Homework 1

Andey Nunes, MS
Jordan Hilton
Mengyu Li
Peter Boss

January 21, 2019

Document Setup

The first step for this week is to set up the R Markdown document options. Be sure that prior to executing code in this document that the following R packages are installed and updated in your R session:

- knitr
- pander
- readxl
- tidyverse

Tidyverse is an ecosystem of packages that work nicely together for data science tools. When the tidyverse package is installed, all the packages and their dependencies are automatically loaded into the R session. The packages included in the tidyverse package are listed here.

broom, cli, crayon, dplyr, dbplyr, forcats, ggplot2, haven, hms, httr, jsonlite, lubridate, magrittr, modelr, purrr, readr, readxl (>=, reprex, rlang, rstudioapi, rvest, stringr, tibble, tidyr, xml2, tidyverse

Next step, load the data sets for the homework. Summaries are included in the appendix.

```
catalog <- read_excel("catalog.xls")
customers <- read_excel("customers.xls")
order_lines <- read_excel("order_lines.xls")#, sheet = "Sheet 1")
orders <- read_excel("orders.xls")
```

At first try, the `order_lines` data table did not load properly. We had to open the file in Excel to find that there are three sheets, two of which are pivot tables of the sheet containing all the data. These pivot tables are ahead of the actual data, so we manually reordered the sheets to put the data (labeled as *Sheet1* in .xls file). Next, we had to fix the column `customer_id` because it had a typo in the VLOOKUP command file name argument that referenced that information from the `orders` data file. Again, in Excel, that formula was fixed, and the cell reference for that column updated. Then the file was resaved and used in our analysis.

Custom functions

This section is for building some custom functions that will come in handy later

```
# count the number of missing data entries
countNA <- function(x) {sum(is.na(x)) }

# get the range of a numeric vector by taking the difference
# between the high and low values from the range output
# if the vector is not numeric, then provide NA
get_range <- function(x) {ifelse(is.numeric(x), diff(range(x)), NA)}

# This function creates the generic structure for the tables in Part B.
# The variable_class use of map_chr() will throw an error on the data-time
# object because that class has multiple assignments
```

```r
# value_type is temporarily NA, reassign one of: "question", "answer", "link"

make_partBtable <- function(x){
   df <- tibble(variable_name = names(x),
                variable_type = NA,
                variable_class = map_chr(x, class),
                count_missing = map_int(x, countNA),
                count_unique = map_dbl(x, ~length(unique(.x)) ), # this
                # column can be removed from the tables using #
                variable_range = map_dbl(x, get_range))

   return(df)

}
```

Homework Questions

Part A: General Questions

1. Key business questions

- What is the company's revenue?
- What is the company's profit?
- How profitable is each product?
- How many orders are there for each product?
- How many active customers are there?
- Which market segment (international, domestic, or military) has the most sales growth over time?
-

2. How does each table relate to answering those questions?

- The catalog table lists each product along with information about that product (such as price, manufacturer, and name).
- The customers table lists each of the company's customers, along with information about that customer (such as location and name).
- The orders table has one record for every order a customer made, with the total cost of that order and information about the number of items in the order and its shipping weight.
- The order_lines table has one record for each different item that was purchased in a single order, along with links to the order.
- The `orders` data has an `order_date` and a `total_amount` for each unique `order_id`, which can be used to join the `order_lines` table to capture the `customer_id`. The `bt_state` field can be reclassified as one of three categories: "domestic" for US states, "international" indicated by the value `INTL`, and "military" indicated by the value "APO". This rebinned field can be used to classify the orders by market segment, using a table made from joining on the `customer_id` field. This final table can be summarized for total order amounts by month or quarter for each market segment then visualized on a timeline to spot trends in sales.

3. How do I have to link the tables in order to be able to answer those questions?

Part B: Specific Questions

For each data set, we include a table that gives the field (variable) names, whether they are a *link*, *answer* or *question* field, the data class, how many missing observations, how many unique entries are in each column, and if the variable is numeric, a range is given.

The reason to include a column for unique entries is to identify two types of columns: unique identifiers, and fields that contain only one kind of entry. If the number of unique entries in a field is equal to the number of observations in the data table, then that variable can be considered a unique identifier and should not be considered to be a number for calculations nor a factor for grouping, rather it is a way to link unique rows between two separate data frames. A perfect exampe of this is the customer id field or the order id. Occasionally, date-time columns will yield this, but its also a good check for duplicate values in those types of columns. When a field contains only one unique entry (NA values are considered a type of entry) then it indicates a value that is descriptive of the entire table and is meaningless in differentiaing observations. It may not be a useless variable, because it could be indicative that our table is a filtered subset of a much larger table where that field had other values, but we would not know unless we knew how the table we are looking at was constructed. The large numbers of unique values also gives us a sense of the size of state space for that field and will indicate where descritizing actions may need to be focused.

Catalog

This data set has 761 observations on 7 variables with details as follows:

```
catalog_table <- make_partBtable(catalog)
catalog_table$variable_type <- c("link", "link", "answer", "question",
                                 "question", "question", "answer")


# pander(catalog_table, caption = "Catalog Data Table Details")
kable(catalog_table, caption = "Catalog Data Table Details")
```

Table 1: Catalog Data Table Details

variable_name	variable_type	variable_class	count_missing	count_unique	variable_range
id	link	numeric	0	761	818
product_code	link	character	1	761	NA
catalog_price	answer	numeric	0	134	654
category1	question	character	645	10	NA
manufact_id	question	numeric	0	5	8
vendor_id	question	numeric	0	5	8
name	answer	character	1	756	NA

Customers

Many of these fields are character string fields or identification fields. While the range values are given, they are not applicable to this data table.

This data set has 22070 observations on 10 variables with details as follows:

```
customers_table <- make_partBtable(customers)


customers_table$variable_type <- c("link", "link", rep("question", 6), "question or answer", "link")
# id variables and customer code are "links"
```

```
# names and bt_* are questions of who and where

#pander(customers_table, caption = "Customers Data Table Details")
kable(customers_table, caption = "Customers Data Table Details")
```

Table 2: Customers Data Table Details

variable_name	variable_type	variable_class	count_missing	count_unique	variable_range
cust_id	link	numeric	0	22070	22482
merchant_id	link	numeric	0	2	1
firstName	question	character	12070	502	NA
lastName	question	character	12070	1001	NA
bt_city	question	character	1	9032	NA
bt_state	question	character	137	67	NA
bt_country	question	character	0	79	NA
bt_zip	question	character	0	12434	NA
cc_type	question or answer	character	0	4	NA
custcode	link	character	0	22069	NA

Order_lines

This data set has 0 observations on 21 variables with details as follows:

```
order_lines_table <- tibble(
  variable_name = names(order_lines),
  variable_type = c("link", "question", # which line in the order?
                    "link", "question", # what line status
                    "question & answer", # time intervals, when
                    rep("answer", 2), # how many
                    "question & answer", # time intervals, when
                    rep("unused", 3),# empty columns
                    "link", "question", # what is the list price
                    rep("unused", 2),# empty columns
                    "link", "questions", # which products
                    rep("question",2),
                    "link", "unused"), # last column is empty
                    # assign one of: "question", "answer", "link"
  variable_class = c(rep("numeric", 3), "character", "date-time",
                     "numeric", "numeric", "date-time",
                     rep("logical", 3), "numeric", "numeric",
                     "logical", "logical", "numeric", "character",
                     rep("numeric", 3), "logical"),
  count_missing = map_int(order_lines, countNA),
  count_unique = map_dbl(order_lines, ~length(unique(.x))),
  variable_range = map_dbl(order_lines, get_range))


pander(order_lines_table, caption = "Order_lines Data Table Details")
```

Table 3: Order_lines Data Table Details (continued below)

variable_name	variable_type	variable_class	count_missing
order_id	link	numeric	2

variable_name	variable_type	variable_class	count_missing	count_unique	variable_range
order_line	question	numeric	2	23266	NA
customer_id	link	numeric	14	22	NA
line_status	question	character	2	22035	NA
line_status_date	question & answer	date-time	2	5	NA
order_qty	answer	numeric	1	1843	NA
shipped_qty	answer	numeric	1	43	NA
bo_exp_date	question & answer	date-time	7055	35	NA
internal_note	unused	logical	31233	186	NA
spec_proc_note	unused	logical	31233	1	NA
spec_proc_id	unused	logical	31233	1	NA
order_line_id	link	numeric	2	1	NA
list_price	question	numeric	2	31232	NA
gift_note	unused	logical	31233	272	NA
distrib_id	unused	logical	31233	1	NA
product_id	link	numeric	2	1	NA
Product name	questions	character	1159	678	NA
Shipped Total	question	numeric	2	651	NA
Ordered Total	question	numeric	2	757	NA
format_id	link	numeric	2	912	NA
options	unused	logical	31233	7	NA
				1	NA

Orders

This data set has 23256 observations on 18 variables with details as follows:

```
orders_table <- tibble(
  variable_name = names(orders),
```

```
  variable_type = c(rep("link",2), "question", #when
                    rep("link",2),
                    rep("question", 2),# which
                    rep("answer", 7),# how much |total
                    rep("question",4)), # when
  # assign one of: "question", "answer", "link"
  variable_class = c("numeric", "numeric","date-time", "character",
                     "numeric", "character", "character",
                     "numeric", "character",rep("numeric", 5), "date-time",
                     "numeric", "logical", "logical"),
  count_missing = map_int(orders, countNA),
  count_unique = map_dbl(orders, ~length(unique(.x))),
  variable_range = map_dbl(orders, get_range))


#pander(orders_table, caption = "Orders Data Table Details")
kable(orders_table, caption = "Orders Data Table Details")
```

Table 5: Orders Data Table Details

variable_name	variable_type	variable_class	count_missing	count_unique	variable_range
order_id	link	numeric	0	23256	23575
merchant_id	link	numeric	0	2	1
order_date	question	date-time	0	2641	NA
po_number	link	character	22742	442	NA
cust_id	link	numeric	0	22034	32482
order_status	question	character	0	4	NA
ship_method	question	character	186	16	NA
items_amount	answer	numeric	0	2105	9590
amt_bracket	answer	character	0	4	NA
total_weight	answer	numeric	0	444	483
total_ship	answer	numeric	0	2298	631
total_hand	answer	numeric	0	1	0
total_tax	answer	numeric	0	1	0
total_amount	answer	numeric	0	10444	9584
order_status_date	question	date-time	0	1801	NA
send_inv_to_bill	question	numeric	0	2	1
coupon_code	question	logical	23256	1	NA
spec_instr	question	logical	23256	1	NA

Part C. Filter/Select Operations

For all these answers indicate clearly what fields you used, and why you chose those particular fields. If there were other fields you could have considered, indicate why you did not choose those.

4. Top 10 states for orders by dollar volume

We need the "state" field from the customers table, along with summed order totals from the order table, so we'll need to join those two tables and group by state.

```
top10states<- customers %>%
  inner_join(orders, by="cust_id") %>%   ## join the customers and orders table using the field cust_id
```

```
  filter(bt_country == "United States") %>%  ##filter to only orders from customers in the US
  select(bt_state, total_amount)        ##reduces the resulting join into the two fields of interest

top10states <- aggregate(top10states$total_amount, list(state=top10states$bt_state), sum)  ##group by s

top10states <- arrange(top10states, -top10states$x) %>%  ##orders the resulting list by order volume de
  head(10)                             ## shows the top 10 results

names(top10states)<-list("State", "Order Volume")

  pander(top10states)
```

State	Order Volume
CA	174920
TX	128744
FL	88951
NY	84106
VA	72133
NC	56886
WA	56838
IL	54843
OR	54689
PA	50150

```
# this is just given as a thru-pipe example of the code above
# we can leave this out of the assignment  by placing
# "eval=FALSE, include=FALSE" after the code chunk name

# join customers and orders using cust_id link
# filter out the two non-state labels from bt_state
# pull out the two fields of interest and group the data by state
# summarize the observations to get a total by state and arrange in
# descending order, then rename the state column and keep only rows 1:10
orders_top_states <- customers %>%
  inner_join(orders, by="cust_id") %>%
  filter(bt_state != "APO",
         bt_state != "INTL") %>%
  select(bt_state, total_amount) %>%
  group_by(bt_state) %>%
  summarize(order_volume = sum(total_amount)) %>%
  arrange(desc(order_volume)) %>%
  rename(state = bt_state) %>%
  slice(1:10)

kable(orders_top_states[1:10,], caption = "Top 10 states for orders by dollar volume")
```

Table 7: Top 10 states for orders by dollar volume

state	order_volume
CA	174920
TX	128754
FL	89137

state	order_volume
NY	84202
VA	72133
NC	56886
WA	56838
OR	55147
IL	54843
PA	50150

5. Top 10 countries for orders by dollar volume

```
head(orders)
```

```
## # A tibble: 6 x 18
##   order_id merchant_id order_date          po_number cust_id order_status
##      <dbl>       <dbl> <dttm>              <chr>       <dbl> <chr>
## 1    14035           1 2003-10-17 00:00:00 <NA>        10034 S
## 2    14034           1 2003-10-16 00:00:00 <NA>        10033 S
## 3    14033           1 2003-10-16 00:00:00 <NA>        10032 S
## 4    14032           1 2003-10-16 00:00:00 <NA>        10031 S
## 5    14031           1 2003-10-16 00:00:00 <NA>        10030 S
## 6    14030           1 2003-10-16 00:00:00 <NA>        10029 S
## # ... with 12 more variables: ship_method <chr>, items_amount <dbl>,
## #   amt_bracket <chr>, total_weight <dbl>, total_ship <dbl>,
## #   total_hand <dbl>, total_tax <dbl>, total_amount <dbl>,
## #   order_status_date <dttm>, send_inv_to_bill <dbl>, coupon_code <lgl>,
## #   spec_instr <lgl>
```

```
head(customers)
```

```
## # A tibble: 6 x 10
##   cust_id merchant_id firstName lastName bt_city bt_state bt_country bt_zip
##     <dbl>       <dbl> <chr>     <chr>    <chr>   <chr>    <chr>      <chr>
## 1   20696           2 Kristina  Chung    Piedmo~ OK       United St~ 73078
## 2   15465           1 Paige     Chen     Cincin~ OH       United St~ 45227
## 3   19830           2 Sherri    Melton   Shelby~ TN       United St~ 37160
## 4   25532           1 Gretchen  Hill     North ~ AZ       United St~ 86052
## 5   16044           1 Karen     Puckett  Petawa~ ON       Canada     K8H 2~
## 6   32394           1 Patrick   Song     Winche~ OR       United St~ 97495
## # ... with 2 more variables: cc_type <chr>, custcode <chr>
```

```
top10_Order_Dollar_byCountry <- inner_join(orders, customers, by = c('cust_id')) %>%
  group_by(bt_country) %>%
  summarise(totDollarVol =  sum(total_amount)) %>%
  arrange(desc(totDollarVol)) %>%
  top_n(10)
```

```
## Selecting by totDollarVol
```

```
top10_Order_Dollar_byCountry
```

```
## # A tibble: 10 x 2
##    bt_country     totDollarVol
##    <chr>                 <dbl>
##  1 United States      1695959.
```

```
##  2 Canada               75096.
##  3 Singapore            22706.
##  4 Australia            17456.
##  5 United Kingdom       14136.
##  6 France               11361.
##  7 Qatar                 9081
##  8 Germany               9063.
##  9 Spain                 7924.
## 10 Denmark               6439.
```

6. Top 10 selling products by units; then by dollar volume

```
head(order_lines)
```

```
## # A tibble: 6 x 21
##   order_id order_line customer_id line_status line_status_date    order_qty
##      <dbl>      <dbl>       <dbl> <chr>       <dttm>                  <dbl>
## 1    34462          1       29522 S           2009-12-18 00:00:00         1
## 2    26061          1       21537 S           2007-07-31 00:00:00         6
## 3    35964          1       30924 S           2010-08-31 00:00:00         5
## 4    35217          1       30246 S           2010-04-27 00:00:00         2
## 5    14053          1       10052 S           2003-10-21 00:00:00         2
## 6    15586          1       11518 S           2004-09-24 00:00:00         2
## # ... with 15 more variables: shipped_qty <dbl>, bo_exp_date <dttm>,
## #   internal_note <lgl>, spec_proc_note <lgl>, spec_proc_id <lgl>,
## #   order_line_id <dbl>, list_price <dbl>, gift_note <lgl>,
## #   distrib_id <lgl>, product_id <dbl>, `Product name` <chr>, `Shipped
## #   Total` <dbl>, `Ordered Total` <dbl>, format_id <dbl>, options <lgl>
```

```
Top10_SellProduct_ByUnit <- order_lines %>%
  group_by(product_id) %>%
  summarise(totUnit =  sum(shipped_qty)) %>%
  arrange(desc(totUnit)) %>%
  top_n(10)
```

```
## Selecting by totUnit
```

```
Top10_SellProduct_ByUnit
```

```
## # A tibble: 10 x 2
##    product_id totUnit
##         <dbl>   <dbl>
##  1        415     682
##  2        652     475
##  3        560     391
##  4        319     374
##  5        358     373
##  6        336     357
##  7        355     349
##  8        414     339
##  9        411     322
## 10        312     313
```

```
Top10_SellProduct_ByDollar <- order_lines %>%
  group_by(product_id) %>%
  summarise(totDollar =  sum(`Shipped Total`)) %>%
```

```
  arrange(desc(totDollar)) %>%
  top_n(10)
```

```
## Selecting by totDollar
```

```
Top10_SellProduct_ByDollar
```

```
## # A tibble: 10 x 2
##    product_id totDollar
##         <dbl>     <dbl>
##  1        411    27507.
##  2        757    21592.
##  3        395    20356.
##  4        336    19116.
##  5        332    18437.
##  6        798    17879.
##  7        408    17739.
##  8        797    17674.
##  9        760    16511.
## 10        321    15541.
```

7. For each of the top two US states and each of the top two countries (excluding the US) in questions 1 and 2, what are the 5 top selling products by units? By dollar volume? (5%)

8. Provide the customer ID's, order dates, and order amounts for all customers who have ordered more than once. (5%)

Part D. Sales increasing strategies

A quick list of sales increasing strategies include;

- We know we have one time and repeat customers, but perhaps are there any other ways to segment customers and offer special promotions to see which customer segments respond to particular sales promotions.
-

References

Appendix

Summary tables

```
# this whole code chunk can be updated to be "include = FALSE"
# the use of head() is redundant since glimpse() shows more of the same information
# but also tells you how many observations are in the data set
# and doesn't truncate the list of variables

pander(summary(catalog), caption = "catalog summary table")
```

Table 8: catalog summary table (continued below)

id	product_code	catalog_price	category1
Min. : 307	Length:761	Min. : 0	Length:761
1st Qu.: 525	Class :character	1st Qu.: 18	Class :character
Median : 728	Mode :character	Median : 34	Mode :character
Mean : 725	NA	Mean : 49	NA
3rd Qu.: 930	NA	3rd Qu.: 57	NA
Max. :1125	NA	Max. :654	NA

manufact_id	vendor_id	name
Min. :0.0	Min. :0.0	Length:761
1st Qu.:1.0	1st Qu.:1.0	Class :character
Median :1.0	Median :1.0	Mode :character
Mean :1.2	Mean :1.2	NA
3rd Qu.:1.0	3rd Qu.:1.0	NA
Max. :8.0	Max. :8.0	NA

```
head(catalog)
```

```
## # A tibble: 6 x 7
##      id product_code catalog_price category1  manufact_id vendor_id name
##   <dbl> <chr>                <dbl> <chr>            <dbl>     <dbl> <chr>
## 1   446 G79761                9.95 accessori~           1         1 Exchan~
## 2   455 plastic               0    <NA>                 1         1 Plasti~
## 3   445 G75329               12.0  fishing              1         1 Silver~
## 4   444 G75328               11.0  fillet               1         1 Silver~
## 5   443 G75231               13.0  fillet               1         1 "Gator~
## 6   442 G75230               12.0  fillet               1         1 "Gator~
```

```
glimpse(catalog)
```

```
## Observations: 761
## Variables: 7
## $ id            <dbl> 446, 455, 445, 444, 443, 442, 438, 439, 440, 441...
## $ product_code  <chr> "G79761", "plastic", "G75329", "G75328", "G75231...
## $ catalog_price <dbl> 9.9, 0.0, 11.9, 10.9, 12.9, 11.9, 9.5, 6.0, 6.0,...
## $ category1     <chr> "accessories", NA, "fishing", "fillet", "fillet"...
## $ manufact_id   <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ vendor_id     <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, ...
## $ name          <chr> "Exchange-A-Blade Sheath for 7 inch saw", "Plast...
```

```
pander(summary(customers), caption = "customers summary table")
```

Table 10: customers summary table (continued below)

cust_id	merchant_id	firstName	lastName
Min. :10000	Min. :1.00	Length:22070	Length:22070
1st Qu.:15930	1st Qu.:1.00	Class :character	Class :character
Median :21448	Median :1.00	Mode :character	Mode :character
Mean :21408	Mean :1.05	NA	NA
3rd Qu.:26965	3rd Qu.:1.00	NA	NA

cust_id	merchant_id	firstName	lastName
Max. :32482	Max. :2.00	NA	NA

Table 11: Table continues below

bt_city	bt_state	bt_country	bt_zip
Length:22070	Length:22070	Length:22070	Length:22070
Class :character	Class :character	Class :character	Class :character
Mode :character	Mode :character	Mode :character	Mode :character
NA	NA	NA	NA
NA	NA	NA	NA
NA	NA	NA	NA

cc_type	custcode
Length:22070	Length:22070
Class :character	Class :character
Mode :character	Mode :character
NA	NA
NA	NA
NA	NA

```
head(customers)
```

```
## # A tibble: 6 x 10
##   cust_id merchant_id firstName lastName bt_city bt_state bt_country bt_zip
##     <dbl>       <dbl> <chr>     <chr>    <chr>   <chr>    <chr>      <chr>
## 1   20696           2 Kristina  Chung    Piedmo~ OK       United St~ 73078
## 2   15465           1 Paige     Chen     Cincin~ OH       United St~ 45227
## 3   19830           2 Sherri    Melton   Shelby~ TN       United St~ 37160
## 4   25532           1 Gretchen  Hill     North ~ AZ       United St~ 86052
## 5   16044           1 Karen     Puckett  Petawa~ ON       Canada     K8H 2~
## 6   32394           1 Patrick   Song     Winche~ OR       United St~ 97495
## # ... with 2 more variables: cc_type <chr>, custcode <chr>
```

```
glimpse(customers)
```

```
## Observations: 22,070
## Variables: 10
## $ cust_id     <dbl> 20696, 15465, 19830, 25532, 16044, 32394, 29572, 3...
## $ merchant_id <dbl> 2, 1, 2, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ firstName   <chr> "Kristina", "Paige", "Sherri", "Gretchen", "Karen"...
## $ lastName    <chr> "Chung", "Chen", "Melton", "Hill", "Puckett", "Son...
## $ bt_city     <chr> "Piedmont", "Cincinnati", "Shelbyville", "North ri...
## $ bt_state    <chr> "OK", "OH", "TN", "AZ", "ON", "OR", "GA", "VA", "K...
## $ bt_country  <chr> "United States", "United States", "United States",...
## $ bt_zip      <chr> "73078", "45227", "37160", "86052", "K8H 2X3", "97...
## $ cc_type     <chr> "Visa", "Visa", "Mastercard", "Visa", "Visa", "Mas...
## $ custcode    <chr> "P20696", "G15465", "P19830", "G25532", "G16044", ...
```

```
pander(summary(order_lines), caption = "order_lines summary table")
```

Table 13: order_lines summary table (continued below)

order_id	order_line	customer_id	line_status
Min. : 0	Min. : 1.0	Min. : 0	Length:31233
1st Qu.:19842	1st Qu.: 1.0	1st Qu.:15484	Class :character
Median :25622	Median : 1.0	Median :20974	Mode :character
Mean :25707	Mean : 1.4	Mean :21083	NA
3rd Qu.:31514	3rd Qu.: 2.0	3rd Qu.:26584	NA
Max. :37575	Max. :21.0	Max. :32482	NA
NA's :2	NA's :2	NA's :14	NA

Table 14: Table continues below

line_status_date	order_qty	shipped_qty
Min. :2003-10-10 00:00:00	Min. : 0	Min. : 0
1st Qu.:2006-05-01 00:00:00	1st Qu.: 1	1st Qu.: 0
Median :2007-06-05 00:00:00	Median : 1	Median : 1
Mean :2007-08-02 15:07:40	Mean : 3	Mean : 2
3rd Qu.:2008-12-15 00:00:00	3rd Qu.: 1	3rd Qu.: 1
Max. :2011-01-21 00:00:00	Max. :41409	Max. :28257
NA's :2	NA's :1	NA's :1

Table 15: Table continues below

bo_exp_date	internal_note	spec_proc_note	spec_proc_id
Min. :1899-12-31 00:00:00	Mode:logical	Mode:logical	Mode:logical
1st Qu.:1899-12-31 00:00:00	NA's:31233	NA's:31233	NA's:31233
Median :1899-12-31 00:00:00	NA	NA	NA
Mean :1903-04-11 12:05:08	NA	NA	NA
3rd Qu.:1899-12-31 00:00:00	NA	NA	NA
Max. :2008-02-15 00:00:00	NA	NA	NA
NA's :7055	NA	NA	NA

Table 16: Table continues below

order_line_id	list_price	gift_note	distrib_id	product_id
Min. : 90	Min. : 0	Mode:logical	Mode:logical	Min. : 307
1st Qu.: 8174	1st Qu.: 18	NA's:31233	NA's:31233	1st Qu.: 408
Median :15982	Median : 35	NA	NA	Median : 560
Mean :15956	Mean : 43	NA	NA	Mean : 586
3rd Qu.:23790	3rd Qu.: 55	NA	NA	3rd Qu.: 744
Max. :31597	Max. :361	NA	NA	Max. :1101
NA's :2	NA's :2	NA	NA	NA's :2

Product name	Shipped Total	Ordered Total	format_id	options
Length:31233	Min. : 0	Min. : 0	Min. : 0	Mode:logical
Class :character	1st Qu.: 0	1st Qu.: 20	1st Qu.: 0	NA's:31233
Mode :character	Median : 22	Median : 37	Median : 0	NA
NA	Mean : 36	Mean : 54	Mean : 0	NA
NA	3rd Qu.: 46	3rd Qu.: 59	3rd Qu.: 0	NA
NA	Max. :6982	Max. :9590	Max. :11	NA
NA	NA's :2	NA's :2	NA's :2	NA

```
head(order_lines)
```

```
## # A tibble: 6 x 21
##   order_id order_line customer_id line_status line_status_date    order_qty
##      <dbl>      <dbl>       <dbl> <chr>       <dttm>                 <dbl>
## 1    34462          1       29522 S           2009-12-18 00:00:00        1
## 2    26061          1       21537 S           2007-07-31 00:00:00        6
## 3    35964          1       30924 S           2010-08-31 00:00:00        5
## 4    35217          1       30246 S           2010-04-27 00:00:00        2
## 5    14053          1       10052 S           2003-10-21 00:00:00        2
## 6    15586          1       11518 S           2004-09-24 00:00:00        2
## # ... with 15 more variables: shipped_qty <dbl>, bo_exp_date <dttm>,
## #   internal_note <lgl>, spec_proc_note <lgl>, spec_proc_id <lgl>,
## #   order_line_id <dbl>, list_price <dbl>, gift_note <lgl>,
## #   distrib_id <lgl>, product_id <dbl>, `Product name` <chr>, `Shipped
## #   Total` <dbl>, `Ordered Total` <dbl>, format_id <dbl>, options <lgl>
```

```
glimpse(order_lines)
```

```
## Observations: 31,233
## Variables: 21
## $ order_id         <dbl> 34462, 26061, 35964, 35217, 14053, 15586, 167...
## $ order_line       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 2, 1, 1, 1, 1, 1, ...
## $ customer_id      <dbl> 29522, 21537, 30924, 30246, 10052, 11518, 127...
## $ line_status      <chr> "S", "S", "S", "S", "S", "S", "S", "S", "S", ...
## $ line_status_date <dttm> 2009-12-18, 2007-07-31, 2010-08-31, 2010-04-...
## $ order_qty        <dbl> 1, 6, 5, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, ...
## $ shipped_qty      <dbl> 11, 6, 5, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2, 2,...
## $ bo_exp_date      <dttm> 1899-12-31, 1899-12-31, 1899-12-31, 1899-12-...
## $ internal_note    <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ spec_proc_note   <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ spec_proc_id     <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ order_line_id    <dbl> 27539, 16544, 29509, 28514, 163, 2217, 3732, ...
## $ list_price       <dbl> 18, 18, 16, 19, 18, 18, 18, 18, 18, 18, 18, 1...
## $ gift_note        <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ distrib_id       <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
## $ product_id       <dbl> 307, 307, 307, 307, 307, 307, 307, 307, 307, ...
## $ `Product name`   <chr> "Carbide Cutter Insert Replacements", "Carbid...
## $ `Shipped Total`  <dbl> 197, 108, 80, 38, 36, 36, 36, 36, 36, 36, 36,...
## $ `Ordered Total`  <dbl> 18, 108, 80, 38, 36, 36, 36, 36, 36, 36, 36, ...
## $ format_id        <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ...
## $ options          <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N...
```

```
pander(summary(orders), caption = "orders summary table")
```

Table 18: orders summary table (continued below)

order_id	merchant_id	order_date	po_number
Min. :14000	Min. :1.00	Min. :2003-10-10 00:00:00	Length:23256
1st Qu.:20134	1st Qu.:1.00	1st Qu.:2006-04-28 00:00:00	Class :character
Median :25948	Median :1.00	Median :2007-07-02 00:00:00	Mode :character
Mean :25918	Mean :1.05	Mean :2007-08-11 16:51:42	NA
3rd Qu.:31761	3rd Qu.:1.00	3rd Qu.:2008-12-19 00:00:00	NA
Max. :37575	Max. :2.00	Max. :2011-01-21 00:00:00	NA

Table 19: Table continues below

cust_id	order_status	ship_method	items_amount
Min. : 0	Length:23256	Length:23256	Min. : 0
1st Qu.:15778	Class :character	Class :character	1st Qu.: 28
Median :21302	Mode :character	Mode :character	Median : 48
Mean :21295	NA	NA	Mean : 73
3rd Qu.:26849	NA	NA	3rd Qu.: 80
Max. :32482	NA	NA	Max. :9590

Table 20: Table continues below

amt_bracket	total_weight	total_ship	total_hand	total_tax
Length:23256	Min. : 0	Min. : 0	Min. :0	Min. :0
Class :character	1st Qu.: 1	1st Qu.: 7	1st Qu.:0	1st Qu.:0
Mode :character	Median : 2	Median : 8	Median :0	Median :0
NA	Mean : 3	Mean : 11	Mean :0	Mean :0
NA	3rd Qu.: 3	3rd Qu.: 10	3rd Qu.:0	3rd Qu.:0
NA	Max. :483	Max. :631	Max. :0	Max. :0

Table 21: Table continues below

total_amount	order_status_date	send_inv_to_bill	coupon_code
Min. : 6	Min. :2003-10-10 00:00:00	Min. :0.00	Mode:logical
1st Qu.: 36	1st Qu.:2006-05-30 18:00:00	1st Qu.:0.00	NA's:23256
Median : 57	Median :2007-07-12 00:00:00	Median :0.00	NA
Mean : 84	Mean :2007-08-21 21:51:27	Mean :0.05	NA
3rd Qu.: 94	3rd Qu.:2008-12-26 00:00:00	3rd Qu.:0.00	NA
Max. :9590	Max. :2011-01-21 00:00:00	Max. :1.00	NA

spec_instr
Mode:logical
NA's:23256
NA
NA
NA
NA

```
head(orders)
```

```
## # A tibble: 6 x 18
##   order_id merchant_id order_date          po_number cust_id order_status
##      <dbl>       <dbl> <dttm>              <chr>       <dbl> <chr>
## 1    14035           1 2003-10-17 00:00:00 <NA>        10034 S
## 2    14034           1 2003-10-16 00:00:00 <NA>        10033 S
## 3    14033           1 2003-10-16 00:00:00 <NA>        10032 S
## 4    14032           1 2003-10-16 00:00:00 <NA>        10031 S
## 5    14031           1 2003-10-16 00:00:00 <NA>        10030 S
## 6    14030           1 2003-10-16 00:00:00 <NA>        10029 S
## # ... with 12 more variables: ship_method <chr>, items_amount <dbl>,
## #   amt_bracket <chr>, total_weight <dbl>, total_ship <dbl>,
## #   total_hand <dbl>, total_tax <dbl>, total_amount <dbl>,
## #   order_status_date <dttm>, send_inv_to_bill <dbl>, coupon_code <lgl>,
## #   spec_instr <lgl>
```

```
glimpse(orders)
```

```
## Observations: 23,256
## Variables: 18
## $ order_id          <dbl> 14035, 14034, 14033, 14032, 14031, 14030, 14...
## $ merchant_id       <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1,...
## $ order_date        <dttm> 2003-10-17, 2003-10-16, 2003-10-16, 2003-10...
## $ po_number         <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
## $ cust_id           <dbl> 10034, 10033, 10032, 10031, 10030, 10029, 10...
## $ order_status      <chr> "S", "S", "S", "S", "S", "S", "S", "S", "S",...
## $ ship_method       <chr> "GND", "3DS", "GND", "GND", "3DS", "1DA", "G...
## $ items_amount      <dbl> 58.9, 8.9, 50.0, 11.9, 9.9, 109.9, 23.9, 40....
## $ amt_bracket       <chr> "C", "A", "B", "B", "A", "D", "B", "B", "A",...
## $ total_weight      <dbl> 2.3, 1.0, 1.0, 1.0, 1.0, 1.0, 1.0, 1.2, 1.0,...
## $ total_ship        <dbl> 5.5, 9.0, 5.2, 5.4, 9.0, 27.3, 5.3, 6.1, 5.4...
## $ total_hand        <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ total_tax         <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ total_amount      <dbl> 64, 18, 55, 17, 19, 137, 29, 46, 15, 23, 29,...
## $ order_status_date <dttm> 2003-10-17, 2003-10-17, 2003-10-17, 2003-10...
## $ send_inv_to_bill  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0,...
## $ coupon_code       <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
## $ spec_instr        <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, ...
```

```
unique_cat <- map_dbl(catalog, ~length(unique(.x)))
kable(unique_cat, caption = "Catalog Data: unique entry counts by data field")
```

Table 23: Catalog Data: unique entry counts by data field

	x
id	761
product_code	761
catalog_price	134
category1	10
manufact_id	5
vendor_id	5
name	756

```
unique_cust <- map_dbl(customers, ~length(unique(.x)))
kable(unique_cust, caption = "Customers Data: unique entry counts by data field")
```

Table 24: Customers Data: unique entry counts by data field

	x
cust_id	22070
merchant_id	2
firstName	502
lastName	1001
bt_city	9032
bt_state	67
bt_country	79
bt_zip	12434
cc_type	4
custcode	22069

```
unique_OL <- map_dbl(order_lines, ~length(unique(.x)))
kable(unique_OL, caption = "Order Lines Data: unique entry counts by data field")
```

Table 25: Order Lines Data: unique entry counts by data field

	x
order_id	23266
order_line	22
customer_id	22035
line_status	5
line_status_date	1843
order_qty	43
shipped_qty	35
bo_exp_date	186
internal_note	1
spec_proc_note	1
spec_proc_id	1
order_line_id	31232
list_price	272
gift_note	1
distrib_id	1
product_id	678
Product name	651
Shipped Total	757
Ordered Total	912
format_id	7
options	1

```
unique_orders <- map_dbl(orders, ~length(unique(.x)))
kable(unique_orders, caption = "Orders Data Table: unique entry counts by data field")
```

Table 26: Orders Data Table: unique entry counts by data field

	x
order_id	23256
merchant_id	2
order_date	2641
po_number	442
cust_id	22034
order_status	4
ship_method	16
items_amount	2105
amt_bracket	4
total_weight	444
total_ship	2298
total_hand	1
total_tax	1
total_amount	10444
order_status_date	1801
send_inv_to_bill	2
coupon_code	1
spec_instr	1