In [62]:

```
import sqlite3 as sql
import numpy as np
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
import plotly.express as px


```

In [63]:

```
db = 'chinook.db'

def run_query(q):
    with sql.connect(db) as conn:
        return pd.read_sql_query(q, conn)

```

In [64]:

```
q = """
WITH usa_tracks AS
    (
    SELECT t.genre_id AS genre_id, il.invoice_line_id
      FROM track as t
      INNER JOIN invoice_line as il ON il.track_id=t.track_id
      INNER JOIN invoice as i ON i.invoice_id = il.invoice_id
      WHERE i.billing_country = 'USA'
      )

SELECT g.name genre_name,
       COUNT(usa.genre_id) num_purchases,
       ROUND((CAST(COUNT(usa.genre_id) AS FLOAT)/ (SELECT COUNT(genre_id)
                                  FROM usa_tracks))*100,2) AS percentage_so
FROM usa_tracks AS usa
INNER JOIN genre as g ON g.genre_id = usa.genre_id

GROUP BY g.name
ORDER BY num_purchases DESC """

df = run_query(q)
df
```

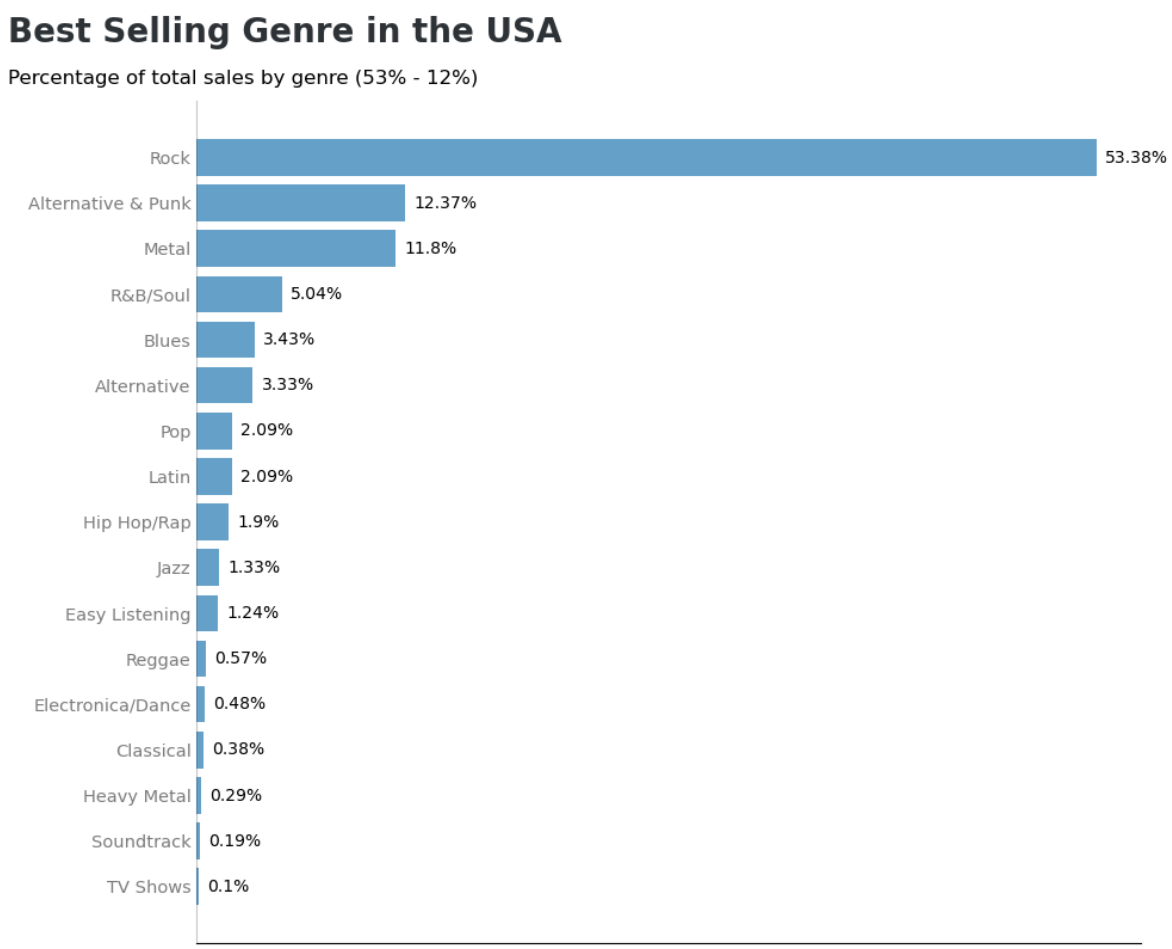
Out[64]:

	genre_name	num_purchases	percentage_sold
0	Rock	561	53.38
1	Alternative & Punk	130	12.37
2	Metal	124	11.80
3	R&B/Soul	53	5.04
4	Blues	36	3.43
5	Alternative	35	3.33
6	Pop	22	2.09
7	Latin	22	2.09
8	Hip Hop/Rap	20	1.90
9	Jazz	14	1.33
10	Easy Listening	13	1.24
11	Reggae	6	0.57
12	Electronica/Dance	5	0.48
13	Classical	4	0.38
14	Heavy Metal	3	0.29
15	Soundtrack	2	0.19
16	TV Shows	1	0.10

In [65]:

```
import matplotlib.pyplot as plt

# Define the data
genres = ['Rock', 'Alternative & Punk', 'Metal', 'R&B/Soul', 'Blues', 'Al
          'Hip Hop/Rap', 'Jazz', 'Easy Listening', 'Reggae', 'Electronica
          'Heavy Metal', 'Soundtrack', 'TV Shows']
num_purchases = [561, 130, 124, 53, 36, 35, 22, 22, 20, 14, 13, 6, 5, 4,
percentage_sold = [53.38, 12.37, 11.80, 5.04, 3.43, 3.33, 2.09, 2.09, 1.9

# Reverse the order of the data
genres.reverse()
percentage_sold.reverse()

# Create the plot
fig, ax = plt.subplots(figsize=(10, 8))

# Plot horizontal bars
bars = ax.barh(genres, percentage_sold, color='#0064AB', alpha=0.6)

# Add text labels to bars
for bar, percentage in zip(bars, percentage_sold):
    ax.text(bar.get_width() + 0.5, bar.get_y() + bar.get_height() / 2, f'
            va='center', ha='left', fontsize=10, color='black')

# Set y-axis labels
ax.set_yticklabels(genres, fontsize=10.5, color='grey')

# Remove x-axis ticks
ax.set_xticks([])

# Add title and subtitle
plt.text(-0.2, 1.07, 'Best Selling Genre in the USA', fontsize=20, fontwe
         transform=ax.transAxes)
plt.text(-0.2, 1.02, 'Percentage of total sales by genre (53% - 12%)', fo

# Remove top and right spines
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)

# Set color and transparency for left spine
ax.spines['left'].set_color('#000000')
ax.spines['left'].set_alpha(0.2)

# Remove tick marks on y-axis
ax.tick_params(axis='y', which='both', length=0)

# Adjust layout and show plot
plt.tight_layout()
plt.show()

```

```
C:\Users\dell\AppData\Local\Temp\ipykernel_10440\958326190.py:26: UserWarnin
g: FixedFormatter should only be used together with FixedLocator
  ax.set_yticklabels(genres, fontsize=10.5, color='grey')
```



Rock : Rock music is the best-selling genre in the USA, constituting approximately 53.38% of total sales. Alternative & Punk: Alternative & Punk music follows as the second most popular genre, with a sales percentage of around 12.37%. Metal: Metal music also holds a significant portion of sales, accounting for approximately 11.80% of total sales. R&B/Soul: R&B/Soul genre accounts for about 5.04% of total sales, making it a notable genre in the market. Blues, Alternative, Pop, and Latin: These genres each contribute to the sales, with percentages ranging from 3.43% to 2.09%. Other genres: Genres such as Hip Hop/Rap, Jazz, Easy Listening, Reggae, Electronica/Dance, Classical, Heavy Metal, Soundtrack, and TV Shows have smaller shares of total sales, ranging from 1.90% to 0.10%. In summary, while Rock, Alternative & Punk, and Metal dominate the market, other genres also play a significant role in the music sales landscape. Understanding these trends can help in making informed decisions regarding product offerings and marketing strategies in the USA market.

In [66]:

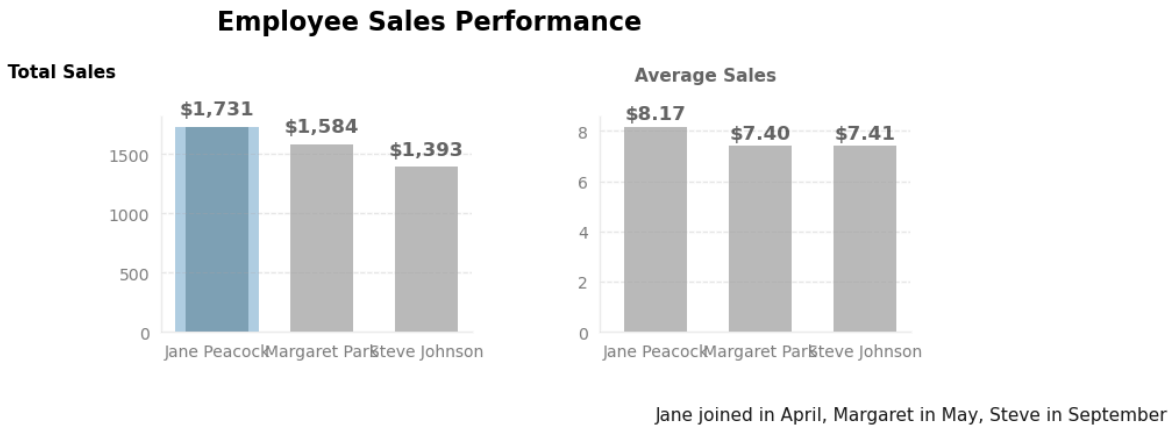
```
q = """
    WITH t1 AS (
        SELECT em.first_name || ' ' || em.last_name AS sales_rep_name,
               em.hire_date,
               COUNT(cu.customer_id) AS num_invoices,
               CAST(SUM(iv.total) AS Integer) AS total_sales
        FROM employee em
        JOIN customer cu ON em.employee_id = cu.support_rep_id
        JOIN invoice iv ON iv.customer_id = cu.customer_id
        GROUP BY 1
        ORDER BY 4 DESC
    )
    SELECT *,
           ROUND(CAST(total_sales AS Float) / num_invoices, 2) AS sales_p
    FROM t1;
"""
df = run_query(q)
df
```

Out[66]:

	sales_rep_name	hire_date	num_invoices	total_sales	sales_per_customer
0	Jane Peacock	2017-04-01 00:00:00	212	1731	8.17
1	Margaret Park	2017-05-03 00:00:00	214	1584	7.40
2	Steve Johnson	2017-10-17 00:00:00	188	1393	7.41

In [67]:

```
import matplotlib.pyplot as plt

# Assuming df is defined with required columns
sales_rep_name = df["sales_rep_name"].values
total_sales = df["total_sales"].values
sales_per_customer = df["sales_per_customer"].values

fig, axes=plt.subplots(nrows=1, ncols=2, figsize=(10, 8))

# Total Sales plot
axes[0].bar(sales_rep_name, total_sales, color='#BABABA', width=0.6)
axes[0].bar(sales_rep_name[0], total_sales[0], color ='#0064AB', alpha=0.

axes[0].text(x=-2.0, y=2150, s= 'Total Sales', size=11, fontweight='bold'

for sales, index in zip(total_sales, range(3)):
    axes[0].text(x=index, y=sales+100, s= '${:,}'.format(sales), ha='cent
              fontweight='bold', alpha=0.6, size=12)

axes[0].text(x=0, y=2550, s='Employee Sales Performance', size=16, fontwe

# Average Sales per Customer plot
axes[1].bar(sales_rep_name, sales_per_customer, color='#BABABA', width=0.
axes[1].bar(sales_rep_name[0], sales_per_customer[0], color='#0064AB', al

axes[1].text(x=-0.2, y=10, s='Average Sales', size=11, fontweight='bold',

for sales, index in zip(sales_per_customer, range(3)):
    axes[1].text(x=index, y=sales+0.3, s= '${:,.2f}'.format(sales), ha='c
                fontweight='bold', alpha=0.6, size=12)

axes[1].text(x=0, y=-3.5, s='Jane joined in April, Margaret in May, Steve
            size=11, alpha=0.9)

# Customize spines and ticks for both plots
for ax in axes:
    ax.spines['top'].set_visible(False)
    ax.spines['right'].set_visible(False)
    ax.spines['left'].set_color('#DDD')
    ax.spines['left'].set_alpha(0.5)
    ax.spines['bottom'].set_color('#DDD')
    ax.spines['bottom'].set_alpha(0.5)

    ax.tick_params(left=False, bottom=False, labelsize=10, labelcolor='gr
    ax.grid(axis='y', linestyle='--', alpha=0.3)

plt.tight_layout(rect=[0, 0.03, 1, 0.95])
plt.show()
```



Results

Jane leads in total sales with $1,731$, $followed by Margaret Park with$1,500, and Steve with $1,393. This sales discrepancy is reasonable, given that Jane and Margaret were employed approximately five months before Steve. Jane's average sales per customer of$ 8.17 supports her top position, the highest among the three employees. Steve also outperforms Margaret marginally in sales per customer, despite the disparity in their employment dates.

The Situation

Chinook seeks to analyze sales distribution across various countries to pinpoint potential growth opportunities. The company aims to identify countries with growth potential, potentially launching advertising campaigns in these regions.

Analysis

To address this inquiry, we will formulate a query that consolidates purchase data from different countries. For each country, we will aggregate the following metrics: total number of customers, total sales value, average sales per customer, and average order value. In cases where a country has only one customer, we will group it under an "Other" category.

In [68]:

```
q = """
       -- Collate the number of customers in each country
       WITH t1 AS (
                   SELECT country, COUNT(customer_id) AS num_customers
                     FROM customer
                     GROUP BY country
                   ),
       -- Collate the total sales in each country
            t2 AS (
                   SELECT cu.country,
                          ROUND(SUM(iv.total), 2) AS total_sales,
                          COUNT(iv.invoice_id) AS num_sales
                   FROM customer cu
                   JOIN invoice iv ON cu.customer_id = iv.customer_id
                   GROUP BY 1
                   ),
       -- Group countries with only one customer as 'Others'
           t3 AS (
                   SELECT CASE WHEN t1.num_customers = 1 THEN 'Others' ELS
                          SUM(t1.num_customers) AS num_customers,
                          SUM(t2.total_sales) AS total_sales,
                          SUM(t2.num_sales) AS num_sales
                   FROM t1
                   JOIN t2 ON t1.country = t2.country
                   GROUP BY 1
                   )
        -- Calculate relevant sales metrics
        SELECT countries,
               num_customers,
               total_sales,
               ROUND(total_sales / num_sales, 2) AS avg_order_value,
               ROUND(total_sales / num_customers, 2) AS sales_per_custome
          FROM (
                SELECT *,
                  CASE WHEN countries = 'Others' THEN 1 ELSE 0 END AS sor
                  FROM t3
                )
          ORDER BY sort, num_customers DESC;
        """

# Execute the query using the run_query function
df = run_query(q)
df
```

Out[68]:

	countries	num_customers	total_sales	avg_order_value	sales_per_customer
0	USA	13	1040.49	7.94	80.04
1	Canada	8	535.59	7.05	66.95
2	Brazil	5	427.68	7.01	85.54
3	France	5	389.07	7.78	77.81
4	Germany	4	334.62	8.16	83.66
5	United Kingdom	3	245.52	8.77	81.84
6	Czech Republic	2	273.24	9.11	136.62
7	India	2	183.15	8.72	91.58
8	Portugal	2	185.13	6.38	92.57
9	Others	15	1094.94	7.45	73.00

In [69]:

```
1 # Additional calculations
2 avg_cust_purchase = df.sales_per_customer.mean()
3 print(avg_cust_purchase)
4 df['pcent_customer'] = round(100*df.num_customers / df.num_customers.sum(
5 df['pcent_sales'] = round(100*df.total_sales / df.total_sales.sum(),1)
6 df['cust_purchase_diff'] = round(100 * (df.sales_per_customer - avg_cust_
7
```

```
86.96099999999998
```

In [70]:

```
1 df
```

Out[70]:

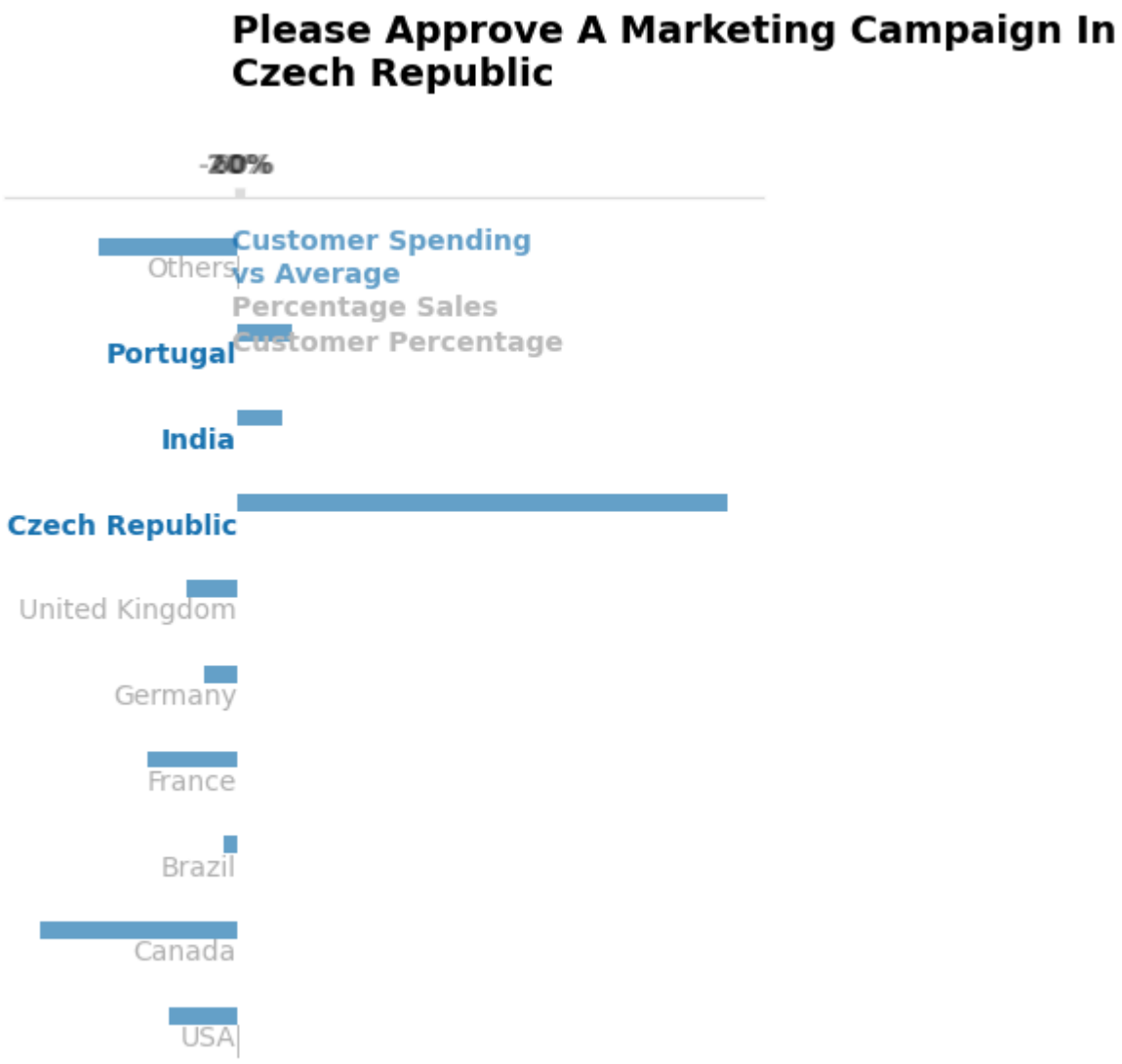
	countries	num_customers	total_sales	avg_order_value	sales_per_customer	pcent_customer
0	USA	13	1040.49	7.94	80.04	22.0
1	Canada	8	535.59	7.05	66.95	13.6
2	Brazil	5	427.68	7.01	85.54	8.5
3	France	5	389.07	7.78	77.81	8.5
4	Germany	4	334.62	8.16	83.66	6.8
5	United Kingdom	3	245.52	8.77	81.84	5.1
6	Czech Republic	2	273.24	9.11	136.62	3.4
7	India	2	183.15	8.72	91.58	3.4
8	Portugal	2	185.13	6.38	92.57	3.4
9	Others	15	1094.94	7.45	73.00	25.4

In [71]:
```
1 (136.62 - 86.96099999999998)
```

Out[71]:
```
49.65900000000002
```

In [72]:

```
#visulization
# Additional calculations
avg_cust_purchase = df.sales_per_customer.mean()
df['pcent_customer'] = round(100 * df.num_customers / df.num_customers.su
df['pcent_sales'] = round(100 * df.total_sales / df.total_sales.sum(), 1)
df['cust_purchase_diff'] = round(100 * (df.sales_per_customer - avg_cust_

# Visualization
y_labs = df.countries.values
y_axes = np.arange(df.countries.size)

fig = plt.figure(figsize=(5, 6))
plt.barh(y_axes - 0.3, df.pcent_customer, height=0.2, color='#BABABA')
plt.barh(y_axes - 0.1, df.pcent_sales, height=0.2, color='#BABABA')
plt.barh(y_axes + 0.1, df.cust_purchase_diff, height=0.2, color='#0064AB'

color_map = ['', '', '', '', '', '', 'Yes', 'Yes', 'Yes', '']

for loc, label, color in zip(y_axes, y_labs, color_map):
    if color == 'Yes':
        plt.text(x=-2, y=loc - 0.25, s=label, ha='right', color='#0064AB'
    else:
        plt.text(x=-2, y=loc - 0.25, s=label, ha='right', size=10, alpha=

plt.text(x=-60, y=8.7, s='Customer Spending\nvs Average', color='#0064AB'
plt.text(x=-60, y=8.3, s='Percentage Sales', color='#BABABA', size=10, fo
plt.text(x=-60, y=7.9, s='Customer Percentage', color='#BABABA', size=10,

plt.text(x=-60, y=11, s='Please Approve A Marketing Campaign In\nCzech Re

for ax in fig.get_axes():
    plt.sca(ax)
    sns.despine(left=True, bottom=True, top=False)
    ax.tick_params(left=False, bottom=False, color='#ddd')
    ax.xaxis.set_ticks_position('top')
    ax.spines['top'].set_color('#DDD')
    plt.yticks([])
    plt.xticks([-20, 0, 20, 40, 60], ['-20%', '0', '20%', '40%', '60%'],

plt.show()

```



Results

- The bulk of Chinook's sales originate from the US and Canada, with these two countries leading both in customer numbers and sales figures. However, customers in these regions tend to spend less per invoice compared to other countries.
- On the contrary, the Czech Republic, Portugal, and India, despite having fewer customers and lower total sales, exhibit higher average spending per invoice. In these countries, customers tend to spend more for each transaction than in other markets.
- To capitalize on this potential for increased revenue, Chinook could implement targeted marketing campaigns aimed at expanding its customer base in these three markets. By focusing efforts on customer acquisition and engagement in the Czech Republic, Portugal, and India, C
  Chinook can potentially boost sales and maximize revenue from these regions.

How Many Tracks Never Sell ?

To answer this question, we will have to distriguish between the entire inventory of tracks in the track table and the distinct instances of tracks from the invoice line table.

In [73]:
```
q = """WITH all_and_purchased AS
       (
       SELECT t.track_id AS all_tracks, il.track_id AS purchased_tracks
          FROM track AS t
          LEFT JOIN invoice_line AS il ON il.track_id = t.track_id
       )

SELECT COUNT(DISTINCT a.all_tracks) AS total_tracks,
    COUNT(DISTINCT a.purchased_tracks) AS tracks_purchased,
    COUNT(DISTINCT a.all_tracks) - COUNT(DISTINCT a.purchased_tracks) AS
    ROUND(COUNT(DISTINCT a.purchased_tracks) / CAST(COUNT(DISTINCT a.all_

FROM
    all_and_purchased AS a;"""

# Execute the query using the run_query function
purchased = run_query(q)
purchased
```

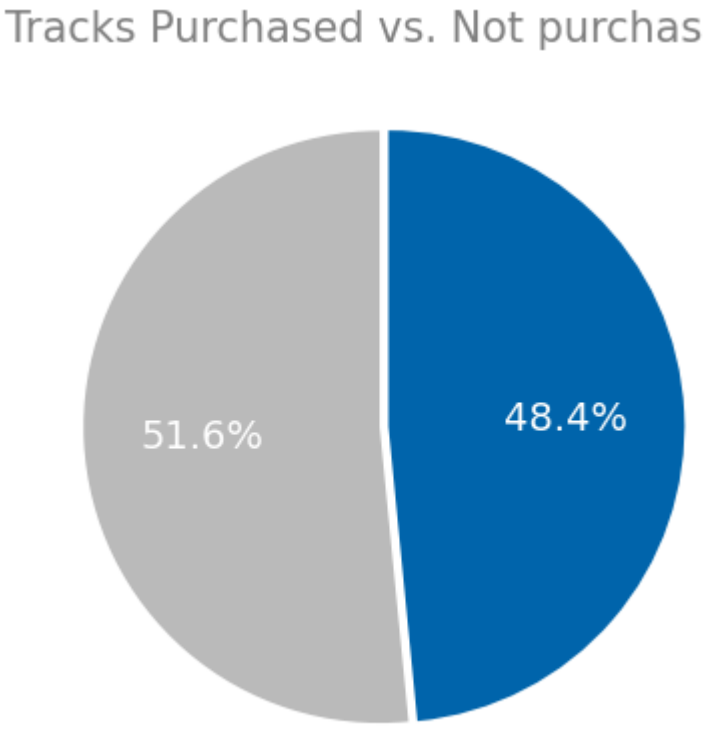
Out[73]:

	total_tracks	tracks_purchased	not_purchased	purchase_percentage
0	3503	1806	1697	1.0

In [74]:
```
purchased = purchased.T
purchased = purchased.iloc[[1,2],0]
purchased_list = purchased.tolist()
```

In [75]:

```
plt.pie(purchased_list,
        explode=(0, 0.03),
        startangle=90,  # Corrected typo here
        autopct='%1.1f%%',
        textprops={'fontsize':14, 'color':'white'},
        colors=('#BABABA','#0064AB'))
plt.title('Tracks Purchased vs. Not purchased', fontsize=15, color="grey"
fig = plt.gcf()
```



Observations

- Suprisingly, almost half of the track inventory at chinook has not sold. lets take a look at the bottom performers and see of we can learn more:

In [76]:

```
q = """
SELECT ar.name AS artist_name,
       g.name AS genre,
       COUNT(il.track_id) AS units_sold
FROM track AS t
LEFT JOIN invoice_line AS il ON il.track_id = t.track_id
INNER JOIN album AS al ON al.album_id = t.album_id
INNER JOIN artist AS ar ON ar.artist_id = al.artist_id
INNER JOIN genre AS g ON g.genre_id = t.genre_id
GROUP BY artist_name
HAVING units_sold = 0
ORDER BY units_sold;"""

run_query(q)
```

Out[76]:

	artist_name	genre	units_sold
0	Aaron Copland & London Symphony Orchestra	Classical	0
1	Academy of St. Martin in the Fields Chamber En...	Classical	0
2	Academy of St. Martin in the Fields, John Birc...	Classical	0
3	Academy of St. Martin in the Fields, Sir Nevil...	Classical	0
4	Adrian Leaper & Doreen de Feis	Classical	0
...	...	...	...
69	The Office	TV Shows	0
70	The Tea Party	Alternative & Punk	0
71	Ton Koopman	Classical	0
72	Toquinho & Vinícius	Bossa Nova	0
73	Various Artists	Pop	0

74 rows × 3 columns

Observations ¶

74 artists have not sold any units, with most of these tracks belonging to the classical music genre.

Half of the Company's inventory remains unsold, potentially tying up working capital without generating returns.

Depending on the payment arrangment with record labels, there are two scenarios to consider:

1.If Chinook incurs a fixed hosting fee for these tracks, it would be prudent to prioritize genres with higher popularity and consider discontinuing contracts with less successful artists.

2.If Chinook pays the record label based on a percentage of sales, there is minimal risk in

Regardless of the scenario, Chinook should explore ways to promote these low-selling artists. suggestions could be integrated into the purchasing process or displayed on the website's cart page to increases exposure.

Albums vs Individual Tracks

The chinook store allows customers to buy music in two ways: either as a complete album or as individual tracks. However, customers cannot buy a fall album and then add individual tracks to the same purchase unless they select each track manually. When customers purchase albums, they are charged the same price as if they bought each track separately.

Management is contemplating a new purchasing approach to cut costs. instead of buying every track from an album, they are considering purchasing only the most popular tracks from each album from record companies.

In [80]:

```
q = """
WITH invoice_data AS
(
    SELECT invoice_id, MIN(track_id) AS track_id
    FROM invoice_line
    GROUP BY invoice_id
),
Album_purchased AS
(
    SELECT invoice_id,
        CASE
            WHEN
            (
                SELECT t2.track_id
                FROM track t1
                JOIN track t2 ON t1.album_id = t2.album_id
                WHERE t1.track_id = invd.track_id
                EXCEPT
                SELECT il.track_id
                FROM invoice_line il
                WHERE il.invoice_id = invd.invoice_id
            ) IS NULL
            AND
            (
                SELECT il.track_id
                FROM invoice_line il
                WHERE il.invoice_id = invd.invoice_id
                EXCEPT
                SELECT t2.track_id
                FROM track t1
                JOIN track t2 ON t1.album_id = t2.album_id
                WHERE t1.track_id = invd.track_id
            ) IS NULL
            THEN 'Yes'
            ELSE 'No'
        END AS purchased_Album
    FROM invoice_data invd
)

SELECT purchased_Album,
       COUNT(invoice_id) AS no_of_invoices,
       CAST(COUNT(invoice_id) AS FLOAT) * 100 /
       (SELECT COUNT(*) FROM Album_purchased) AS percent
FROM Album_purchased
GROUP BY 1

"""

run_query(q)
```

Out[80]:

	purchased_Album	no_of_invoices	percent
0	No	500	81.433225
1	Yes	114	18.566775

Resluts

Most purchases (81%) from the store are individual tracks. However, in about 19% cases, customers buy entire albums. Chinook Should be careful with purchasing only the most popular tracks since it risks losing revenue from customers who purchase entire albums.

Conclusion and Recommendations

Throughout this project, we have provided insights and recommendations to help a fictional company enhance its profitability. By addressing various business scenarios, we've offered guidance on different aspects of the company's operations.

*** Genre Selection for New Albums:**

Our analysis suggests that Chinook should prioritize genres with high popularity in the USA. We recommend selecting albums from Hip-Hop,punk, and Pop genres, as they show promissing sales potential. Additionally, Keeping an eye on Rock songs, which costitute they majority of sales in the USA, could further boost revenue.

*** Employee Performance :**

Among the sales representatives, Jane Peacock stands out as the top performer. While Steve Johnson appears to have lower total sales, it's important to note that he joined the team later than the others. Thus, he may require additional support and training to reach this full potential.

*** Sales Analysis bu Country:**

While the USA and Canada have the largest customer base, customers in these countries tend to spend less per transaction. On the other hand, the Czech Republic, India and Portugal show higher average spending per customer. Launching targeted marketing campaigns in these regions could attract more customers and increase sales.

*** Track Purchasing Strategy:**

Chinook's plan to focus solely on popular tracks may seem appealing, but it risks alienating customers who perfer to purchase entire albums. Conducting customer surveys and gathering feedback before implementing any changes would provide valuable insights into customer prefrences.

In [ ]:
```
1
```