506-HW-5

Vasudha

Problem - 1

OOP Programming

1 - `Constructor` and `Validator` (with rationality check)

```
library(methods)

# Rational S4 class
setClass(
  "Rational",
  slots = list(
    numerator = "numeric",
    denominator = "numeric"
  ),
  validity = function(object) {
    if (object@denominator == 0) {
      return("Denominator cannot be zero.")
    }
    TRUE
  }
)

# Constructor function
rational <- function(numerator, denominator) {
  new("Rational", numerator = numerator, denominator = denominator)
}
```

3 - `Show` Method

```
setMethod(
  "show",
  "Rational",
  function(object) {
    cat(paste0(object@numerator, "/", object@denominator, "\n"))
  }
)
```

4 - Simplify Method

```
library(Rcpp)
```

```
Warning: package 'Rcpp' was built under R version 4.4.2
```

```
# Define GCD and LCM functions
Rcpp::cppFunction("
int gcd(int a, int b) {
  while (b != 0) {
    int temp = b;
    b = a % b;
    a = temp;
  }
  return a;
}
")

# Simplify method
setGeneric("simplify", function(object) standardGeneric("simplify"))
```

```
[1] "simplify"
```

```
setMethod(
  "simplify",
  "Rational",
  function(object) {
    g <- gcd(object@numerator, object@denominator)
    object@numerator <- object@numerator / g
    object@denominator <- object@denominator / g
    object
  }
)
```

5 - Quotient Method

```
# Quotient method
setGeneric("quotient", function(object, digits = 7) standardGeneric("quotient"))
```

```
[1] "quotient"
```

```
setMethod(
  "quotient",
  "Rational",
  function(object, digits = 7) {
    if (!is.numeric(digits) || digits <= 0) {
      stop("Digits must be a positive numeric value.")
    }
    res <- object@numerator / object@denominator
    cat(format(res, digits = digits), "\n")
    res
  }
)
```

```r
# Arithmetic methods
setMethod("+", signature(e1 = "Rational", e2 = "Rational"), function(e1, e2) {
  numerator <- e1@numerator * e2@denominator + e2@numerator * e1@denominator
  denominator <- e1@denominator * e2@denominator
  simplify(rational(numerator, denominator))
})

setMethod("-", signature(e1 = "Rational", e2 = "Rational"), function(e1, e2) {
  numerator <- e1@numerator * e2@denominator - e2@numerator * e1@denominator
  denominator <- e1@denominator * e2@denominator
  simplify(rational(numerator, denominator))
})

setMethod("*", signature(e1 = "Rational", e2 = "Rational"), function(e1, e2) {
  numerator <- e1@numerator * e2@numerator
  denominator <- e1@denominator * e2@denominator
  simplify(rational(numerator, denominator))
})

setMethod("/", signature(e1 = "Rational", e2 = "Rational"), function(e1, e2) {
  numerator <- e1@numerator * e2@denominator
  denominator <- e1@denominator * e2@numerator
  simplify(rational(numerator, denominator))
})
```

6 - Testing the Methods & Creating Rational Objects

```r
# Rational objects
r1 <- rational(3, 4)
r2 <- rational(5, 8)
r3 <- rational(2, 4)

# Testing
r1
```

```
3/4
```

```r
r2
```

```
5/8
```

```r
r1 + r2
```

```
11/8
```

```r
r1 - r2
```

```
1/8
```

```r
r1 * r2
```

```
15/32
```

```r
r1 / r2
```

```
6/5
```

```
simplify(r1)
```

```
3/4
```

```
quotient(r1)
```

```
0.75
[1] 0.75
```

```
quotient(r2, digits = 3)
```

```
0.625
[1] 0.625
```

Problem - 2

1 - Regenerating plots from the previous homework

Here, I'm first re-framing and cleaning the art data set using my code from the last homework

```
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
```

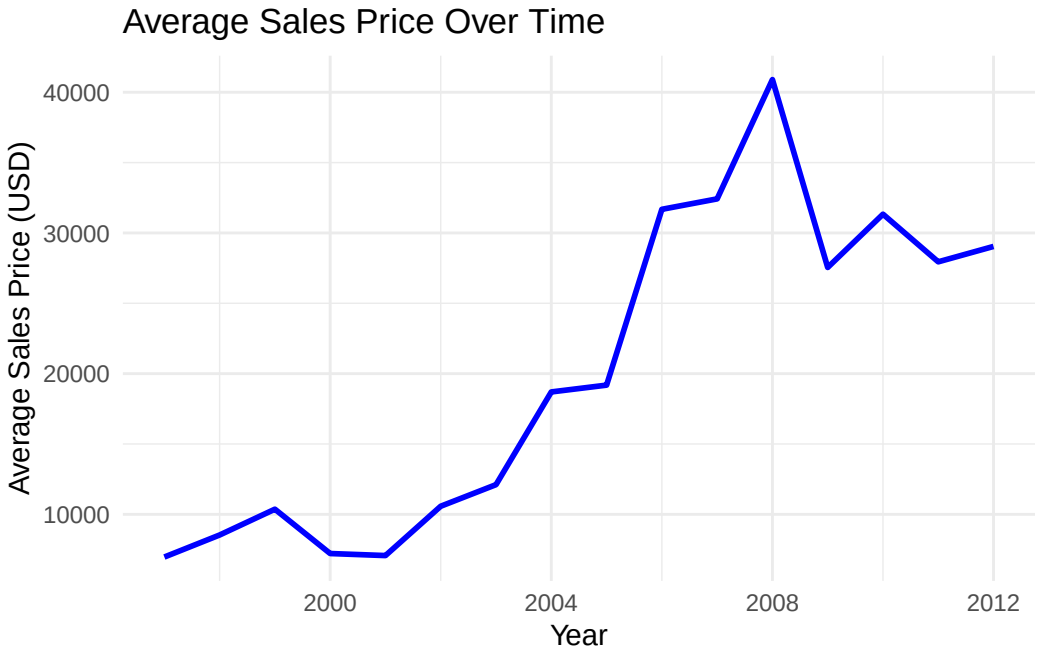
```
library(ggplot2)
library(plotly)
```

```

Attaching package: 'plotly'
The following object is masked from 'package:ggplot2':

    last_plot
The following object is masked from 'package:stats':

    filter
The following object is masked from 'package:graphics':

    layout
```

```
art_sales <- read.csv("df_for_ml_improved_new_market.csv")
art_sales <- as.data.frame(art_sales)
```

```
sales_price_over_time <- art_sales %>%
  group_by(year) %>%
  summarise(mean_price = mean(price_usd, na.rm = TRUE)) %>%
  ggplot(aes(x = year, y = mean_price)) +
```

```
  geom_line(color = "blue", linewidth = 1) +
  labs(title = "Average Sales Price Over Time", x = "Year", y = "Average Sales Price (USD)") +
  theme_minimal()

# Display the plot
sales_price_over_time
```



Redefining Column Data

```
art_sales <- art_sales %>%
  mutate(genre = case_when(
    Genre___Photography == 1 ~ "Photography",
    Genre___Print == 1 ~ "Print",
    Genre___Sculpture == 1 ~ "Sculpture",
    Genre___Painting == 1 ~ "Painting",
    Genre___Others == 1 ~ "Others",
    TRUE ~ NA_character_
  ))
print(head(art_sales[[113]],10)) # checking if transformation
```

```
 [1] "Painting"    "Sculpture"   "Sculpture"   "Painting"    "Photography"
 [6] "Painting"    "Painting"    "Painting"    "Painting"    "Sculpture"
```

```
                                # was successful
```

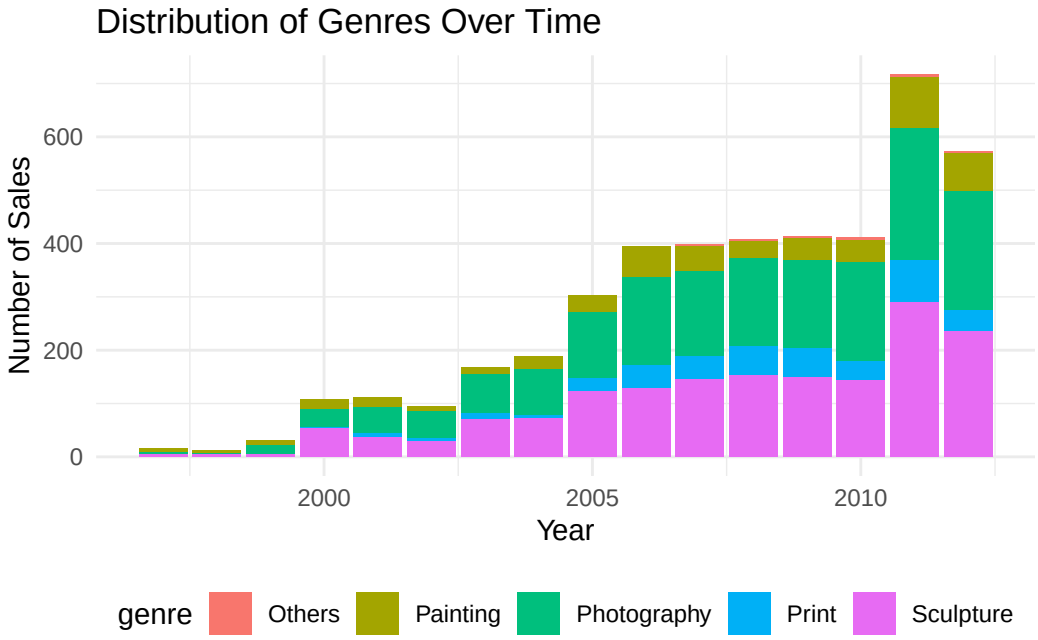
Fluctuation in the selling price of each genre over time

```
genre_distribution <- art_sales %>%
  group_by(year,genre) %>%
  summarise(num_sales = n()) %>%
  ggplot(aes(x = year, y = num_sales, fill = genre)) +
  geom_bar(stat = "identity", position = "stack") +
  labs(title = "Distribution of Genres Over Time", x = "Year", y = "Number of Sales") +
  theme_minimal() +
  theme(legend.position = "bottom")
```

```
`summarise()` has grouped output by 'year'. You can override using the
```

`.groups` argument.

```
## Display plot
genre_distribution
```



2 - Interactive plot with `plotly()`

```
library(htmlwidgets)

plot <- plot_ly(art_sales, x = ~year, y = ~price_usd, color = ~genre, type = 'scatter', mode =
  layout(
    title = "Change in Sales Price Over Time by Genre",
    xaxis = list(title = "Year"),
    yaxis = list(title = "Sales Price (USD)"),
    legend = list(title = list(text = "Genre"))
  )

# Save the plot as an HTML file
saveWidget(plot, "interactive_plot.html")
```

You can view the interactive plot by clicking here.

Problem - 3

Solution to HW-4 Problem-1 using data.table

```
library(data.table)
```

```
Warning: package 'data.table' was built under R version 4.4.2
```

```
Attaching package: 'data.table'
```

```
The following objects are masked from 'package:dplyr':

    between, first, last
```

```
library(nycflights13)
library(kableExtra)
```

```
Attaching package: 'kableExtra'
The following object is masked from 'package:dplyr':

    group_rows
```

```
library(knitr)
```

```
Warning: package 'knitr' was built under R version 4.4.2
```

```
# Convert relevant datasets to data.table
flights_dt <- as.data.table(flights)
airports_dt <- as.data.table(airports)
planes_dt <- as.data.table(planes)

# Departure delay summary
departure_delay_summary <- flights_dt[
  !is.na(dep_delay), # Exclude rows with NA dep_delay
  .(
    `Mean Delay` = mean(dep_delay, na.rm = TRUE),
    `Median Delay` = median(dep_delay, na.rm = TRUE),
    Flights = .N
  ), by = dest
][
  Flights >= 10
][
  order(-`Mean Delay`)
][
  airports_dt, on = .(dest = faa), nomatch = 0
][
  , .(name, `Mean Delay`, `Median Delay`, Flights) # Use the column directly after the join
]

if (nrow(departure_delay_summary) == 0) {
  departure_delay_summary <- data.table(name = NA, `Mean Delay` = NA, `Median Delay` = NA, Flig
}

kable(departure_delay_summary, caption = "Departure Delays", digits = 1, align = 'c')
```

Table 1: Departure Delays

name	Mean Delay	Median Delay	Flights
Albuquerque International Sunport	13.7	0.0	254
Nantucket Mem	6.5	-3.0	265
Albany Intl	23.6	1.0	419
Hartsfield Jackson Atlanta Intl	12.5	-2.0	16898

name	Mean Delay	Median Delay	Flights
Austin Bergstrom Intl	13.0	-1.0	2418
Asheville Regional Airport	8.2	-3.0	263
Bradley Intl	17.7	-1.0	412
Bangor Intl	19.5	-2.0	360
Birmingham Intl	29.7	1.0	272
Nashville Intl	16.0	-1.0	6104
General Edward Lawrence Logan Intl	8.7	-3.0	15049
Burlington Intl	13.6	-2.0	2513
Buffalo Niagara Intl	13.4	-2.0	4576
Bob Hope	13.5	-1.0	370
Baltimore Washington Intl	16.4	-2.0	1696
Gallatin Field	11.5	0.0	35
Columbia Metropolitan	35.6	14.0	107
Akron Canton Regional Airport	20.8	0.0	843
Charlottesville-Albemarle	21.4	-2.5	46
Charleston Afb Intl	14.7	-2.0	2775
Cleveland Hopkins Intl	13.4	-2.0	4408
Charlotte Douglas Intl	9.2	-3.0	13698
Port Columbus Intl	12.2	-3.0	3338
Yeager	17.0	-4.0	137
Cincinnati Northern Kentucky Intl	19.5	-2.0	3740
James M Cox Dayton Intl	17.5	-2.0	1402
Ronald Reagan Washington Natl	10.3	-3.0	9157
Denver Intl	15.2	1.0	7201
Dallas Fort Worth Intl	8.7	-3.0	8463
Des Moines Intl	26.2	-1.0	528
Detroit Metro Wayne Co	11.8	-3.0	9060
Eagle Co Rgnl	15.5	-1.0	208
Key West Intl	3.6	0.0	17
Fort Lauderdale Hollywood Intl	12.7	-1.0	11934
Gerald R Ford Intl	19.5	-1.0	735
Piedmont Triad	19.4	-1.0	1500
Greenville-Spartanburg International	19.3	-1.0	794
Yampa Valley	12.3	6.5	14
Honolulu Intl	9.3	-1.0	705
William P Hobby	14.3	0.0	2090
Washington Dulles Intl	17.0	-2.0	5391
George Bush Intercontinental	10.8	0.0	7103
Wilmington Intl	19.4	-3.0	108
Indianapolis Intl	14.0	-2.0	1991
Jackson Hole Airport	26.5	13.5	22
Jacksonville Intl	16.5	-1.0	2634
Mc Carran Intl	9.4	-1.0	5962
Los Angeles Intl	9.4	-1.0	16076
Long Beach	11.2	-1.0	664
Kansas City Intl	20.3	-1.0	1896
Orlando Intl	11.3	-1.0	13982
Chicago Midway Intl	18.6	2.0	4044
Memphis Intl	15.7	-1.0	1694

name	Mean Delay	Median Delay	Flights
Manchester Regional Airport	21.0	0.0	932
Miami Intl	8.9	-2.0	11633
General Mitchell Intl	18.8	0.0	2718
Dane Co Rgnl Truax Fld	23.6	-1.0	562
Minneapolis St Paul Intl	13.3	-2.0	6958
Louis Armstrong New Orleans Intl	14.2	-2.0	3724
Montrose Regional Airport	17.6	3.0	14
Martha\'s Vineyard	7.1	-2.0	213
Myrtle Beach Intl	15.8	-1.0	58
Metropolitan Oakland Intl	13.3	0.0	311
Will Rogers World	30.6	10.0	327
Eppley Afld	20.2	-1.0	822
Chicago Ohare Intl	13.6	-2.0	16642
Norfolk Intl	17.6	-2.0	1440
Palm Beach Intl	13.0	0.0	6495
Portland Intl	16.3	1.0	1348
Philadelphia Intl	12.0	-3.0	1549
Phoenix Sky Harbor Intl	10.4	-1.0	4611
Pittsburgh Intl	13.7	-2.0	2759
Palm Springs Intl	-2.9	-4.0	18
Theodore Francis Green State	21.8	0.0	358
Portland Intl Jetport	16.5	-2.0	2295
Raleigh Durham Intl	12.4	-2.0	7796
Richmond Intl	23.6	-1.0	2349
Greater Rochester Intl	16.2	-2.0	2362
Southwest Florida Intl	8.3	-2.0	3509
San Diego Intl	11.1	0.0	2724
San Antonio Intl	20.7	1.0	678
Savannah Hilton Head Intl	18.3	-1.0	753
South Bend Rgnl	21.1	14.0	10
Louisville International Airport	16.4	-2.0	1117
Seattle Tacoma Intl	10.7	-1.0	3904
San Francisco Intl	12.9	0.0	13230
Norman Y Mineta San Jose Intl	10.1	-1.0	328
Salt Lake City Intl	9.0	-1.0	2458
Sacramento Intl	18.7	2.0	282
John Wayne Arpt Orange Co	7.8	-1.0	819
Sarasota Bradenton Intl	7.3	-3.0	1203
Lambert St Louis Intl	16.0	-1.0	4168
Syracuse Hancock Intl	14.4	-2.0	1711
Tampa Intl	12.1	-1.0	7407
Tulsa Intl	34.9	8.0	299
Cherry Capital Airport	22.1	-3.0	96
Mc Ghee Tyson	28.5	0.0	579
NW Arkansas Regional	6.5	-5.0	1011

```
# Arrival delay summary
arrival_delay_summary <- flights_dt[
```

```
  !is.na(arr_delay), # Exclude rows with NA arr_delay
  .(
    mean_arr_delay = mean(arr_delay, na.rm = TRUE),
    median_arr_delay = median(arr_delay, na.rm = TRUE),
    num_flights = .N
  ), by = dest
][
  num_flights >= 10
][
  order(-mean_arr_delay)
][
  airports_dt, on = .(dest = faa), nomatch = 0
][
  , .(name, mean_arr_delay, median_arr_delay, num_flights) # Use the column directly after the
]

if (nrow(arrival_delay_summary) == 0) {
  arrival_delay_summary <- data.table(name = NA, mean_arr_delay = NA, median_arr_delay = NA, nu
}

kable(arrival_delay_summary, align = 'c')
```

name	mean_arr_delay	median_arr_delay	num_flights
Albuquerque International Sunport	4.3818898	-5.5	254
Nantucket Mem	4.8522727	-3.0	264
Albany Intl	14.3971292	-4.0	418
Hartsfield Jackson Atlanta Intl	11.3001128	-1.0	16837
Austin Bergstrom Intl	6.0199088	-5.0	2411
Asheville Regional Airport	8.0038314	-1.0	261
Bradley Intl	7.0485437	-10.0	412
Bangor Intl	8.0279330	-9.0	358
Birmingham Intl	16.8773234	-2.0	269
Nashville Intl	11.8124589	-2.0	6084
General Edward Lawrence Logan Intl	2.9143922	-9.0	15022
Burlington Intl	8.9509960	-4.0	2510
Buffalo Niagara Intl	8.9459519	-5.0	4570
Bob Hope	8.1756757	-3.0	370
Baltimore Washington Intl	10.7267338	-5.0	1687
Gallatin Field	7.6000000	-2.0	35
Columbia Metropolitan	41.7641509	28.0	106
Akron Canton Regional Airport	19.6983373	3.0	842
Charlottesville-Albemarle	9.5000000	-5.0	46
Charleston Afb Intl	10.5929685	-4.0	2759
Cleveland Hopkins Intl	9.1816113	-5.0	4394
Charlotte Douglas Intl	7.3603189	-3.0	13674
Port Columbus Intl	10.6013229	-3.0	3326
Yeager	14.6716418	-1.5	134
Cincinnati Northern Kentucky Intl	15.3645638	-3.0	3725
James M Cox Dayton Intl	12.6804861	-3.0	1399
Ronald Reagan Washington Natl	9.0669520	-2.0	9111

name	mean_arr_delay	median_arr_delay	num_flights
Denver Intl	8.6065002	-2.0	7169
Dallas Fort Worth Intl	0.3221268	-9.0	8388
Des Moines Intl	19.0057361	0.0	523
Detroit Metro Wayne Co	5.4299635	-7.0	9031
Eagle Co Rgnl	6.3043478	-4.0	207
Key West Intl	6.3529412	7.0	17
Fort Lauderdale Hollywood Intl	8.0821215	-3.0	11897
Gerald R Ford Intl	18.1895604	1.0	728
Piedmont Triad	14.1126005	-2.0	1492
Greenville-Spartanburg International	15.9354430	-0.5	790
Yampa Valley	2.1428571	2.0	14
Honolulu Intl	-1.3651926	-7.0	701
William P Hobby	7.1761882	-4.0	2083
Washington Dulles Intl	13.8642021	-3.0	5383
George Bush Intercontinental	4.2407904	-5.0	7085
Wilmington Intl	4.6355140	-7.0	107
Indianapolis Intl	9.9404341	-3.0	1981
Jackson Hole Airport	28.0952381	15.0	21
Jacksonville Intl	11.8448342	-2.0	2623
Mc Carran Intl	0.2577285	-8.0	5952
Los Angeles Intl	0.5471109	-7.0	16026
Long Beach	-0.0620272	-10.0	661
Kansas City Intl	14.5140584	0.0	1885
Orlando Intl	5.4546431	-5.0	13967
Chicago Midway Intl	12.3642236	-1.0	4025
Memphis Intl	10.6453144	-2.5	1686
Manchester Regional Airport	14.7875536	-3.0	932
Miami Intl	0.2990598	-9.0	11593
General Mitchell Intl	14.1672204	0.0	2709
Dane Co Rgnl Truax Fld	20.1960432	1.0	556
Minneapolis St Paul Intl	7.2701689	-5.0	6929
Louis Armstrong New Orleans Intl	6.4901750	-6.0	3715
Montrose Regional Airport	1.7857143	-10.5	14
Martha\'s Vineyard	-0.2857143	-11.0	210
Myrtle Beach Intl	4.6034483	-13.0	58
Metropolitan Oakland Intl	3.0776699	-9.0	309
Will Rogers World	30.6190476	16.0	315
Eppley Afld	14.6988984	-2.0	817
Chicago Ohare Intl	5.8766148	-8.0	16566
Norfolk Intl	10.9490934	-4.0	1434
Palm Beach Intl	8.5629721	-3.0	6487
Portland Intl	5.1415797	-5.0	1342
Philadelphia Intl	10.1271901	-3.0	1541
Phoenix Sky Harbor Intl	2.0970473	-6.0	4606
Pittsburgh Intl	7.6809905	-5.0	2746
Palm Springs Intl	-12.7222222	-13.5	18
Theodore Francis Green State	16.2346369	1.0	358
Portland Intl Jetport	11.6604021	-4.0	2288
Raleigh Durham Intl	10.0523810	-3.0	7770

name	mean_arr_delay	median_arr_delay	num_flights
Richmond Intl	20.1112532	1.0	2346
Greater Rochester Intl	11.5606446	-5.0	2358
Southwest Florida Intl	3.2381496	-5.0	3502
San Diego Intl	3.1391657	-5.0	2709
San Antonio Intl	6.9453718	-9.0	659
Savannah Hilton Head Intl	15.1295060	-1.0	749
South Bend Rgnl	6.5000000	-3.5	10
Louisville International Airport	12.6693841	-2.0	1104
Seattle Tacoma Intl	-1.0990991	-11.0	3885
San Francisco Intl	2.6728915	-8.0	13173
Norman Y Mineta San Jose Intl	3.4481707	-7.0	328
Salt Lake City Intl	0.1762546	-8.0	2451
Sacramento Intl	12.1099291	4.0	282
John Wayne Arpt Orange Co	-7.8682266	-11.0	812
Sarasota Bradenton Intl	3.0824313	-5.0	1201
Lambert St Louis Intl	11.0784645	-3.0	4142
Syracuse Hancock Intl	8.9039250	-5.0	1707
Tampa Intl	7.4085250	-4.0	7390
Tulsa Intl	33.6598639	14.0	294
Cherry Capital Airport	12.9684211	-10.0	95
Mc Ghee Tyson	24.0692042	2.0	578
NW Arkansas Regional	7.4657258	-2.0	992

```
# Calculate speed
flights_dt[, speed_mph := distance / (air_time / 60)]
flights_dt <- flights_dt[!is.na(speed_mph)] # Exclude rows with NA speeds

# Find the aircraft with the fastest average speed
fastest_aircraft <- flights_dt[
  , .(
    avg_speed = mean(speed_mph, na.rm = TRUE),
    num_flights = .N
  ), by = tailnum
][
  order(-avg_speed)
][
  1
]

if (nrow(fastest_aircraft) == 0) {
  fastest_model <- data.table(model = NA, avg_speed = NA, num_flights = NA)
} else {
  # Join with planes to get the model
  fastest_model <- fastest_aircraft[
    planes_dt, on = "tailnum", nomatch = 0
  ][
    , .(model, avg_speed, num_flights)
  ]
```

```
}

# Display the fastest model
fastest_model
```

```
     model avg_speed num_flights
    <char>     <num>       <int>
1: 777-222  500.8163           1
```

The computed results can be compared back to HW-4. They are precisely the same.