Analyzing Car Expenses Based on Receipts

This notebook goes through some interesting stats about how much it has cost us to own a car. Since purchasing the car, I have been saving receipts from all car-related expenses. There are periods where other people have borrowed the car and fueled it up without saving the receipt, so consider this a lower bound on the expenses.

Table of Contents:

In [1]:
```
# Packages
from pathlib import Path
import os
import pandas as pd
import numpy as np
from datetime import datetime

import plotly.io as pio
import plotly.express as px
import plotly.graph_objects as go
from plotly.subplots import make_subplots

#pio.renderers.default = "svg"   ## for showing plotly graphs when uploading notebook to
```

In [2]:
```
## Setting CWD
cwd = os.getcwd()
root = Path(cwd).parents[0]
datadir = str(Path(root)) + str('/data')
outputdir = str(Path(root)) + str('/figures')
print("Root: " + str(root))
```

```
Root: /Users/madeaa/Dropbox/IFN/Programming/Python_projects/analyzing_car_expenses
```

In [3]:
```
infile = datadir + "/raw_data.xlsx"

df = pd.read_excel(infile, sheet_name='Blad1')

df_bränsle = pd.read_excel(infile, sheet_name='bränsle')

df_bränsle_snittpriser = pd.read_excel(infile, sheet_name='bränsle_snittpriser')
```

In [4]:
```
df = df.append(df_bränsle[['Datum','Utgift']], ignore_index=True)      ## Appending rele
df['Kategori'].replace(np.nan, 'Bränsle', inplace=True)                ## Fixing Category
```

```
/var/folders/_q/sr1nk96s4mb_g3qysrj841j40000gq/T/ipykernel_45844/1139251548.py:1: FutureWarning: The frame.append method is deprecated and will be removed from pandas in a future version. Use pandas.concat instead.
  df = df.append(df_bränsle[['Datum','Utgift']], ignore_index=True)      ## Appending relevant fuel data
```

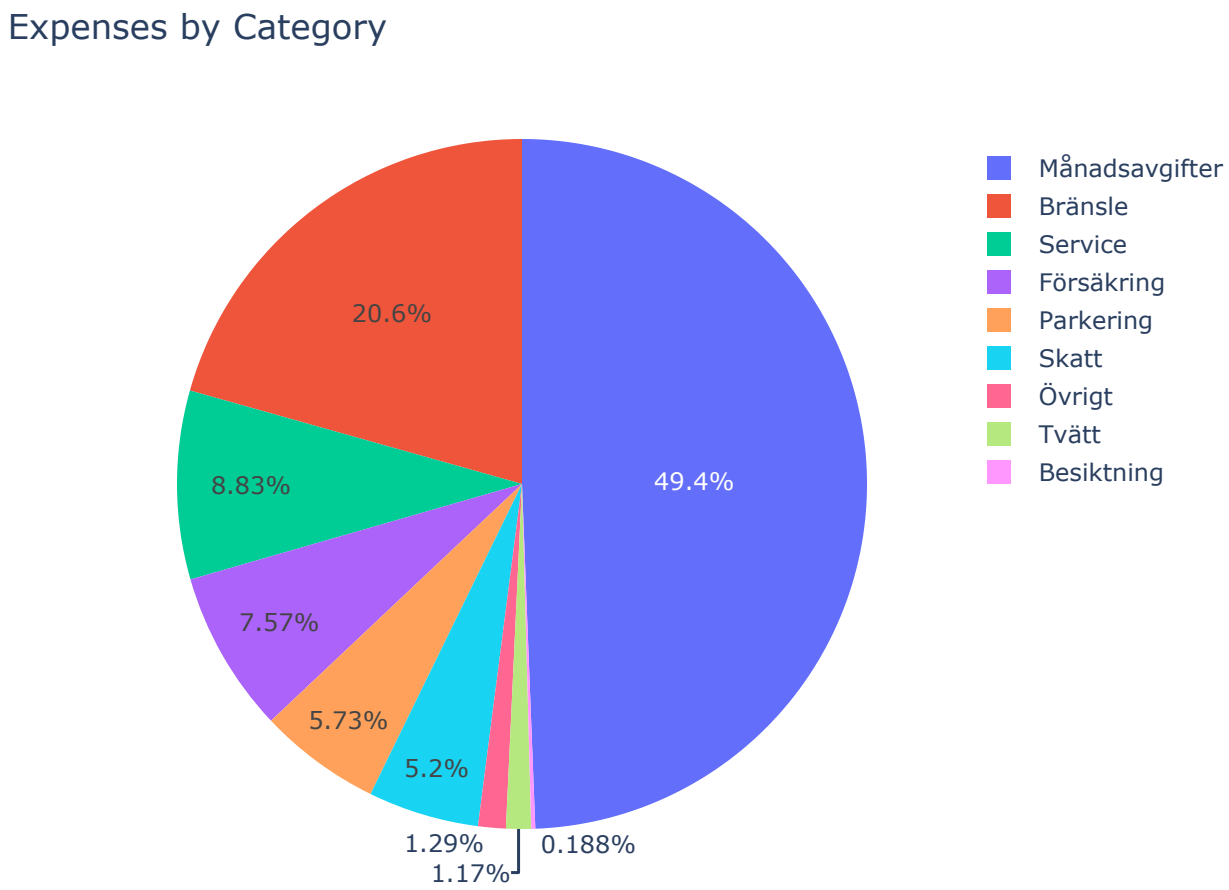
Where Did the Money Go?

Let's break up the total costs of the car by the category.

In [5]:
```
print(f"The data contains information on the following categories: {list(df.Kategori.uni
```

```
The data contains information on the following categories: ['Parkering', 'Tvätt', 'Övrigt', 'Skatt', 'Service', 'Försäkring', 'Månadsavgifter', 'Besiktning', 'Bränsle']
```

In [6]:
```
df_kategori = df.groupby('Kategori').sum().reset_index()

fig = px.pie(df_kategori, values='Utgift', names='Kategori', title='Expenses by Category
fig.show()

outfile = outputdir + "/1_expenses_by_category.png"
fig.write_image(outfile)
```

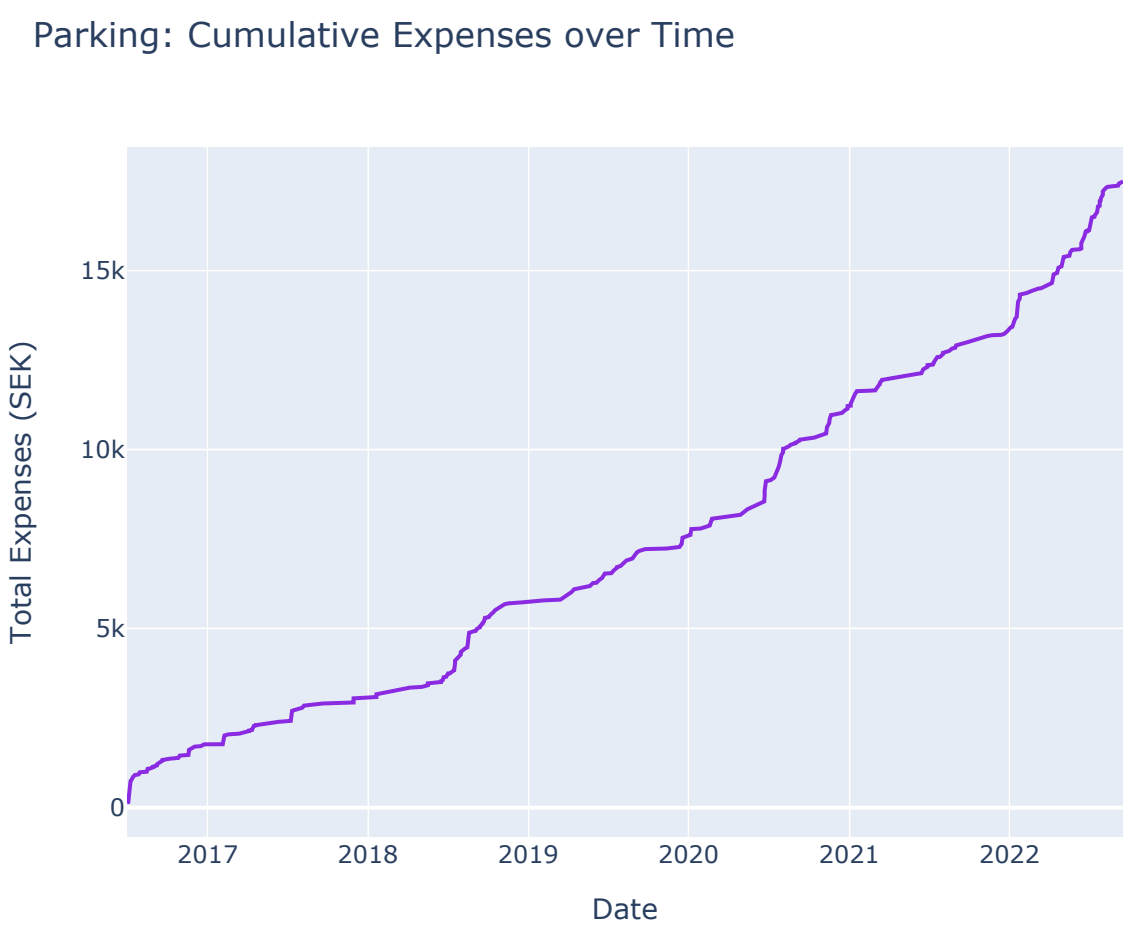


In the figure above, we see that the greatest expenses was purchasing the car. This is followed by paying for fuel and insurance.

Parking: What's Space Worth Anyways?

How much have we paid for parking, in total? Let's take a look!

In [7]:
```
df_parkering = df[df['Kategori']=="Parkering"].reset_index()
df_parkering['Utgift_cumulative'] = df_parkering.loc[::-1, 'Utgift'].cumsum()[::-1]

fig = px.line(df_parkering,
              x='Datum',
              y='Utgift_cumulative',
              title = f'Parking: Cumulative Expenses over Time',
              labels={"Utgift_cumulative": "Total Expenses (SEK)","Datum": "Date"},
              color_discrete_sequence=["blueviolet"])

fig.show()

outfile = outputdir + "/2_parking_cumulative_expenses_over_time.png"
fig.write_image(outfile)
```



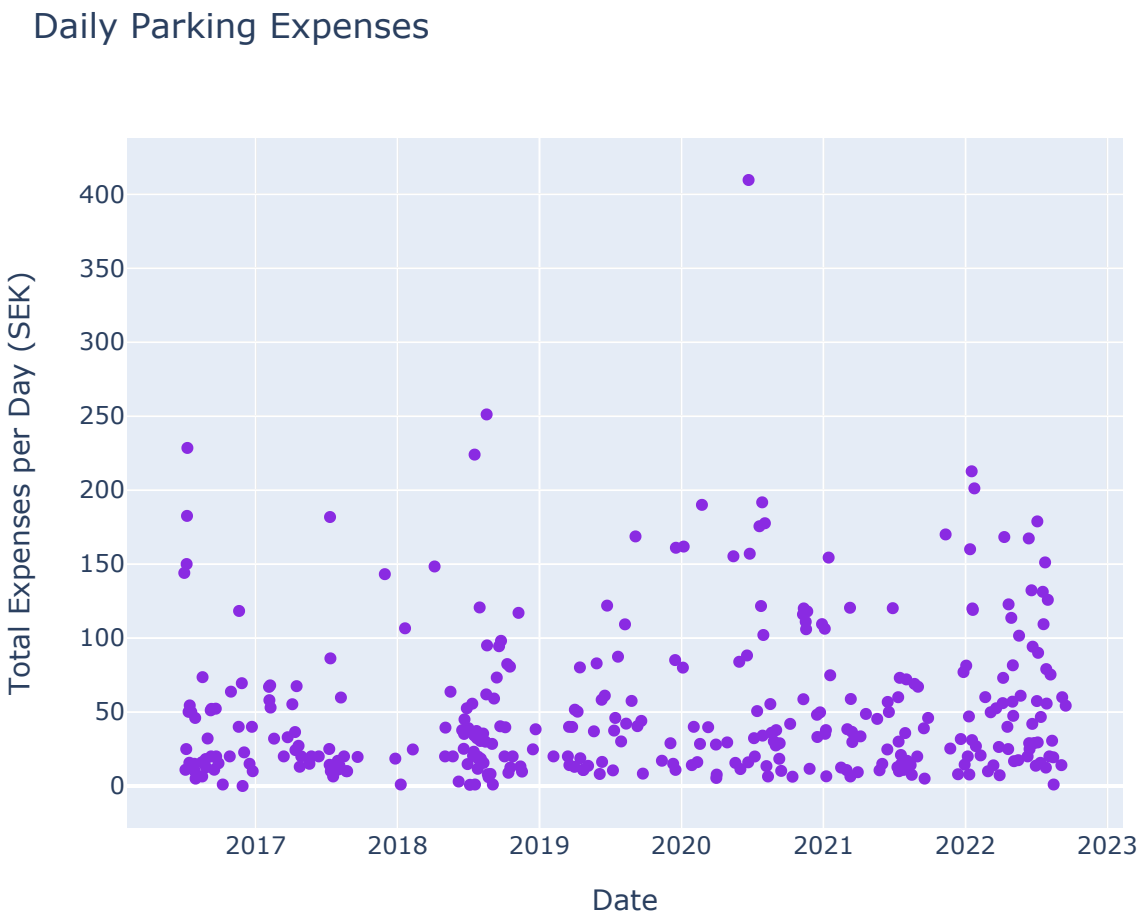
In [8]:
```
print(f"Between the {df_parkering.Datum.min().strftime('%d %B %Y')} and {df_parkering.Da
```

```
Between the 01 July 2016 and 16 September 2022, 433 parking occasions have cost us a tot
al of 17481.130000000023 SEK.
```

Seasonality in Parking

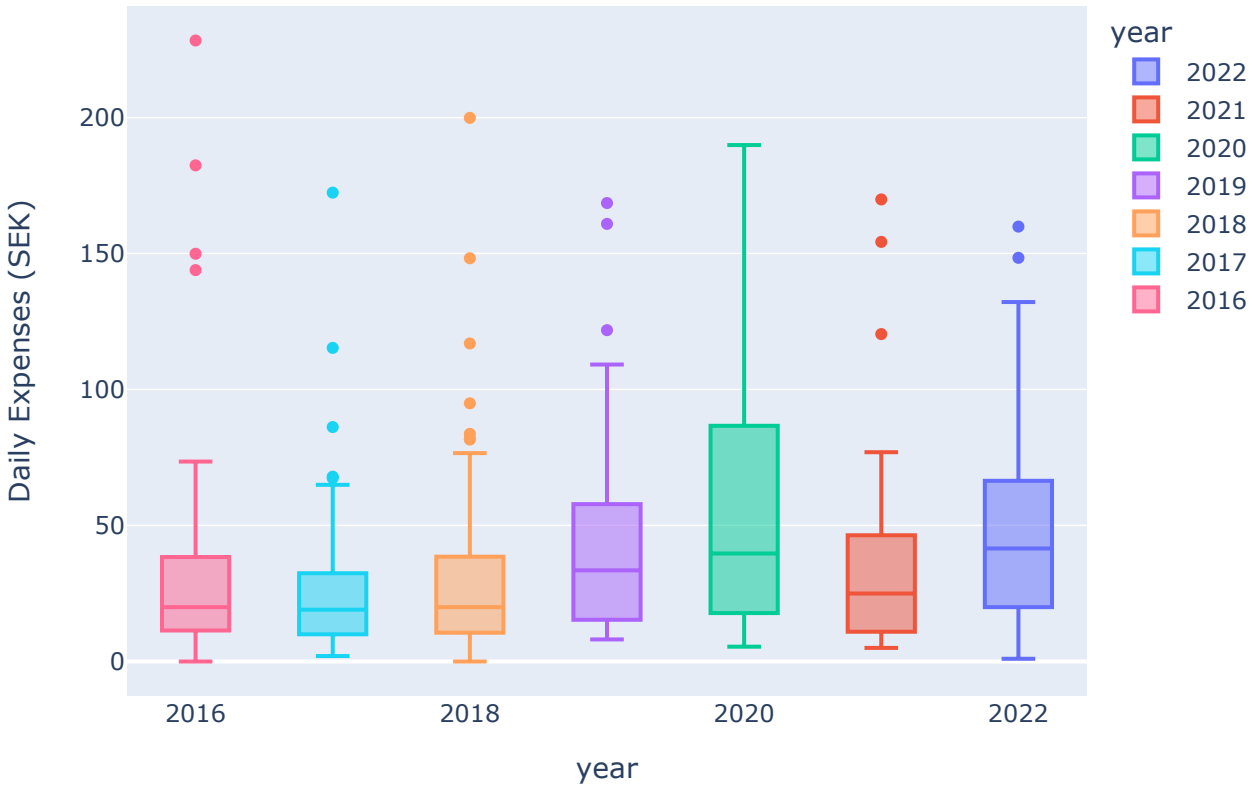
What does the daily parking price tag look like?

In [9]:
```
fig = px.scatter(df_parkering[['Datum','Utgift']].groupby('Datum').sum().reset_index(),
                 x='Datum',
                 y='Utgift',
                 title = 'Daily Parking Expenses',
                 labels={"Utgift": "Total Expenses per Day (SEK)","Datum": "Date"},
                 color_discrete_sequence=["blueviolet"]
                )

fig.show()

outfile = outputdir + "/3_daily_parking_expenses.png"
fig.write_image(outfile)
```



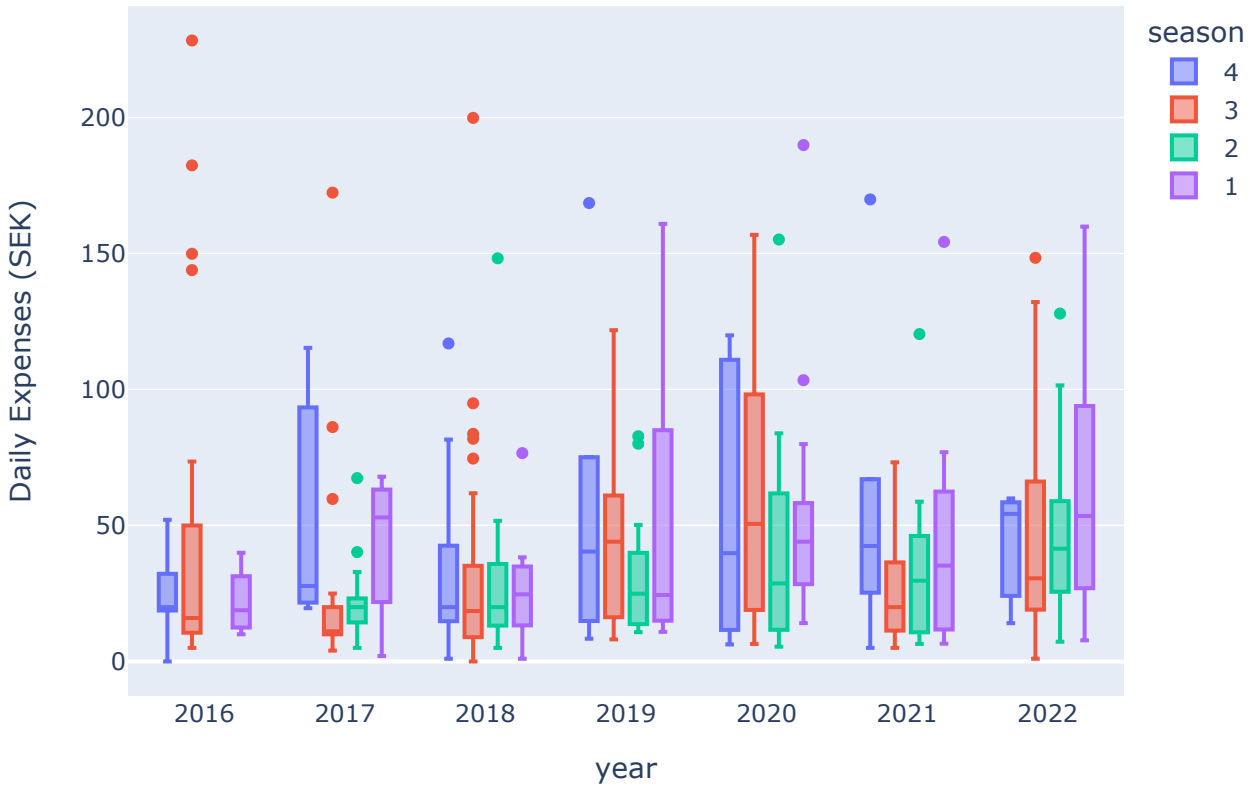
How has my average spending on parking changed over time?

In [10]:
```
df_parkering['year'], df_parkering['month'] = df_parkering['Datum'].dt.year, df_parkerin

fig = px.box(df_parkering,
             x='year',
             y='Utgift',
             color='year',
             title='Parking Expenses by Year')

fig.update_yaxes(title_text="Daily Expenses (SEK)")

fig.show()

outfile = outputdir + "/4_parking_expenses_by_year.png"
fig.write_image(outfile)
```



In [11]:

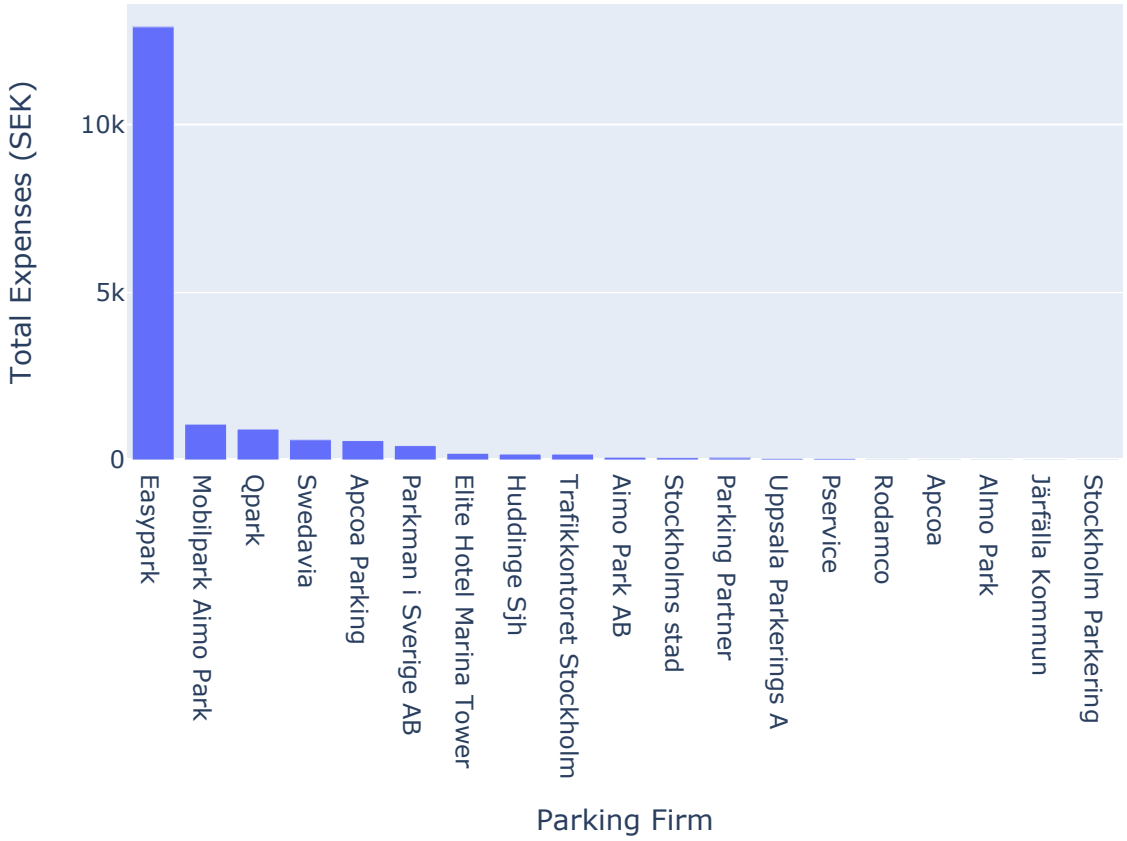
```
df_parkering['season'] = df_parkering.Datum.dt.month%12 // 3 + 1

fig = px.box(df_parkering,
             x='year',
             y='Utgift',
             color='season',
             title='Parking Expenses by Season and Year')

fig.update_yaxes(title_text="Daily Expenses (SEK)")

fig.show()

outfile = outputdir + "/5_parking_expenses_by_season_and_year.png"
fig.write_image(outfile)
```



Who Do We Pay for Parking?

Let's break down the parking data by parking firm.

In [12]:
```
fig = px.bar(df_parkering[['Typ','Utgift']].groupby('Typ').sum(),
             title='Parking Expenses by Parking Firm')

fig.update_yaxes(title_text="Total Expenses (SEK)")
fig.update_xaxes(title_text="Parking Firm", categoryorder="total descending")
fig.update_layout(showlegend=False)

fig.show()

outfile = outputdir + "/7_parking_expenses_by_parking_firm.png"
fig.write_image(outfile)
```



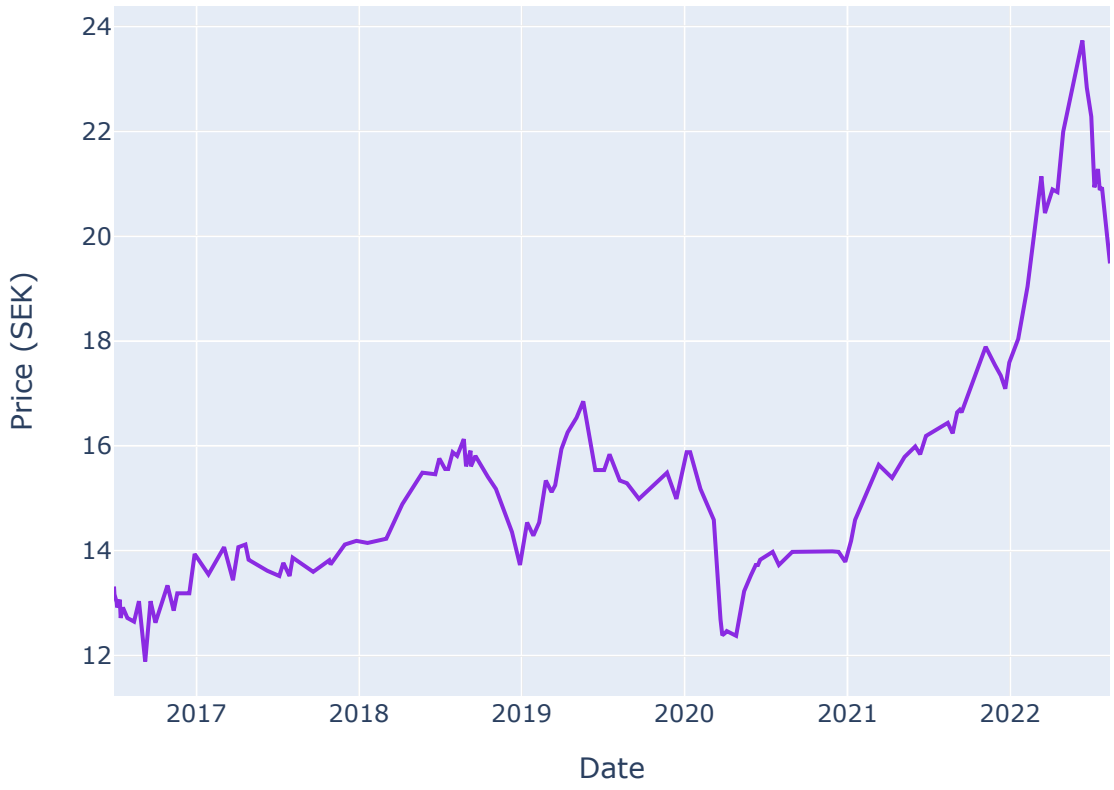
The winner is: Easypark!

Gas Prices: Fuelling my Pain!

In second place in terms of expenses, after paying for the actual car, was paying for fuel. Let's explore the fuel receipts!

Gas Prices over Time

In [13]:
```
fig = px.line(df_bränsle,
              x='Datum',
              y='Pris',
              title = f'E95: Price over Time',
              labels={"Pris": "Price (SEK)","Datum": "Date"},
              color_discrete_sequence=["blueviolet"])

fig.show()

outfile = outputdir + "/8_E95_price_over_time.png"
fig.write_image(outfile)
```



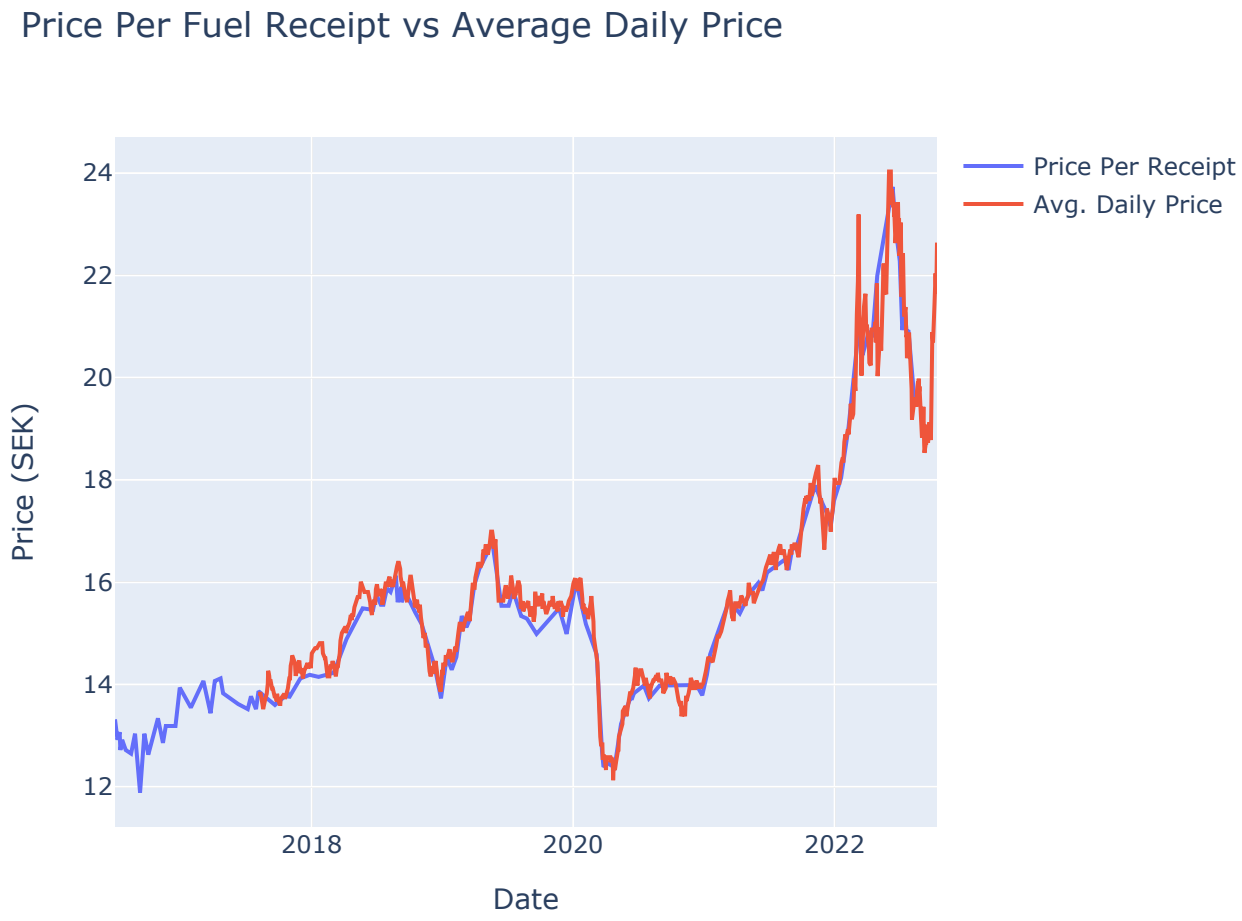
In [14]:
```
print(f"The figure above shows that fuel prices have been as low as {df_bränsle.Pris.min
```

```
The figure above shows that fuel prices have been as low as 11.89 SEK and reached a maxi
mum of 23.73 SEK.
The latest price is 19.48 SEK.
```

How good am I at fueling from price-worthy gas stations? Let's compare the prices from my fuel receipts with average daily prices of fuel in Sweden!

In [15]:
```
fig = make_subplots()

# Add traces
fig.add_trace(
    go.Scatter(x=df_bränsle['Datum'], y=df_bränsle['Pris'], name="Price Per Receipt")
)

fig.add_trace(
    go.Scatter(x=df_bränsle_snittpriser['Datum'], y=df_bränsle_snittpriser['E95'], name=
)

# Add figure title
fig.update_layout(
    title_text="Price Per Fuel Receipt vs Average Daily Price"
)

# Set x-axis title
fig.update_xaxes(title_text="Date")

# Set y-axes titles
fig.update_yaxes(title_text="Price (SEK)")
```

```
fig.show()

outfile = outputdir + "/9_price_per_fuel_receipt_vs_average_daily_price.png"
fig.write_image(outfile)
```



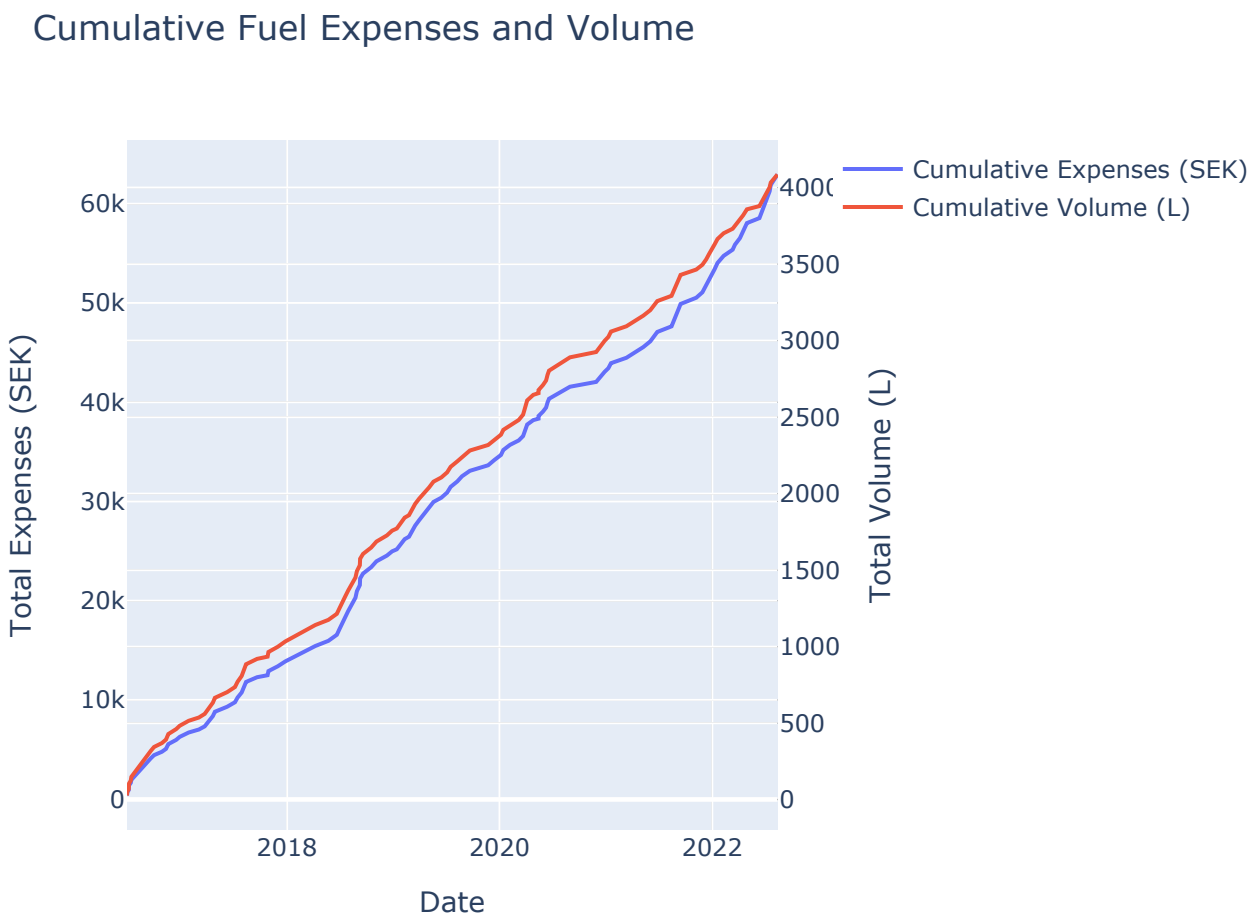
The price at which I fuel the car seems to be consistently equal to or less than the average price of fuel. Not too bad!

What's the Final Price Tag for Fuel Consumption?

In [16]:

```
df_bränsle['Utgift_cumulative'] = df_bränsle.loc[::-1, 'Utgift'].cumsum()[::-1]
df_bränsle['L_cumulative'] = df_bränsle.loc[::-1, 'L'].cumsum()[::-1]

fig = make_subplots(specs=[[{"secondary_y": True}]])

# Add traces
fig.add_trace(
    go.Scatter(x=df_bränsle['Datum'], y=df_bränsle['Utgift_cumulative'], name="Cumulativ
    secondary_y=False,
)

fig.add_trace(
    go.Scatter(x=df_bränsle['Datum'], y=df_bränsle['L_cumulative'], name="Cumulative Vol
    secondary_y=True,
)

# Add figure title
fig.update_layout(
    title_text="Cumulative Fuel Expenses and Volume"
```

```
)

# Set x-axis title
fig.update_xaxes(title_text="Date")

# Set y-axes titles
fig.update_yaxes(title_text="Total Expenses (SEK)", secondary_y=False)
fig.update_yaxes(title_text="Total Volume (L)", secondary_y=True)

fig.show()

outfile = outputdir + "/10_cumulative_fuel_expenses_and_volume.png"
fig.write_image(outfile)
```



In [17]:
```
print(f"The final price tag is{df_bränsle.Utgift_cumulative.head(1).mean(): .2f} SEK, fo
```

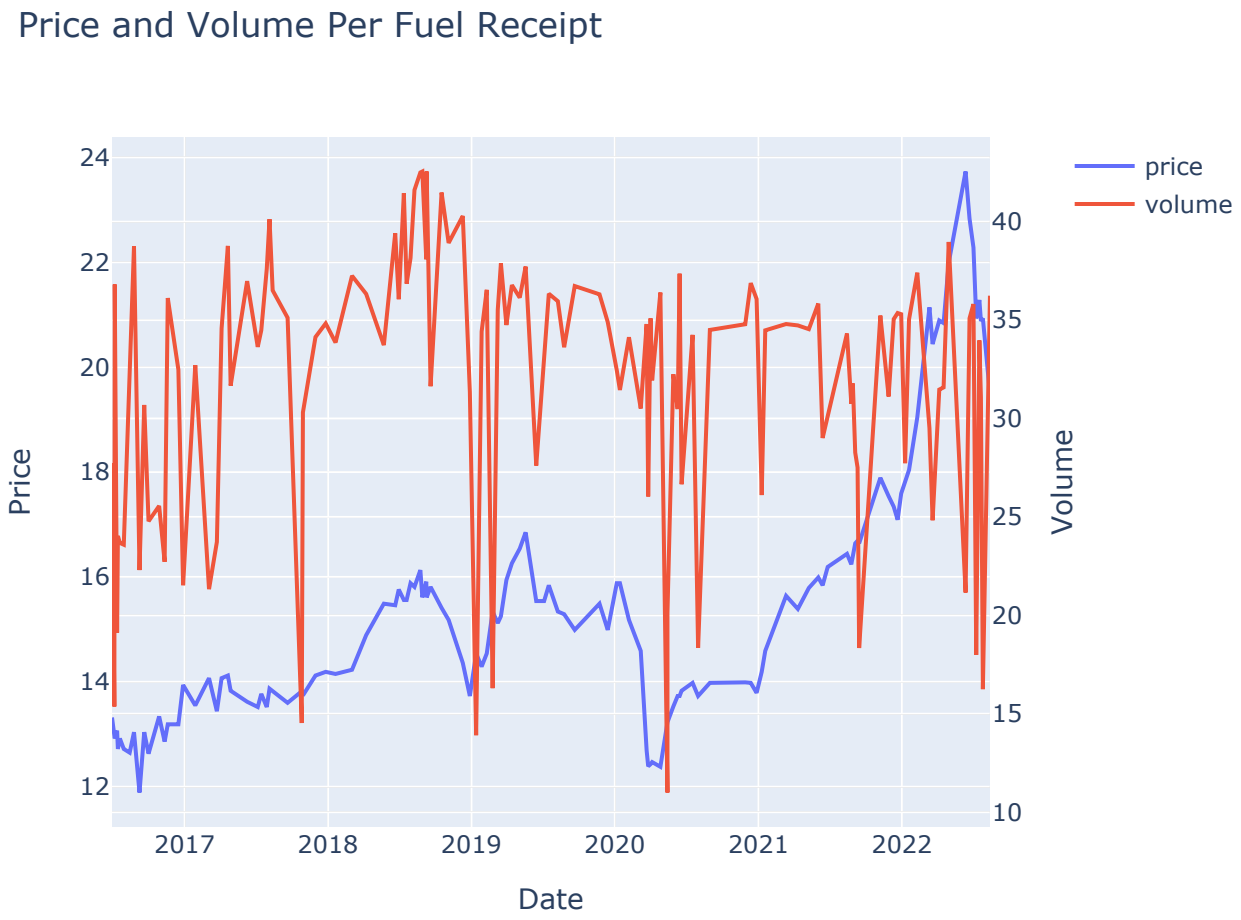
```
The final price tag is 62946.81 SEK, for a total of 4085.39 litres of fuel.
That is equivalent to around 102.13 full tanks.
```

Buy Low, Wait High?

Do I tend to buy more fuel when the prices are relatively low?

In [18]:
```
fig = make_subplots(specs=[[{"secondary_y": True}]])

# Add traces
fig.add_trace(
    go.Scatter(x=df_bränsle['Datum'], y=df_bränsle['Pris'], name="price"),
    secondary_y=False,
)
```

```
fig.add_trace(
    go.Scatter(x=df_bränsle['Datum'], y=df_bränsle['L'], name="volume"),
    secondary_y=True,
)

# Add figure title
fig.update_layout(
    title_text="Price and Volume Per Fuel Receipt"
)

# Set x-axis title
fig.update_xaxes(title_text="Date")

# Set y-axes titles
fig.update_yaxes(title_text="Price", secondary_y=False)
fig.update_yaxes(title_text="Volume", secondary_y=True)

fig.show()

outfile = outputdir + "/11_price_and_volume_per_fuel_receipt.png"
fig.write_image(outfile)
```



A quick visual inspection of price and volume data shows that there may be such a negative co-movement until July 2018. Since then, which co-incides with me moving from Uppsala to Stockholm, there isn't a clear pattern. If anything, we fuel up more when prices are higher!

In [19]:
```
print(f"In summary, between the {df_bränsle.Datum.min().strftime('%d of %B %Y')} and {df
print(f"- Recorded entries: {df_bränsle.Datum.count()}\n")
print(f"- Min. Price: {df_bränsle.Pris.min()} SEK")
```

```
print(f"- Avg. Price: {df_bränsle.Pris.mean():.2f} SEK")
print(f"- Max. Price: {df_bränsle.Pris.max()} SEK\n")
print(f"- Min. Fuel Litres: {df_bränsle.L.min()} SEK")
print(f"- Avg. Fuel Litres: {df_bränsle.L.mean():.2f} SEK")
print(f"- Max. Fuel Litres: {df_bränsle.L.max()} SEK\n")

print(f"- Total Expenses: {df_bränsle.Utgift_cumulative.head(1).mean():.2f} SEK")
print(f"- Total Litres: {df_bränsle.L_cumulative.head(1).mean():.2f} SEK\n")

print(f"- Nr Gas Station Chains: {df_bränsle.Företag.nunique()}")
```

```
In summary, between the 30 of June 2016 and 13 of August 2022:

- Recorded entries: 128

- Min. Price: 11.89 SEK
- Avg. Price: 15.37 SEK
- Max. Price: 23.73 SEK

- Min. Fuel Litres: 11.0 SEK
- Avg. Fuel Litres: 31.92 SEK
- Max. Fuel Litres: 42.55 SEK

- Total Expenses: 62946.81 SEK
- Total Litres: 4085.39 SEK

- Nr Gas Station Chains: 5
```

To The Cheapest Gas Station, Please!

Which gas station chains do I usually fuel up at, and are they the cheapest?

In [20]:
```
df_bränsle_grouped = pd.pivot_table(df_bränsle, values='Pris', index=['Datum'],columns='
print(f"From the table below, we see that I usually fuel up at Ingo (a total of {df_brän
print(f"On average, St1 has lower gas prices than the rest ({df_bränsle_grouped['St1'].d
print(f"However, I used to fuel at St1 when I lived in Uppsala, where fuel is generally

df_bränsle_grouped.describe()
```

```
From the table below, we see that I usually fuel up at Ingo (a total of 84 times).
On average, St1 has lower gas prices than the rest (14.27 SEK).
However, I used to fuel at St1 when I lived in Uppsala, where fuel is generally cheaper.
```

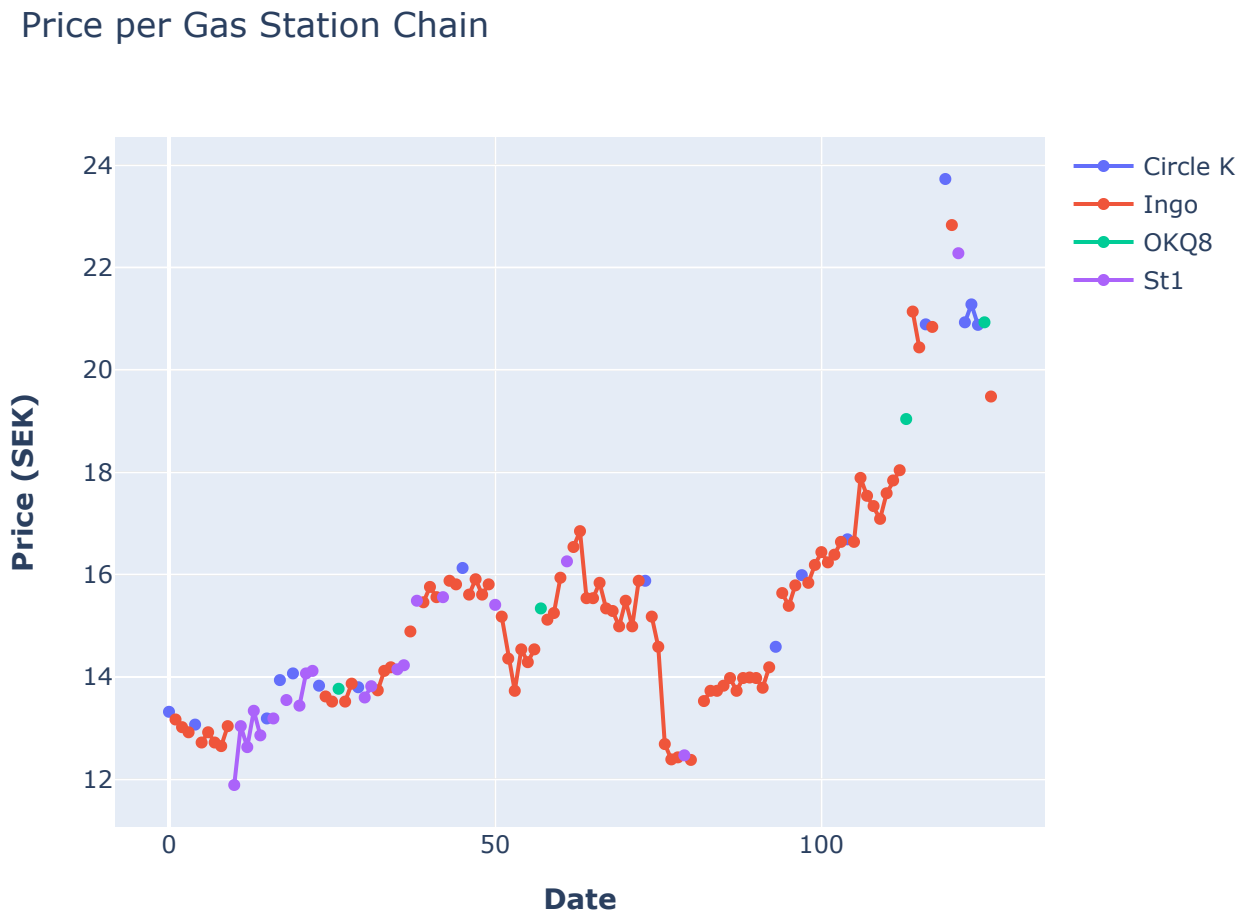
Out[20]:

Företag	Circle K	Ingo	OKQ8	Shell	St1
count	17.000000	84.000000	4.000000	2.000000	20.000000
mean	16.600588	15.269762	17.270000	24.225000	14.270000
std	3.510761	2.033701	3.291575	3.160767	2.196119
min	13.070000	12.380000	13.770000	21.990000	11.890000
25%	13.830000	13.777500	14.947500	23.107500	13.152500
50%	15.880000	15.215000	17.190000	24.225000	13.710000
75%	20.880000	15.917500	19.512500	25.342500	14.525000
max	23.730000	22.830000	20.930000	26.460000	22.280000

Let's compare fuel prices from the gas station chains, based on my saved receipts.

In [21]:
```
fig = go.Figure()
for col in df_bränsle_grouped[['Circle K','Ingo','OKQ8','St1']].columns:
    fig.add_trace(go.Scatter(x=df_bränsle_grouped.index, y=df_bränsle_grouped[col].value
```

```
                                name = col,
                                mode = 'markers+lines',
                                line=dict(shape='linear')
                                )
                    )

fig.update_layout(
    title_text="Price per Gas Station Chain"
)

fig.update_xaxes(title_text="<b>Date</b>")
fig.update_yaxes(title_text="<b>Price (SEK)</b>")

fig.show()

outfile = outputdir + "/12_price_per_gas_station_chain.png"
fig.write_image(outfile)
```



There are many problems with trying to compare fuel prices from the gas chains based on my receipts. First, since I rarely fuel up from multiple gas Stations on the same day, we cannot compare the fuel prices. Second of all, it is not random where I fuel up the car. My favorite chain is Ingo, and I often drive past a couple of other gas stations (which sell fuel at a higher price) on my way there.

End