Music Biz

Using SQL & R on the chinook database.

Set working directory and load libraries:

```
setwd("~/Desktop/School/R stuff/r_sql")
library(RSQLite)
library(DBI)
library(tidyverse)
```

Create run_query function so database is only open when in use:

```
run_query <- function(q) {
  conn <- dbConnect(SQLite(), 'chinook.db')
  result <- dbGetQuery(conn, q)
  dbDisconnect(conn)
  return(result)
}
```

Create function to view "tables" and "views"

```
show_tables <- function() {
  q = "SELECT name, type FROM sqlite_master WHERE type IN ('table','view')"
  return(run_query(q))
}

show_tables()
```

```
##             name  type
## 1          album table
## 2         artist table
## 3       customer table
## 4       employee table
## 5          genre table
## 6        invoice table
## 7   invoice_line table
## 8     media_type table
## 9       playlist table
## 10 playlist_track table
## 11         track table
```

A Glimpse into Overall Sales in the US

Want to see which genres sell the most tracks in the USA

```r
query <- "WITH purchased_genres AS
          (
          SELECT il.invoice_line_id, g.name
          FROM invoice_line AS il
          LEFT JOIN track AS t ON il.track_id = t.track_id
          LEFT JOIN genre AS g ON t.genre_id = g.genre_id
          LEFT JOIN invoice AS i ON i.invoice_id = il.invoice_id
          WHERE billing_country = 'USA'
          ),
          genres_with_pct AS
          (
          SELECT
            name,
            COUNT(*) AS num_purchases
          FROM purchased_genres
          GROUP BY name
          ORDER BY 2 DESC
          )
          SELECT
            name,
            num_purchases,
            ROUND(CAST(num_purchases AS FLOAT) /
              (SELECT COUNT(*)
               FROM purchased_genres)*100) AS percent
          FROM genres_with_pct
          GROUP BY name
          ORDER BY 2 DESC
          LIMIT 10;"
```

```r
top_10_g <- run_query(query)
top_10_g
```

```
##                   name num_purchases percent
## 1                 Rock           561      53
## 2  Alternative & Punk           130      12
## 3                Metal           124      12
## 4             R&B/Soul            53       5
## 5                Blues            36       3
## 6          Alternative            35       3
## 7                  Pop            22       2
## 8                Latin            22       2
## 9          Hip Hop/Rap            20       2
## 10                Jazz            14       1
```

The Rock genre appears to have over half of all music sales in the US and would likely be the best option for a new record based on popularity. The options for a new record however only include Punk, Hip-Hop, Blues, and Pop.

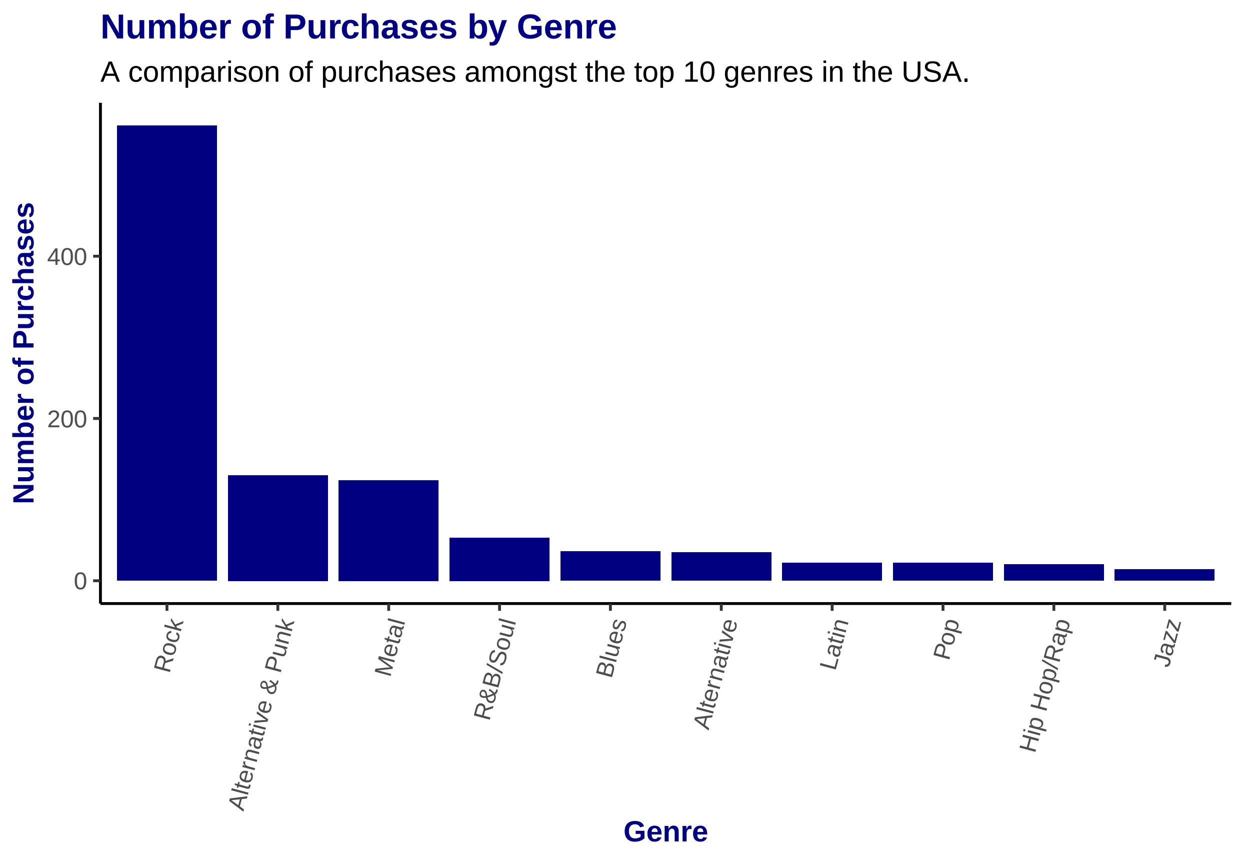
```r
g_opts <- c('Hip Hop/Rap','Alternative & Punk','Pop','Blues')
g_opts[g_opts %in% top_10_g$name[1:8]]
```

```
## [1] "Alternative & Punk" "Pop"                "Blues"
```

All options except hip-hop are present in the top-8 genre list. Therefore, if only able to choose 3 out of the 4 options, Punk, Pop, and Blues would be the more popular choices.

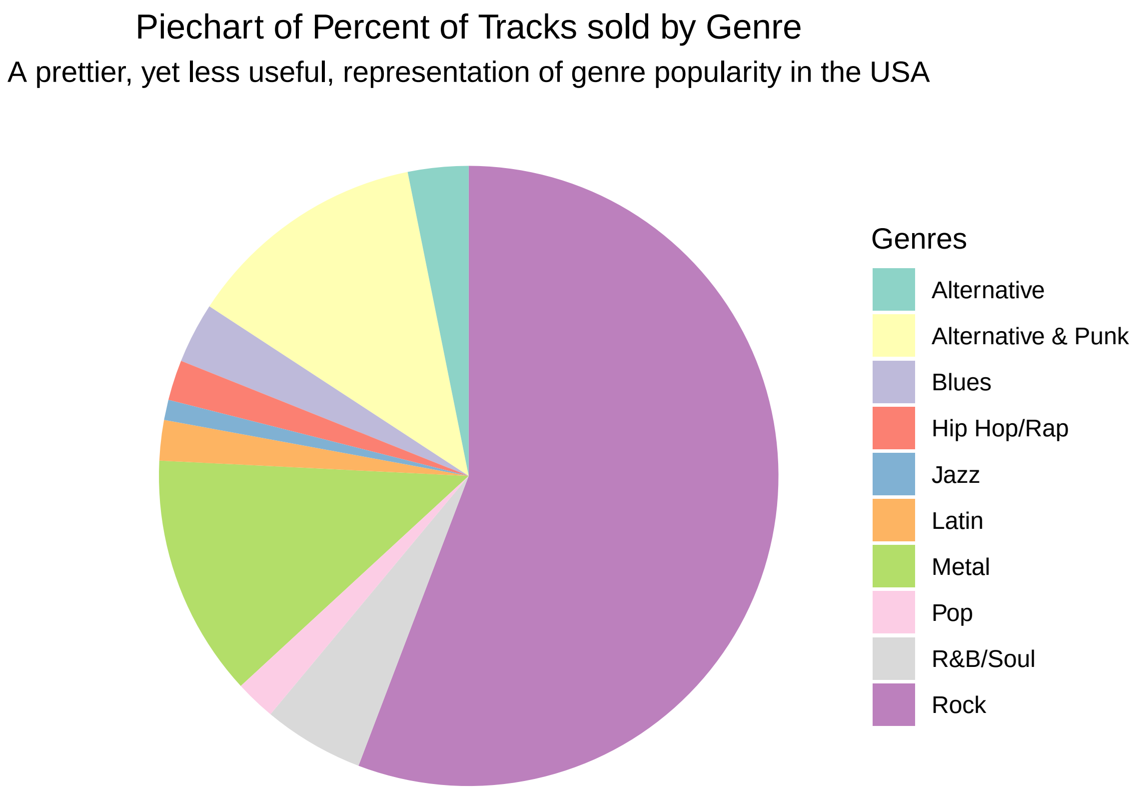
Graph findings:

```
top_10_g %>% ggplot(aes(x=reorder(name, -num_purchases),
                        y=num_purchases)) +
  geom_bar(stat='identity', fill = 'navy') +
  labs(title = 'Number of Purchases by Genre',
       subtitle = 'A comparison of purchases amongst the top 10 genres in the USA.',
       x = 'Genre',
       y = 'Number of Purchases') +
  theme_classic() +
  theme(axis.text.x = element_text(angle=75,
                                   hjust=1),
        plot.title = element_text(face = 'bold',
                                  color = 'navy'),
        axis.title = element_text(face = 'bold',
                                  color = 'navy'))
```




```
top_10_g %>% ggplot(aes(x='', y=percent, fill=name)) +
  geom_bar(stat = 'identity', width=1) +
  coord_polar('y', start = 0) +
  labs(title = 'Piechart of Percent of Tracks sold by Genre',
       subtitle = 'A prettier, yet less useful, representation of genre popularity in the USA') +
  theme_void() +
  theme(plot.title = element_text(hjust=0.5),
```

```r
       plot.subtitle = element_text(hjust = 0.5)) +
  scale_fill_brewer(palette = 'Set3',
                    name = "Genres")
```



Looking into Employee Performance

```r
query <- "SELECT
            e.first_name || ' ' || e.last_name AS employee_name,
            e.title,
            e.birthdate,
            e.hire_date,
            SUM(i.total) AS total_sold
          FROM employee AS e
          LEFT JOIN customer AS c ON e.employee_id = c.support_rep_id
          LEFT JOIN invoice AS i ON c.customer_id = i.customer_id
          GROUP BY e.employee_id
          ORDER BY 2 DESC;"
```

```r
employee_sales <- run_query(query)
employee_sales
```

```
##      employee_name                 title           birthdate           hire_date
## 1     Jane Peacock Sales Support Agent 1973-08-29 00:00:00 2017-04-01 00:00:00
## 2    Margaret Park Sales Support Agent 1947-09-19 00:00:00 2017-05-03 00:00:00
## 3    Steve Johnson Sales Support Agent 1965-03-03 00:00:00 2017-10-17 00:00:00
```

```
## 4    Nancy Edwards       Sales Manager 1958-12-08 00:00:00 2016-05-01 00:00:00
## 5      Robert King            IT Staff 1970-05-29 00:00:00 2017-01-02 00:00:00
## 6   Laura Callahan            IT Staff 1968-01-09 00:00:00 2017-03-04 00:00:00
## 7 Michael Mitchell          IT Manager 1973-07-01 00:00:00 2016-10-17 00:00:00
## 8     Andrew Adams     General Manager 1962-02-18 00:00:00 2016-08-14 00:00:00
##   total_sold
## 1    1731.51
## 2    1584.00
## 3    1393.92
## 4         NA
## 5         NA
## 6         NA
## 7         NA
## 8         NA
```

Factors contributing to sales: - Employee title: only "Sales Support Agents" have any sales - Hire date: the Sales Agent hired first sold the most and the Sales Agent hired last sold least.

Birthdate (old vs young) does not seem to impact sales.

Analyze sales from customers from each country

Want: total number of customers, total sales, avg sale per customer, and avg order value for each country.

Countries who have had more than 1 customer:

```
query <- "WITH country_sales AS
          (
          SELECT
            c.country,
            COUNT(DISTINCT(c.customer_id)) AS num_customers,
            COUNT(DISTINCT(i.invoice_id)) AS total_orders,
            SUM(i.total) AS total_sales
          FROM customer AS c
          LEFT JOIN invoice AS i ON i.customer_id = c.customer_id
          GROUP BY c.country
          HAVING num_customers > 1
          ORDER BY 2 DESC
          )
          SELECT
            country,
            num_customers,
            total_sales,
            ROUND(total_sales / num_customers, 2) AS avg_sale_per_customer,
            ROUND(total_sales / total_orders, 2) AS avg_sale_per_order
          FROM country_sales;"
```

```
sales_by_country <- run_query(query)
sales_by_country
```

```
##          country num_customers total_sales avg_sale_per_customer
## 1            USA            13     1040.49                 80.04
```

```
## 2         Canada            8      535.59                    66.95
## 3         France            5      389.07                    77.81
## 4         Brazil            5      427.68                    85.54
## 5        Germany            4      334.62                    83.66
## 6 United Kingdom            3      245.52                    81.84
## 7       Portugal            2      185.13                    92.57
## 8          India            2      183.15                    91.58
## 9 Czech Republic            2      273.24                   136.62
##   avg_sale_per_order
## 1               7.94
## 2               7.05
## 3               7.78
## 4               7.01
## 5               8.16
## 6               8.77
## 7               6.38
## 8               8.72
## 9               9.11
```

Which country/countries has/have the highest potential?

```
# A few countries with the highest average sales per customer:
highest_sales_per_customer <- sales_by_country %>%
  filter(avg_sale_per_customer >= 0.6*max(avg_sale_per_customer)) %>%
  arrange(-avg_sale_per_customer)

# A few countries with the highest average sales per order:
highest_sales_per_order <- sales_by_country %>%
  filter(avg_sale_per_order >= 0.9*max(avg_sale_per_order)) %>%
  arrange(-avg_sale_per_order)

highest_sales_per_customer %>% select(country) %>%
  filter(country %in% highest_sales_per_order$country)
```

```
##          country
## 1 Czech Republic
## 2          India
```

Both the Czech Republic and India place high in average sales per customer and average sales per order. These countries would likely be good starting points for increased marketing.

Looking at countries who have only had 1 customer

```
query_lonely <- "SELECT
                  c.country,
                  COUNT(DISTINCT(i.invoice_id)) AS total_orders,
                  SUM(i.total) AS total_sales
                FROM customer AS c
                LEFT JOIN invoice AS i ON i.customer_id = c.customer_id
                GROUP BY c.country
                HAVING COUNT(DISTINCT(c.customer_id)) = 1;"
```

```r
lonely_countries <- run_query(query_lonely)
lonely_countries
```

```
##        country total_orders total_sales
## 1    Argentina            5       39.60
## 2    Australia           10       81.18
## 3      Austria            9       69.30
## 4      Belgium            7       60.39
## 5        Chile           13       97.02
## 6      Denmark           10       37.62
## 7      Finland           11       79.20
## 8      Hungary           10       78.21
## 9      Ireland           13      114.84
## 10       Italy            9       50.49
## 11 Netherlands           10       65.34
## 12      Norway            9       72.27
## 13      Poland           10       76.23
## 14       Spain           11       98.01
## 15      Sweden           10       75.24
```

Condense the countries with 1 customer into a single observation: "others"

```r
others <- lonely_countries %>%
  mutate(country = "other",
         num_customers = nrow(lonely_countries),
         total_orders = sum(total_orders),
         total_sales = sum(total_sales),
         avg_sale_per_customer = total_sales/num_customers,
         avg_sale_per_order = total_sales/total_orders) %>%
  select(country, num_customers, total_sales, avg_sale_per_customer, avg_sale_per_order)
others <- others[1,]
others
```

```
##   country num_customers total_sales avg_sale_per_customer avg_sale_per_order
## 1   other            15     1094.94                72.996           7.448571
```

Join "others" to countries

```r
countries <- rbind(sales_by_country, others)
countries <- countries %>% arrange(-total_sales)
countries
```

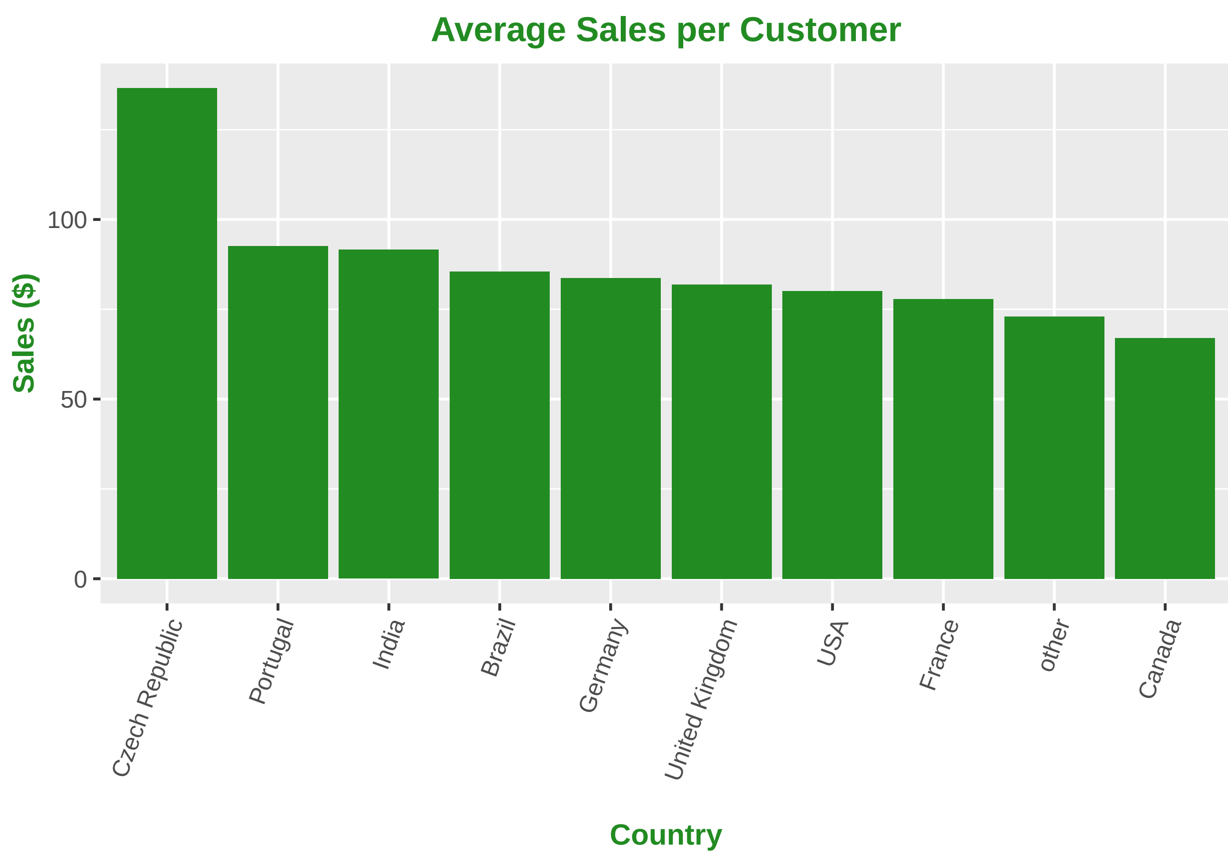
```
##           country num_customers total_sales avg_sale_per_customer
## 1           other            15     1094.94                72.996
## 2             USA            13     1040.49                80.040
## 3          Canada             8      535.59                66.950
## 4          Brazil             5      427.68                85.540
## 5          France             5      389.07                77.810
## 6         Germany             4      334.62                83.660
## 7  Czech Republic             2      273.24               136.620
## 8  United Kingdom             3      245.52                81.840
```

```
## 9        Portugal             2      185.13                 92.570
## 10          India             2      183.15                 91.580
##    avg_sale_per_order
## 1            7.448571
## 2            7.940000
## 3            7.050000
## 4            7.010000
## 5            7.780000
## 6            8.160000
## 7            9.110000
## 8            8.770000
## 9            6.380000
## 10           8.720000
```

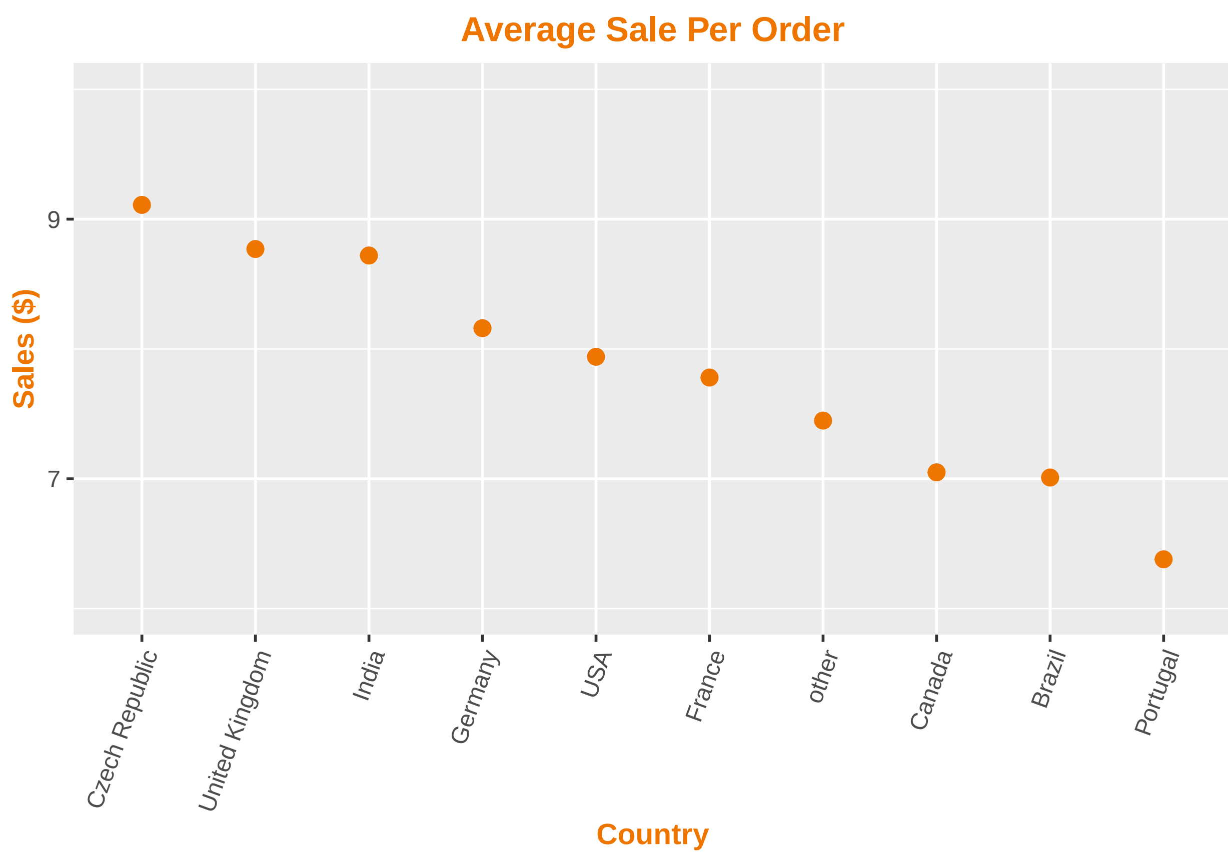
This analysis may be skewed since the aggregate "other" observation has more customers than any other country. On the whole, more data is needed for all countries for more accurate analyses.

Plot results by country

```
countries %>% ggplot(aes(x=reorder(country, -avg_sale_per_customer),
                         y=avg_sale_per_customer)) +
  geom_bar(stat='identity', fill='forestgreen') +
  theme(axis.text.x = element_text(angle=70,
                                   hjust=1)) +
  labs(x='\nCountry',
       y='Sales ($)',
       title = 'Average Sales per Customer') +
  theme(plot.title = element_text(face='bold',
                                  hjust = 0.5,
                                  color='forestgreen'),
        axis.title = element_text(face='bold',
                                  color='forestgreen'))
```




```r
countries %>% ggplot(aes(x=reorder(country, -avg_sale_per_order),
                         y=avg_sale_per_order)) +
  geom_point(color='darkorange2',
             size=2.5) +
  theme(axis.text.x = element_text(angle=70,
                                   hjust=1)) +
  scale_y_continuous(breaks=c(7, 9), limits = c(6, 10)) +
  labs(x='Country',
       y='Sales ($)',
       title='Average Sale Per Order') +
  theme(plot.title = element_text(face = 'bold',
                                  hjust = 0.5,
                                  color = 'darkorange2'),
        axis.title = element_text(face = 'bold',
                                  color = 'darkorange2'))
```



All countries have similar average sales per order (between 7 and 9 dollars).

Analyzing artists and purchase patterns

Which artist is used the most in playlists?

```
query <- "SELECT a.name, COUNT(DISTINCT(pt.track_id)) AS num_tracks
          FROM playlist_track AS pt
          LEFT JOIN track AS t ON pt.track_id = t.track_id
          LEFT JOIN album AS al ON t.album_id = al.album_id
          LEFT JOIN artist AS a ON al.artist_id = a.artist_id
          GROUP BY a.name
          ORDER BY 2 DESC
          LIMIT 10;"
```

```
top_playlist_artists <- run_query(query)
top_playlist_artists
```

```
##                name num_tracks
## 1      Iron Maiden        213
## 2               U2        135
## 3     Led Zeppelin        114
## 4        Metallica        112
## 5             Lost         92
## 6      Deep Purple         92
## 7        Pearl Jam         67
```

```
## 8    Lenny Kravitz         57
## 9  Various Artists         56
## 10      The Office         53
```

Iron Maiden appears to be the artist on the most playlists world-wide.

How many tracks have been purchased vs not purchased?

```
query <- "WITH purch_info AS
          (
          SELECT
            t.track_id,
            il.quantity
          FROM track AS t
          LEFT JOIN invoice_line AS il ON il.track_id = t.track_id
          )
          SELECT
            COUNT(quantity) AS purchased,
            COUNT(quantity IS NULL) AS not_purchased
          FROM purch_info;"
run_query(query)
```

```
##   purchased not_purchased
## 1      4757          6454
```

Purchased: 4757 Not purchased: 6454.

Do "protected" vs "non-protected" media types have an effect on popularity?

```
query <- "WITH invoice_to_media_type AS (
          SELECT
            il.invoice_line_id,
            il.track_id,
            m.name AS media_type_name
          FROM invoice_line AS il
          LEFT JOIN track AS t ON il.track_id = t.track_id
          LEFT JOIN media_type AS m ON t.media_type_id = m.media_type_id
          )
          SELECT
            media_type_name,
            COUNT(*) AS num_purchases
          FROM invoice_to_media_type
          GROUP BY 1
          ORDER BY 2 DESC;"
```

```
media_types <- run_query(query)
media_types <- media_types %>%
  mutate(type = ifelse(grepl('^Protected', media_type_name),
              "Protected",
              "Not protected")) %>%
  select(type, num_purchases) %>%
```

```r
  group_by(type) %>%
  summarise(num_purchases = sum(num_purchases))
media_types
```

```
## # A tibble: 2 x 2
##   type          num_purchases
##   <chr>                 <int>
## 1 Not protected          4315
## 2 Protected               442
```

It appears the non-protected files are more popular than the protected ones.