	5.90	3.77
0.77	Ideal	E	SI2	62.1	56	2753	5.84	5.86	3.63
0.74	Good	D	SI1	63.1	59	2753	5.71	5.74	3.61
0.90	Very Good	J	SI1	63.2	60	2753	6.12	6.09	3.86
0.76	Premium	I	VS1	59.3	62	2753	5.93	5.85	3.49
0.76	Ideal	I	VVS1	62.2	55	2753	5.89	5.87	3.66
0.70	Very Good	E	VS2	62.4	60	2755	5.57	5.61	3.49
0.70	Very Good	E	VS2	62.8	60	2755	5.59	5.65	3.53
0.70	Very Good	D	VS1	63.1	59	2755	5.67	5.58	3.55
0.73	Ideal	I	VS2	61.3	56	2756	5.80	5.84	3.57
0.73	Ideal	I	VS2	61.6	55	2756	5.82	5.84	3.59
0.79	Ideal	I	SI1	61.6	56	2756	5.95	5.97	3.67
0.71	Ideal	E	SI1	61.9	56	2756	5.71	5.73	3.54
0.79	Good	F	SI1	58.1	59	2756	6.06	6.13	3.54
0.79	Premium	E	SI2	61.4	58	2756	6.03	5.96	3.68
0.71	Ideal	G	VS1	61.4	56	2756	5.76	5.73	3.53
0.71	Premium	E	SI1	60.5	55	2756	5.79	5.74	3.49
0.71	Premium	F	SI1	59.8	62	2756	5.74	5.73	3.43
0.70	Very Good	E	VS2	60.5	59	2757	5.71	5.76	3.47
0.70	Very Good	E	VS2	61.2	59	2757	5.69	5.72	3.49
0.72	Premium	D	SI1	62.7	59	2757	5.69	5.73	3.58
0.72	Ideal	D	SI1	60.8	57	2757	5.75	5.76	3.50
0.72	Good	D	SI1	63.1	55	2757	5.69	5.75	3.61
0.70	Very Good	D	SI1	62.8	60	2757	5.66	5.68	3.56
0.86	Premium	H	SI2	61.0	58	2757	6.15	6.12	3.74
0.75	Ideal	D	SI2	62.2	55	2757	5.83	5.87	3.64

In [ ]: