Group and Summarize Notebook

```
library(nycflights13)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'

## The following objects are masked from 'package:stats':
##
##     filter, lag

## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
# Summarize all
flights_mean_delay <- flights %>%
  summarize(delay_avg = mean(dep_delay, na.rm = TRUE))



#Summarize by group_by()
flights_mean_delay <- flights %>%
  group_by(carrier) %>%
  summarize(delay_avg = mean(dep_delay, na.rm = TRUE))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
#Summarize multiple groups

flights_mean_delay <- flights %>%
  group_by(month, carrier) %>%
  summarize(delay_avg = mean(dep_delay, na.rm = TRUE))
```

```
## `summarise()` regrouping output by 'month' (override with `.groups` argument)
```

```
#Summarize multiple variables

flights_mean_delay <- flights %>%
  group_by(month, carrier) %>%
  summarize(total_flights = n_distinct(flight), delay_avg = mean(dep_delay, na.rm = TRUE))
```

```
## `summarise()` regrouping output by 'month' (override with `.groups` argument)
```

```r
# A more complex example

flights_mean_delay <- flights %>%
  group_by(carrier, month, .add = F) %>%
  summarize(count = n(), avg_delay = mean(arr_delay, na.rm = TRUE)) %>%
  mutate(pct_of_max = avg_delay/max(avg_delay))
```

```
## `summarise()` regrouping output by 'carrier' (override with `.groups` argument)
```

```r
# How to deal with complexity

#
# covid_raw_best_date_f <- cases %>%
#   select(date = best_date, cases, deaths, location_level, location) %>%
#   arrange(location, date) %>%
#   group_by(location, date) %>%
#   calculate_sum(., "cases", "cases_new_best_date", cumulative = FALSE) %>%
#   calculate_sum(., "deaths", "deaths_new_best_date", cumulative = FALSE) %>%
#   select(-c(cases,deaths)) %>%
#   distinct()%>%
#   full_join(cases_best_shell,by=c("location_level","location","date")) %>%
#   replace_na(list(cases_new_best_date= 0, deaths_new_best_date = 0)) %>%
#   # bridge_location_pop(.) %>%
#   arrange(location,date) %>%
#   group_by(location) %>%
#   calculate_sum(., "cases_new_best_date", "cases_cumulative_best_date", cumulative = TRUE) %>%
#   calculate_sum(., "deaths_new_best_date", "deaths_cumulative_best_date", cumulative = TRUE) %>%
#   select(location_level, location, date, cases_new_best_date, deaths_new_best_date, cases_cumulative_
#   filter(date>as_date("2020-01-01") & date<cases_file_date & !is.na(location) & location != "Unassign
#   pivot_longer(
#     cols = c("cases_new_best_date","deaths_new_best_date","cases_cumulative_best_date","deaths_cumula
#     names_to = "variable",
#     values_to = "value"
#   )
```