In [1]:

```
#Khalid Kadri
#Department of computer science
# City, University of London
# London United Kingdom
# Khalid.kadri@city.ac.uk

#Title: A Comparative Analysis of Lewis Hamilton and Sebastian Vettel's Performance in the 2018 Formula 1 Season

#In this analysis, we are interested in understanding how the performance of Lewis Hamilton and Sebastian Vettel in the 2018 Formula 1 season compares. Specifically, we want to d
```

In [2]:

```
#The data for this analysis was obtained from a user on kaggle and includes detailed information on the races, podium finishes, and points earned by each driver during the 2018 s
```

In [3]:

```
import numpy as np
from scipy import stats
from tqdm import tqdm
import pandas as pd
import datetime as dt
import matplotlib.pyplot as plt

#imported Libraries
circuits_df = pd.read_csv('circuits.csv')
constructor_results_df = pd.read_csv('constructor_results.csv')
constructor_standings_df = pd.read_csv('constructor_standings.csv')
constructors_df = pd.read_csv('constructors.csv')
driver_standings_df = pd.read_csv('driver_standings.csv')
drivers_df = pd.read_csv('drivers.csv')
lap_times_df = pd.read_csv('lap_times.csv')
pit_stops_df = pd.read_csv('pit_stops.csv')
qualifying_df = pd.read_csv('qualifying.csv')
races_df = pd.read_csv('races.csv')
results_df = pd.read_csv('results.csv')
seasons_df = pd.read_csv('seasons.csv')
sprint_results_df = pd.read_csv('sprint_results.csv')
status_df = pd.read_csv('status.csv')

# To compare lap times, we  calculate the average lap time for each driver and perform a t-test to determine whether there is a statistically significant difference between the tw
# for pit stops, we could calculate the total number of pit stops for each driver and compare the results using a t-test.
# we also compared lap 1 gains or losses, by calculating the total number of lap 1 gains or losses for each driver and compare the results using a t-test.

# Calculate the means and standard deviations of the two samples
#mean1 = np.mean(sample1)
#mean2 = np.mean(sample2)
#std1 = np.std(sample1)
#std2 = np.std(sample2)

# Calculate the t-value and degrees of freedom
#t, p = stats.ttest_ind(sample1, sample2, equal_var=True)
#df = len(sample1) + len(sample2) - 2

# Print the t-value and p-value
#print(f't-value: {t:.3f}')
#print(f'p-value: {p:.3f}')


laptime_df = pd.merge(lap_times_df,drivers_df[['driverId','code','driverRef']],how='left', on='driverId')
laptime_df =pd.merge(laptime_df,races_df[['raceId','name','date','year']],how= 'left', on='raceId')

laptime_df['time']=pd.to_timedelta(laptime_df['milliseconds'], unit='ms')
```

```
laptime_df['seconds']=laptime_df['milliseconds']/1000
laptime_df
```

Out[3]:

	raceId	driverId	lap	position	time	milliseconds	code	driverRef	name	date	year	seconds
0	841	20	1	1	0 days 00:01:38.109000	98109	VET	vettel	Australian Grand Prix	2011-03-27	2011	98.109
1	841	20	2	1	0 days 00:01:33.006000	93006	VET	vettel	Australian Grand Prix	2011-03-27	2011	93.006
2	841	20	3	1	0 days 00:01:32.713000	92713	VET	vettel	Australian Grand Prix	2011-03-27	2011	92.713
3	841	20	4	1	0 days 00:01:32.803000	92803	VET	vettel	Australian Grand Prix	2011-03-27	2011	92.803
4	841	20	5	1	0 days 00:01:32.342000	92342	VET	vettel	Australian Grand Prix	2011-03-27	2011	92.342
...	...	...	...	...	...	...	...	...	...	...	...	...
528780	1086	842	65	13	0 days 00:01:25.989000	85989	GAS	gasly	Hungarian Grand Prix	2022-07-31	2022	85.989
528781	1086	842	66	12	0 days 00:01:25.187000	85187	GAS	gasly	Hungarian Grand Prix	2022-07-31	2022	85.187
528782	1086	842	67	12	0 days 00:01:47.355000	107355	GAS	gasly	Hungarian Grand Prix	2022-07-31	2022	107.355
528783	1086	842	68	12	0 days 00:01:37.587000	97587	GAS	gasly	Hungarian Grand Prix	2022-07-31	2022	97.587
528784	1086	842	69	12	0 days 00:01:29.072000	89072	GAS	gasly	Hungarian Grand Prix	2022-07-31	2022	89.072

528785 rows × 12 columns

In [4]:

```
#lap analysis for HAM VS VET in 2018
laps_df = laptime_df[(laptime_df['year']==2018)&((laptime_df['code']=='VET')|(laptime_df['code']=='HAM'))].copy()
laps_df.rename(columns={'position':'lap position'},inplace=True)
laps_df = laps_df.merge(results_df[['raceId','driverId','position']],how='left',on=['raceId','driverId'])
laps_df = laps_df.merge(pit_stops_df[['raceId','driverId','lap','stop']],how='left',on=['raceId','driverId','lap'])
laps_df['stop'].fillna(0,inplace=True)
laps_df['stop']=laps_df['stop'].astype(int)
laps_df['stop'][laps_df['stop']==0] = ''
laps_df
```

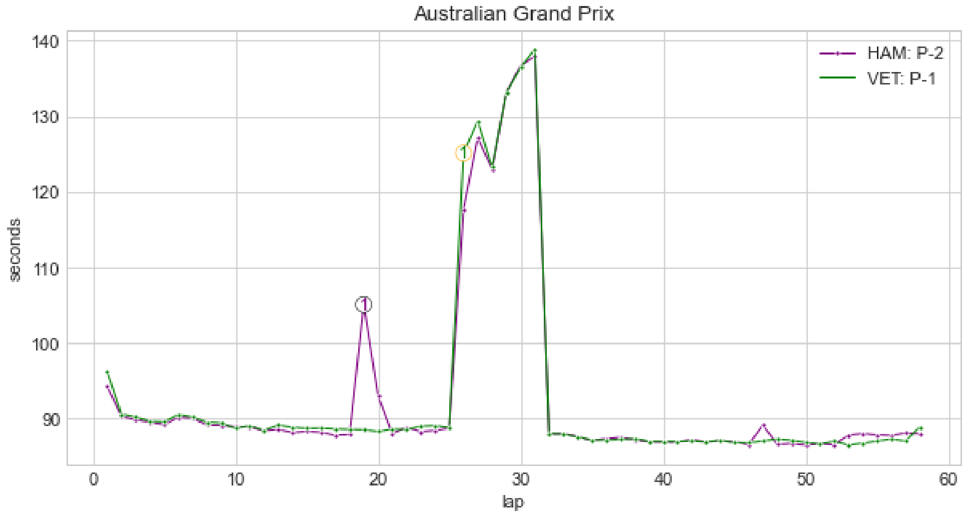
```
C:\Users\kmkad\AppData\Local\Temp\ipykernel_43300\3746640263.py:8: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  laps_df['stop'][laps_df['stop']==0] = ''
```

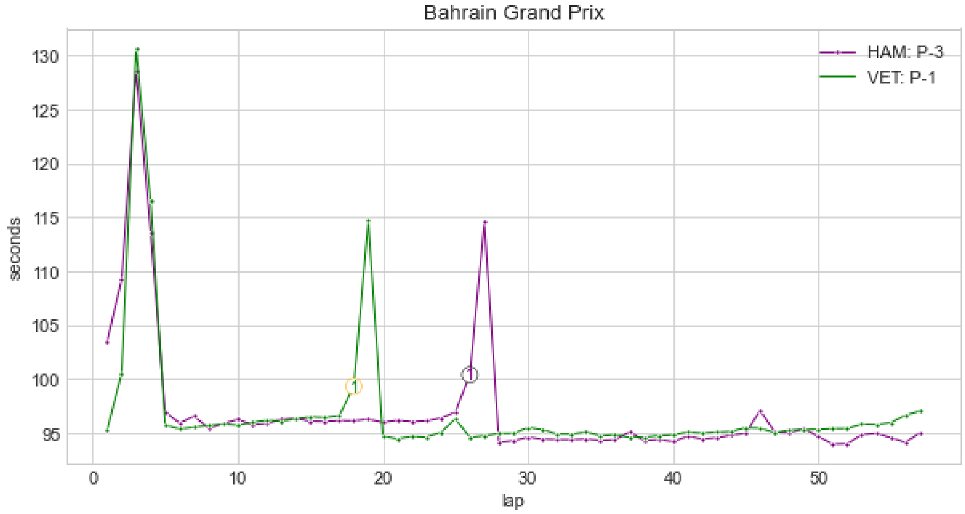
```
Out[4]:
      raceId  driverId  lap  position                     time  milliseconds  code  driverRef                   name        date  year  seconds  position  stop
   0     989        20   58         1  0 days 00:01:28.863000         88863   VET     vettel   Australian Grand Prix  2018-03-25  2018   88.863         1
   1     989        20   57         1  0 days 00:01:26.990000         86990   VET     vettel   Australian Grand Prix  2018-03-25  2018   86.990         1
   2     989        20   56         1  0 days 00:01:27.236000         87236   VET     vettel   Australian Grand Prix  2018-03-25  2018   87.236         1
   3     989        20   55         1  0 days 00:01:26.983000         86983   VET     vettel   Australian Grand Prix  2018-03-25  2018   86.983         1
   4     989        20   54         1  0 days 00:01:26.711000         86711   VET     vettel   Australian Grand Prix  2018-03-25  2018   86.711         1
 ...     ...       ...  ...       ...                      ...           ...   ...        ...                     ...         ...   ...      ...       ...   ...
2498    1009        20   51         2  0 days 00:01:41.450000        101450   VET     vettel     Abu Dhabi Grand Prix  2018-11-25  2018  101.450         2
2499    1009        20   52         2  0 days 00:01:42.233000        102233   VET     vettel     Abu Dhabi Grand Prix  2018-11-25  2018  102.233         2
2500    1009        20   53         2  0 days 00:01:41.488000        101488   VET     vettel     Abu Dhabi Grand Prix  2018-11-25  2018  101.488         2
2501    1009        20   54         2  0 days 00:01:40.867000        100867   VET     vettel     Abu Dhabi Grand Prix  2018-11-25  2018  100.867         2
2502    1009        20   55         2  0 days 00:01:41.296000        101296   VET     vettel     Abu Dhabi Grand Prix  2018-11-25  2018  101.296         2

2503 rows × 14 columns
```

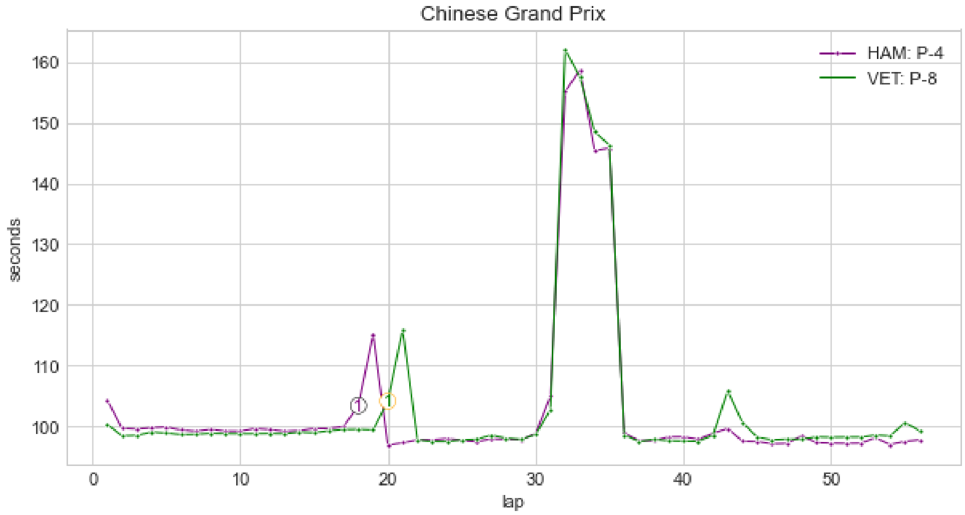
```
In [5]:
import seaborn as sns
plt.style.use('seaborn-whitegrid')
plt.rcParams['figure.figsize']=10,5
plt.rcParams['font.family'] = 'Arial'
plt.rcParams['font.size'] = 11
plt.rcParams['lines.linewidth'] = 1
plt.rcParams['axes.labelsize']=11
plt.rcParams['xtick.labelsize']=11
plt.rcParams['ytick.labelsize']=11
plt.rcParams['legend.fontsize']=11

plt.figure(figsize=(10, 100))
for i, f in tqdm(enumerate(laps_df['name'].unique())):
    try:
        HAM_pos = list(set(laps_df[(laps_df['name']==f)&(laps_df['code']=='HAM')]['position']))[0]
        if HAM_pos == r'\N':
            HAM_pos = 'DNF'
    except:
        HAM_pos = 'DNS'
    try:
        VET_pos = list(set(laps_df[(laps_df['name']==f)&(laps_df['code']=='VET')]['position']))[0]
        if VET_pos == r'\N':
            VET_pos = 'DNF'
    except:
        VET_pos = 'DNS'
    fig,ax = plt.subplots(1,1)
    plt.title(f)
    sns.lineplot(data=laps_df[(laps_df['name']==f)&(laps_df['code']=='HAM')],
                 x='lap',
                 y='seconds',
                 hue='code',
                 palette=['Purple'],
                 ax=ax,
                 marker='.',
                 # marker_size=3
                 )
    HAM_stops = laps_df[(laps_df['name']==f)&(laps_df['code']=='HAM')]
    for j,label in enumerate(HAM_stops['stop']):
        plt.annotate(label, (HAM_stops['lap'].iloc[j],
                             HAM_stops['seconds'].iloc[j]
                             ),
                     color = 'purple',
                     bbox=dict(boxstyle="circle,pad=0", fc="white", ec="black", lw=0.5),
                     ha="center", va="center",
                     )

    sns.lineplot(data=laps_df[(laps_df['name']==f)&(laps_df['code']=='VET')],
                 x='lap',
                 y='seconds',
                 hue='code',
                 ax=ax,
                 palette = ['green'],
                 marker='.',
                 )

    VET_stops = laps_df[(laps_df['name']==f)&(laps_df['code']=='VET')]
    for j,label in enumerate(VET_stops['stop']):
        plt.annotate(label, (VET_stops['lap'].iloc[j],
                             VET_stops['seconds'].iloc[j]
                             ),
                     color='green',
                     bbox=dict(boxstyle="circle,pad=0", fc="white", ec="orange", lw=0.5),
                     ha="center", va="center",
                     )

    if ax.get_legend_handles_labels()[1][0]=='HAM':
        plt.legend(['HAM: P-'+str(HAM_pos), 'VET: P-'+str(VET_pos)])
        leg = ax.get_legend()
        leg.legendHandles[0].set_color('purple')
        leg.legendHandles[1].set_color('green')
    elif ax.get_legend_handles_labels()[1][0]=='VET':
        plt.legend(['VET: P-'+str(VET_pos), 'HAM: P-'+str(HAM_pos)])
        leg = ax.get_legend()
        leg.legendHandles[0].set_color('purple')
        leg.legendHandles[1].set_color('green')


plt.tight_layout()
plt.show()
```

```
19it [00:01, 10.17it/s]C:\Users\kmkad\AppData\Local\Temp\ipykernel_43300\997538898.py:26: RuntimeWarning: More than 20 figures have been opened. Figures created through the pyplo
t interface (`matplotlib.pyplot.figure`) are retained until explicitly closed and may consume too much memory. (To control this warning, see the rcParam `figure.max_open_warning
`).
  fig,ax = plt.subplots(1,1)
21it [00:02, 10.22it/s]
```

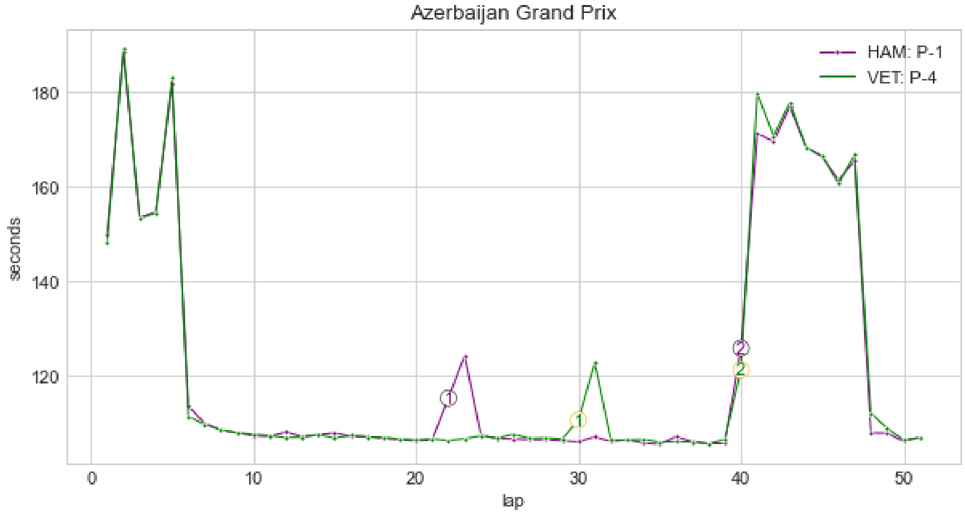
```
<Figure size 720x7200 with 0 Axes>
```


Australian Grand Prix
HAM: P-2
VET: P-1
seconds
140
130
120
110
100
90
0
10
20
30
40
50
60
lap
1
1

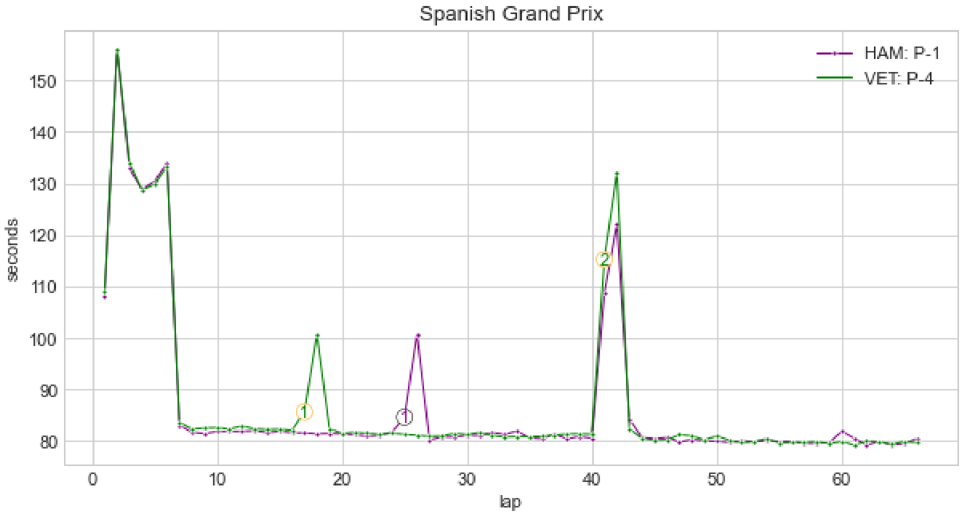

Bahrain Grand Prix
HAM: P-3
VET: P-1
seconds
130
125
120
115
110
105
100
95
0
10
20
30
40
50
lap
1
1



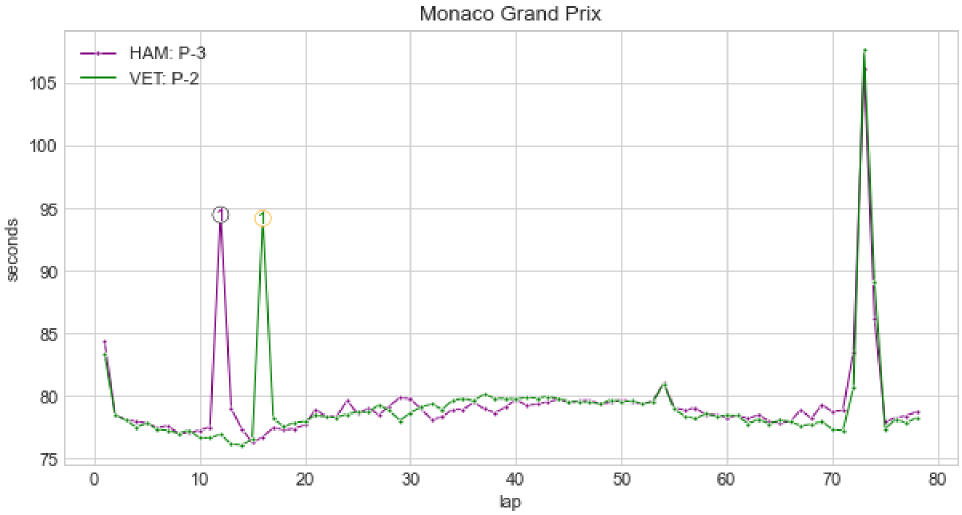
Chinese Grand Prix



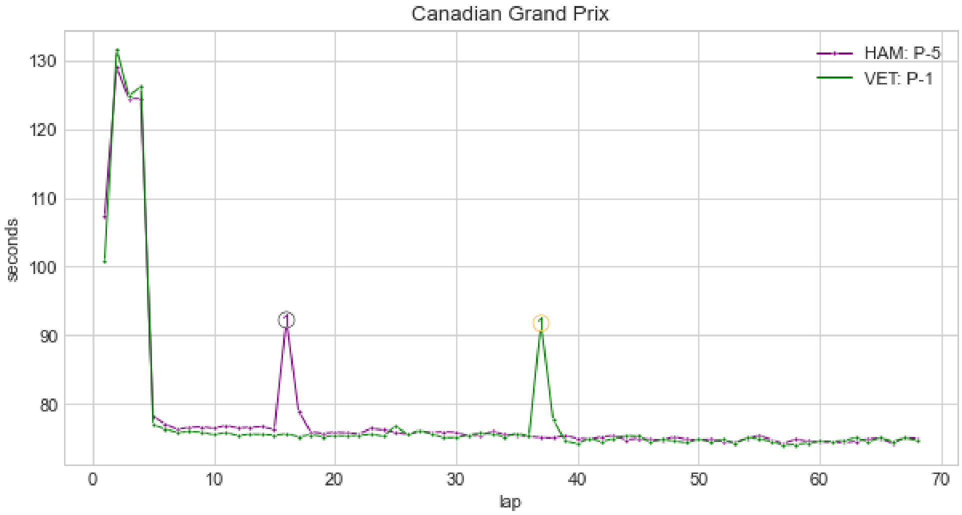
Azerbaijan Grand Prix

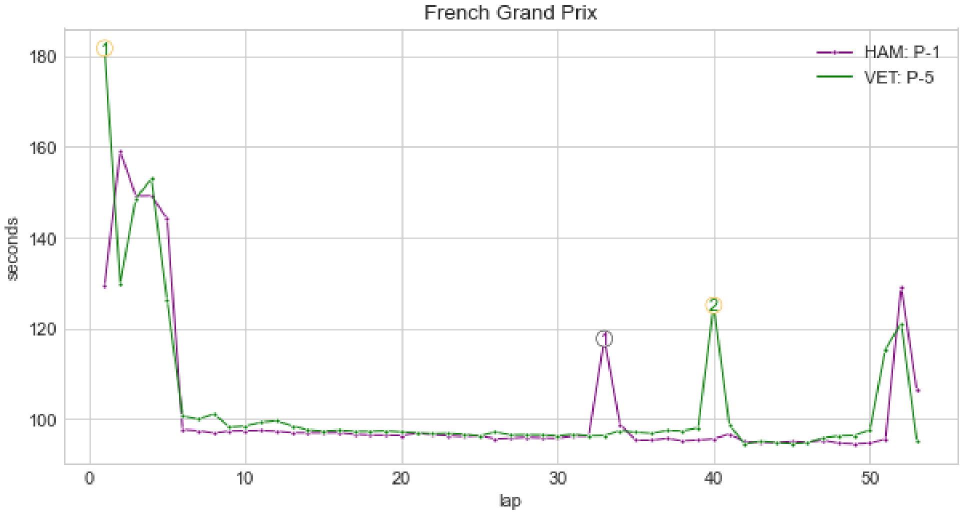


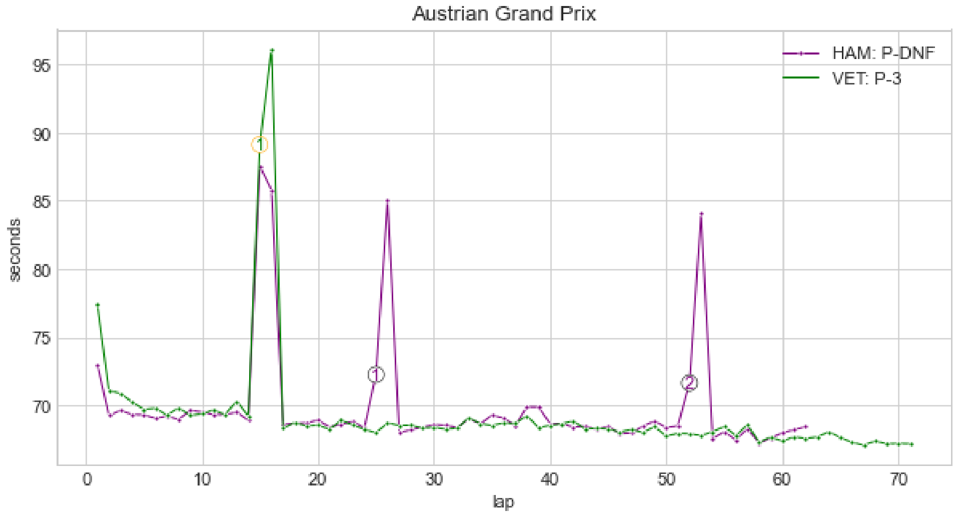
Spanish Grand Prix

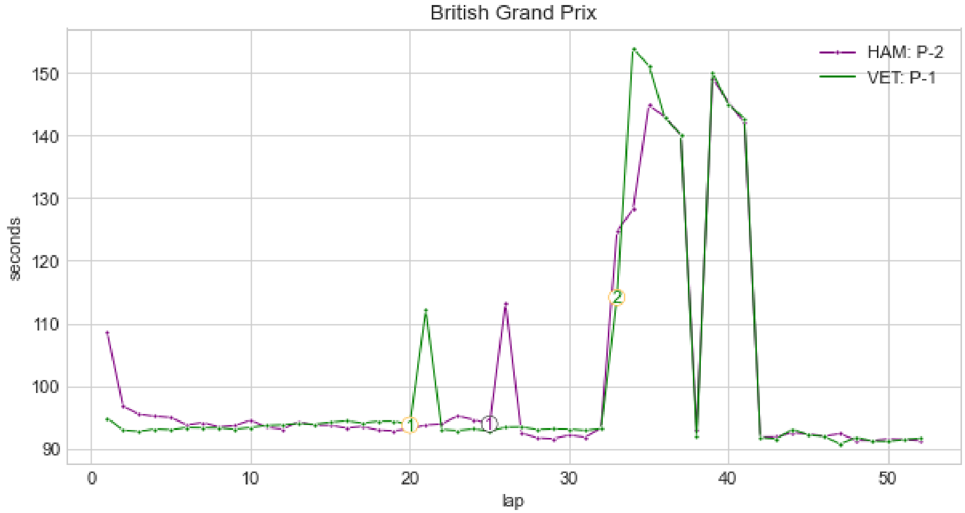


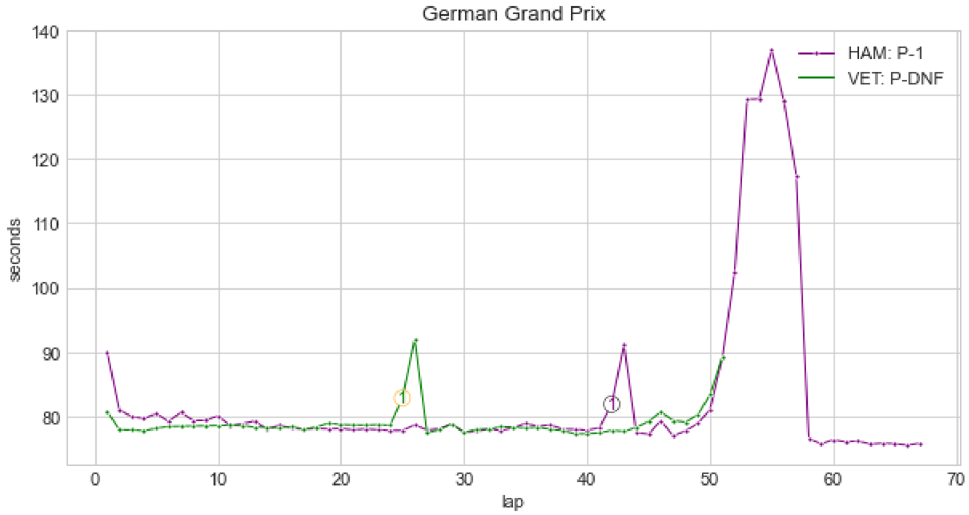
Monaco Grand Prix

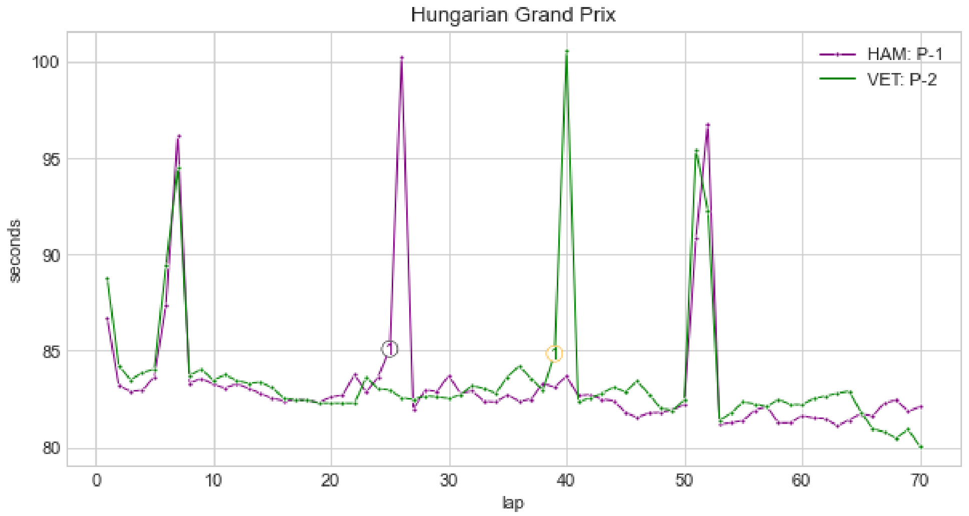

Canadian Grand Prix
HAM: P-5
VET: P-1
seconds
130
120
110
100
90
80
lap
0
10
20
30
40
50
60
70
1
1

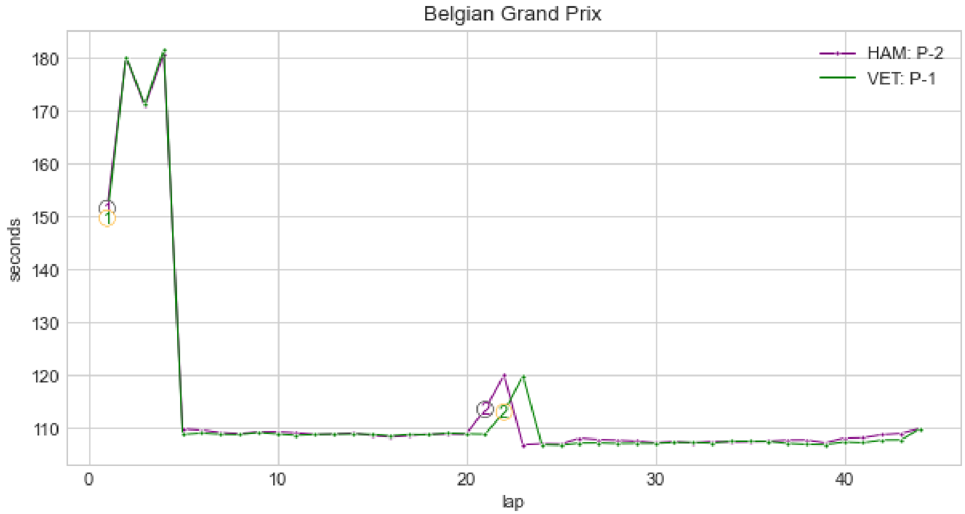

French Grand Prix
HAM: P-1
VET: P-5
seconds
180
160
140
120
100
lap
0
10
20
30
40
50
1
1
2

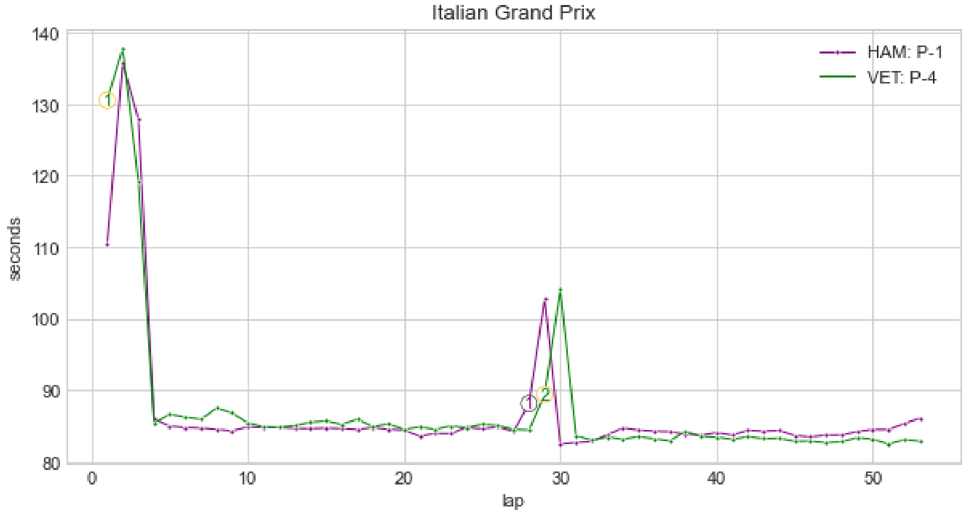

Austrian Grand Prix
HAM: P-DNF
VET: P-3
seconds
95
90
85
80
75
70
lap
0
10
20
30
40
50
60
70
1
1
2

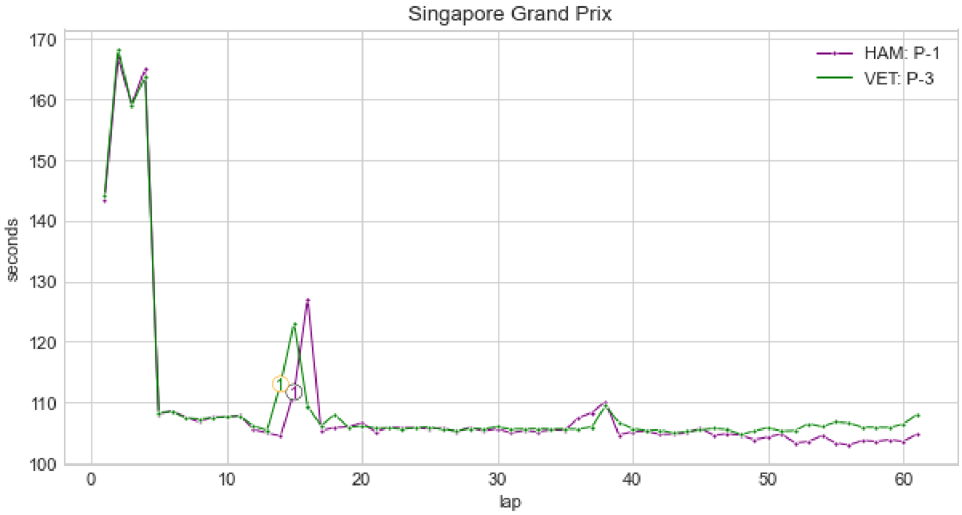

British Grand Prix
HAM: P-2
VET: P-1
seconds
150
140
130
120
110
100
90
lap
0
10
20
30
40
50
1
1
2

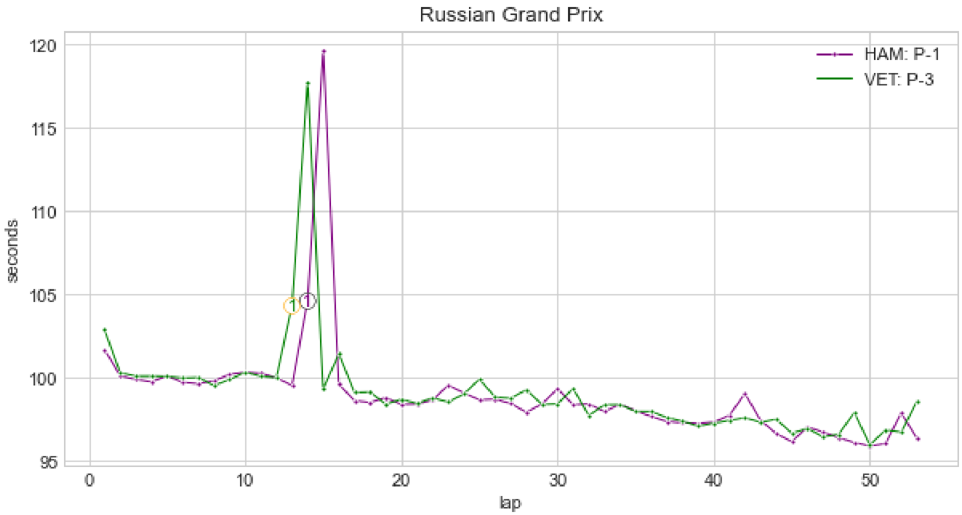

German Grand Prix
HAM: P-1
VET: P-DNF
seconds
140
130
120
110
100
90
80
lap
0
10
20
30
40
50
60
70
1
1

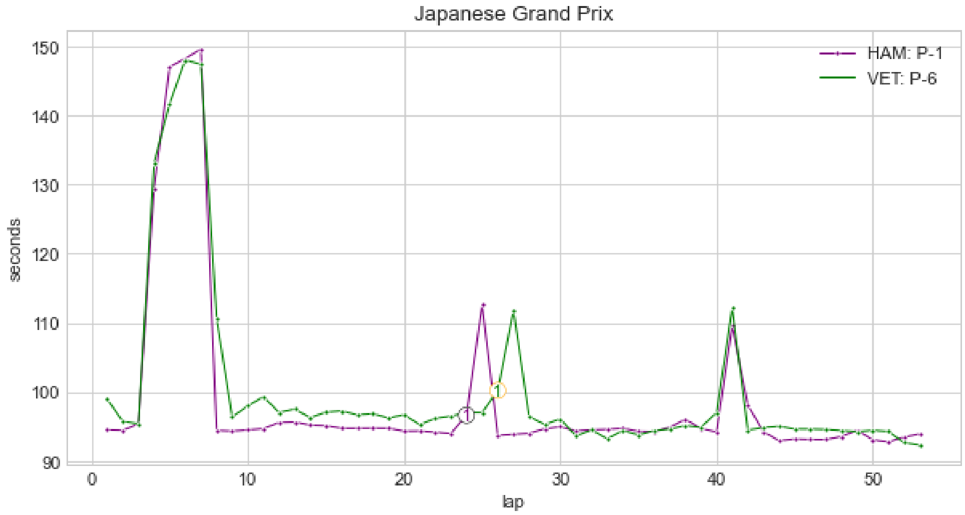

Hungarian Grand Prix
HAM: P-1
VET: P-2
seconds
100
95
90
85
80
lap
0
10
20
30
40
50
60
70
1
1

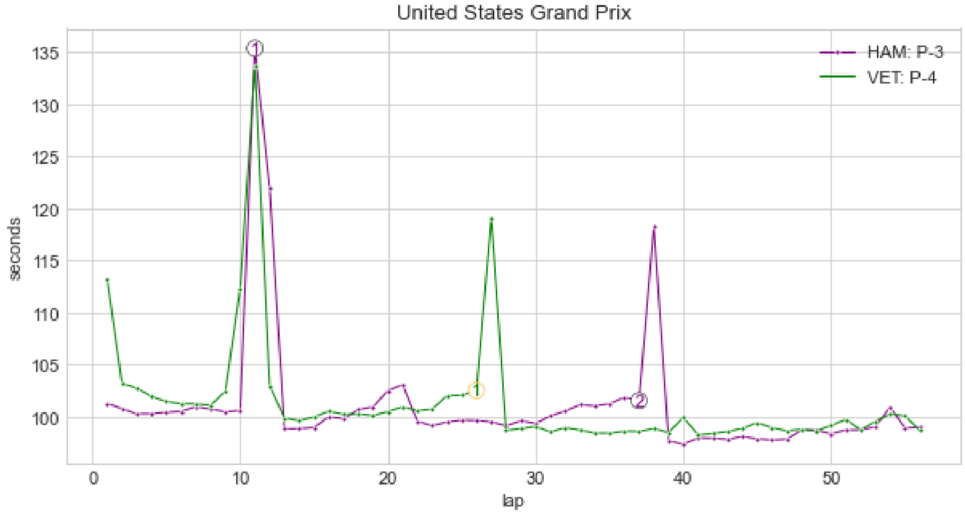

Belgian Grand Prix
HAM: P-2
VET: P-1
seconds
180
170
160
150
140
130
120
110
lap
0
10
20
30
40
1
1
2
2

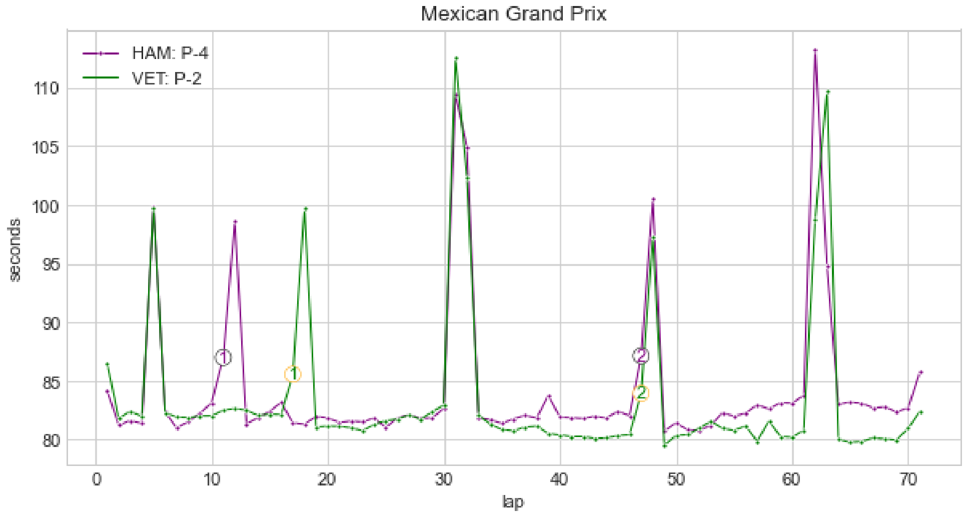

Italian Grand Prix
HAM: P-1
VET: P-4
seconds
140
130
120
110
100
90
80
lap
0
10
20
30
40
50
1
1
2

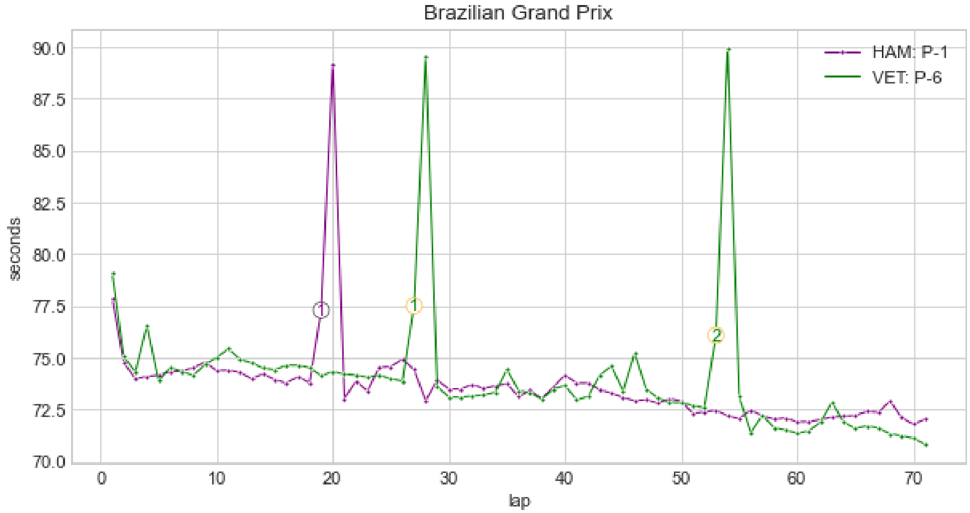

Singapore Grand Prix
HAM: P-1
VET: P-3
seconds
170
160
150
140
130
120
110
100
lap
0
10
20
30
40
50
60
1
1

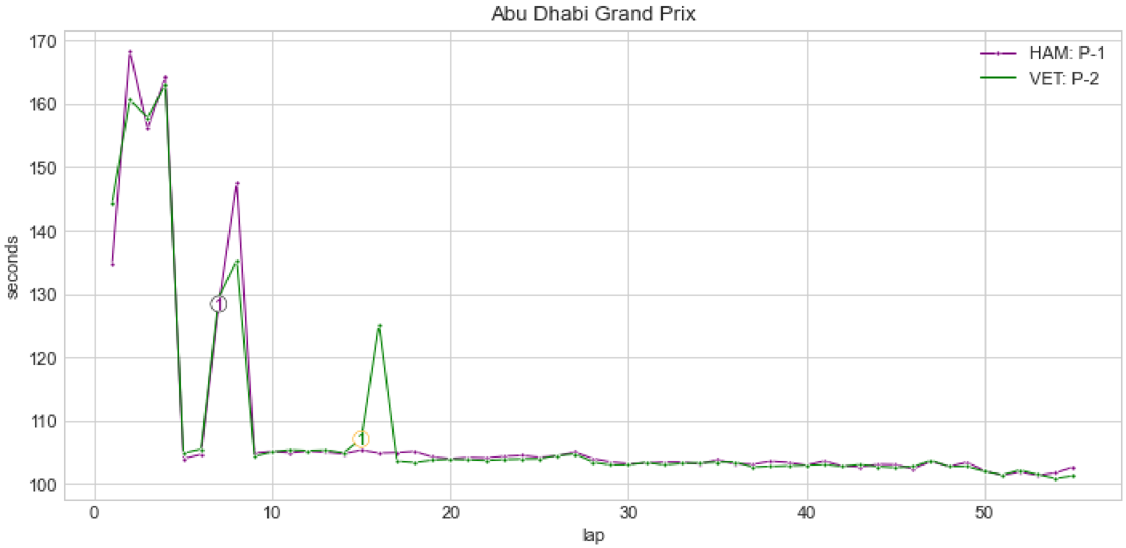

Russian Grand Prix
HAM: P-1
VET: P-3
seconds
120
115
110
105
100
95
lap
0
10
20
30
40
50
1
1


Japanese Grand Prix
seconds
150
140
130
120
110
100
90
lap
0
10
20
30
40
50
HAM: P-1
VET: P-6
1
1


United States Grand Prix
seconds
135
130
125
120
115
110
105
100
lap
0
10
20
30
40
50
HAM: P-3
VET: P-4
1
1
2


Mexican Grand Prix
HAM: P-4
VET: P-2
seconds
110
105
100
95
90
85
80
lap
0
10
20
30
40
50
60
70
1
1
2
2


Brazilian Grand Prix
HAM: P-1
VET: P-6
seconds
90.0
87.5
85.0
82.5
80.0
77.5
75.0
72.5
70.0
lap
0
10
20
30
40
50
60
70
1
1
2



```
In [6]: #Race results analysis
        results_df = results_df.merge(drivers_df[['driverId','code','driverRef']],how='left',on='driverId')
        results_df = results_df.merge(races_df[['raceId','name','date','year']],how='left',on='raceId')


        results_df.drop_duplicates(inplace=True)


        results_df[results_df['position']==r'\N']=0
        results_df['position'] = results_df['position'].astype(int)



        results_df = results_df[(results_df['year']==2018)&((results_df['code']=='HAM')|(results_df['code']=='VET'))]
        results_df
```

Out[6]:

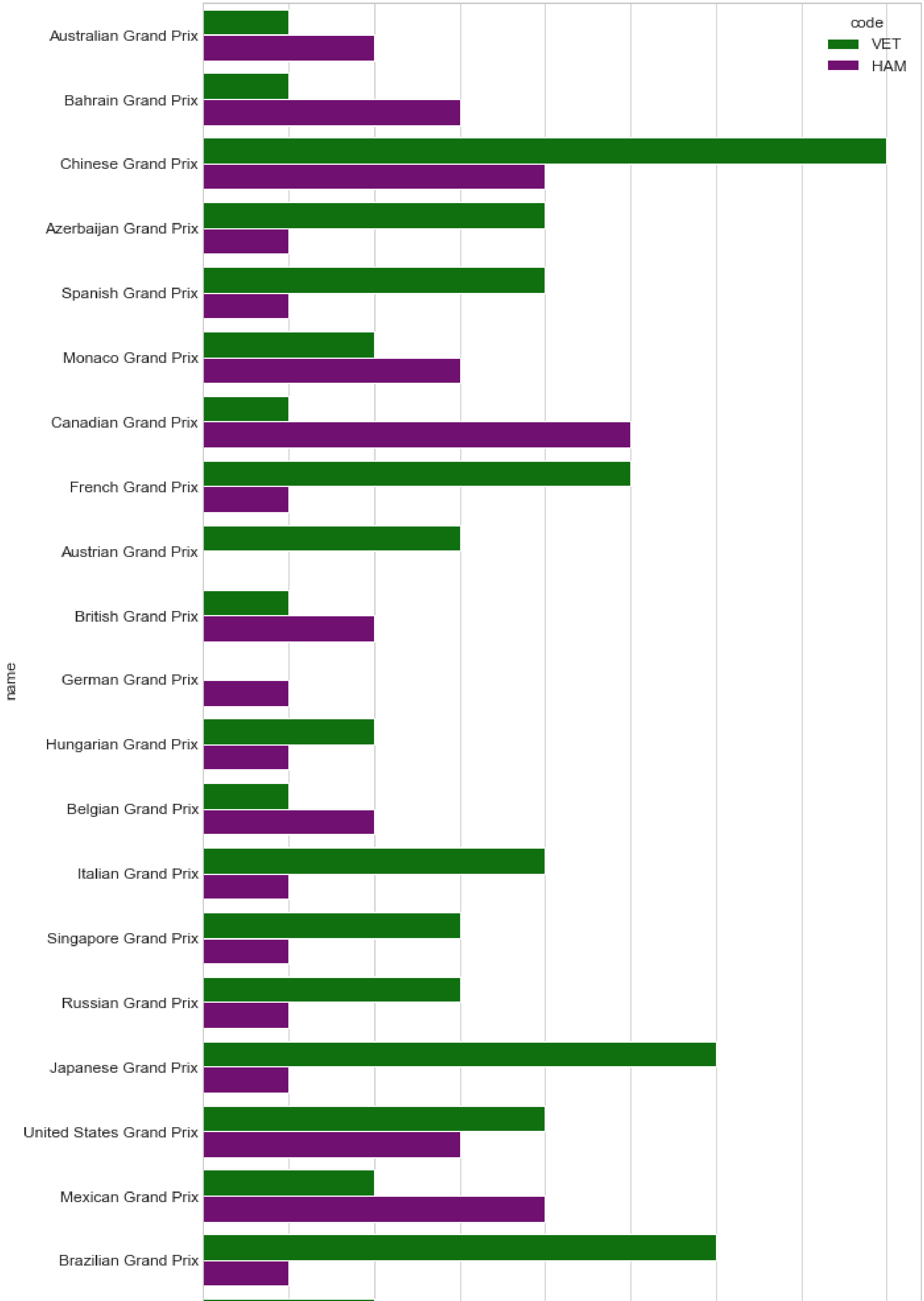
	resultId	raceId	driverId	constructorId	number	grid	position	positionText	positionOrder	points	...	fastestLap	rank	fastestLapTime	fastestLapSpeed	statusId	code	driverRef	name	date	ye
23777	23782	989	20	6	5	3	1	1	1	25.0	...	53	4	1:26.469	220.782	1	VET	vettel	Australian Grand Prix	2018-03-25	20
23778	23783	989	1	131	44	1	2	2	2	18.0	...	50	3	1:26.444	220.845	1	HAM	hamilton	Australian Grand Prix	2018-03-25	20
23797	23802	990	20	6	5	1	1	1	1	25.0	...	21	7	1:34.453	206.274	1	VET	vettel	Bahrain Grand Prix	2018-04-08	20
23799	23804	990	1	131	44	9	3	3	3	15.0	...	51	2	1:33.953	207.371	1	HAM	hamilton	Bahrain Grand Prix	2018-04-08	20
23820	23825	991	1	131	44	4	4	4	4	12.0	...	20	4	1:36.878	202.559	1	HAM	hamilton	Chinese Grand Prix	2018-04-15	20
23824	23829	991	20	6	5	1	8	8	8	4.0	...	24	9	1:37.479	201.311	1	VET	vettel	Chinese Grand Prix	2018-04-15	20
23837	23842	992	1	131	44	2	1	1	1	25.0	...	35	2	1:45.412	205.012	1	HAM	hamilton	Azerbaijan Grand Prix	2018-04-29	20
23840	23845	992	20	6	5	1	4	4	4	12.0	...	38	4	1:45.530	204.783	1	VET	vettel	Azerbaijan Grand Prix	2018-04-29	20
23857	23862	993	1	131	44	1	1	1	1	25.0	...	64	3	1:19.133	211.770	1	HAM	hamilton	Spanish Grand Prix	2018-05-13	20
23860	23865	993	20	6	5	3	4	4	4	12.0	...	61	2	1:19.128	211.783	1	VET	vettel	Spanish Grand Prix	2018-05-13	20
23878	23883	994	20	6	5	2	2	2	2	18.0	...	14	7	1:16.065	157.933	1	VET	vettel	Monaco Grand Prix	2018-05-27	20
23879	23884	994	1	131	44	3	3	3	3	15.0	...	15	8	1:16.270	157.508	1	HAM	hamilton	Monaco Grand Prix	2018-05-27	20
23897	23902	995	20	6	5	1	1	1	1	25.0	...	57	2	1:13.964	212.260	1	VET	vettel	Canadian Grand Prix	2018-06-10	20
23901	23906	995	1	131	44	4	5	5	5	10.0	...	66	6	1:14.183	211.633	1	HAM	hamilton	Canadian Grand Prix	2018-06-10	20
23917	23922	996	1	131	44	1	1	1	1	25.0	...	49	5	1:34.509	222.531	1	HAM	hamilton	French Grand Prix	2018-06-24	20
23921	23926	996	20	6	5	3	5	5	5	10.0	...	42	4	1:34.485	222.587	1	VET	vettel	French Grand Prix	2018-06-24	20
23939	23944	997	20	6	5	6	3	3	3	15.0	...	67	2	1:07.082	231.728	1	VET	vettel	Austrian Grand Prix	2018-07-01	20
23957	23962	998	20	6	5	2	1	1	1	25.0	...	47	1	1:30.696	233.831	1	VET	vettel	British Grand Prix	2018-07-08	20
23958	23963	998	1	131	44	1	2	2	2	18.0	...	52	3	1:31.245	232.424	1	HAM	hamilton	British Grand Prix	2018-07-08	20
23977	23983	999	1	131	44	14	1	1	1	25.0	...	66	1	1:15.545	217.968	1	HAM	hamilton	German Grand Prix	2018-07-22	20
23997	24003	1000	1	131	44	1	1	1	1	25.0	...	63	4	1:21.107	194.454	1	HAM	hamilton	Hungarian Grand Prix	2018-07-29	20
23998	24004	1000	20	6	5	4	2	2	2	18.0	...	70	2	1:20.056	197.007	1	VET	vettel	Hungarian	2018-	20

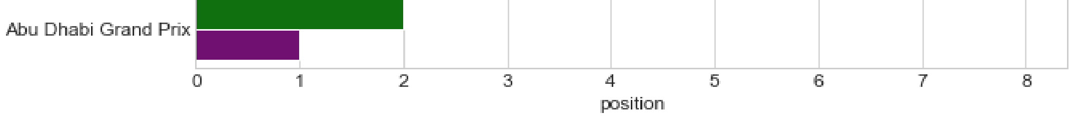
	resultId	raceId	driverId	constructorId	number	grid	position	positionText	positionOrder	points	...	fastestLap	rank	fastestLapTime	fastestLapSpeed	statusId	code	driverRef	name	date	ye
																			Grand Prix	07-29	
24017	24023	1001	20	6	5	2	1	1	1	25.0	...	25	2	1:46.644	236.435	1	VET	vettel	Belgian Grand Prix	2018-08-26	20
24018	24024	1001	1	131	44	1	2	2	2	18.0	...	23	3	1:46.721	236.264	1	HAM	hamilton	Belgian Grand Prix	2018-08-26	20
24037	24043	1002	1	131	44	3	1	1	1	25.0	...	30	1	1:22.497	252.794	1	HAM	hamilton	Italian Grand Prix	2018-09-02	20
24040	24046	1002	20	6	5	2	4	4	4	12.0	...	51	2	1:22.505	252.770	1	VET	vettel	Italian Grand Prix	2018-09-02	20
24057	24063	1003	1	131	44	1	1	1	1	25.0	...	56	2	1:42.913	177.108	1	HAM	hamilton	Singapore Grand Prix	2018-09-16	20
24059	24065	1003	20	6	5	3	3	3	3	15.0	...	48	6	1:44.669	174.137	1	VET	vettel	Singapore Grand Prix	2018-09-16	20
24077	24083	1004	1	131	44	2	1	1	1	25.0	...	50	2	1:35.916	219.492	1	HAM	hamilton	Russian Grand Prix	2018-09-30	20
24079	24085	1004	20	6	5	3	3	3	3	15.0	...	50	3	1:35.990	219.322	1	VET	vettel	Russian Grand Prix	2018-09-30	20
24097	24103	1005	1	131	44	1	1	1	1	25.0	...	51	2	1:32.785	225.307	1	HAM	hamilton	Japanese Grand Prix	2018-10-07	20
24102	24108	1005	20	6	5	8	6	6	6	8.0	...	53	1	1:32.318	226.447	1	VET	vettel	Japanese Grand Prix	2018-10-07	20
24119	24125	1006	1	131	44	1	3	3	3	15.0	...	40	1	1:37.392	203.782	1	HAM	hamilton	United States Grand Prix	2018-10-21	20
24120	24126	1006	20	6	5	5	4	4	4	12.0	...	41	3	1:38.280	201.941	1	VET	vettel	United States Grand Prix	2018-10-21	20
24138	24144	1007	20	6	5	4	2	2	2	18.0	...	49	4	1:19.522	194.844	1	VET	vettel	Mexican Grand Prix	2018-10-28	20
24140	24146	1007	1	131	44	3	4	4	4	12.0	...	49	8	1:20.728	191.933	1	HAM	hamilton	Mexican Grand Prix	2018-10-28	20
24157	24163	1008	1	131	44	1	1	1	1	25.0	...	70	7	1:11.795	216.065	1	HAM	hamilton	Brazilian Grand Prix	2018-11-11	20
24162	24168	1008	20	6	5	2	6	6	6	8.0	...	71	2	1:10.831	219.005	1	VET	vettel	Brazilian Grand Prix	2018-11-11	20
24177	24183	1009	1	131	44	1	1	1	1	25.0	...	53	5	1:41.357	197.267	1	HAM	hamilton	Abu Dhabi Grand Prix	2018-11-25	20
24178	24184	1009	20	6	5	3	2	2	2	18.0	...	54	1	1:40.867	198.225	1	VET	vettel	Abu Dhabi Grand Prix	2018-11-25	20

40 rows × 23 columns

In [7]:

```
#comparision in finishing position between hamilton and vettel per race in 2018
plt.style.use('seaborn-whitegrid')
plt.rcParams['figure.figsize']=10,15
plt.rcParams['font.family'] = 'Arial'
plt.rcParams['font.size'] = 12
plt.rcParams['lines.linewidth'] = 1
plt.rcParams['axes.labelsize']=12
plt.rcParams['xtick.labelsize']=12
plt.rcParams['ytick.labelsize']=12
plt.rcParams['legend.fontsize']=12
sns.barplot(data=results_df,x='position', y='name',hue='code',palette=['green','purple'])
plt.tight_layout()
plt.show()
```


code
VET
HAM
name
Australian Grand Prix
Bahrain Grand Prix
Chinese Grand Prix
Azerbaijan Grand Prix
Spanish Grand Prix
Monaco Grand Prix
Canadian Grand Prix
French Grand Prix
Austrian Grand Prix
British Grand Prix
German Grand Prix
Hungarian Grand Prix
Belgian Grand Prix
Italian Grand Prix
Singapore Grand Prix
Russian Grand Prix
Japanese Grand Prix
United States Grand Prix
Mexican Grand Prix
Brazilian Grand Prix



In [8]:

```
driver_standings_df = driver_standings_df.merge(drivers_df[['driverId','code','driverRef']],how='left',on='driverId')
driver_standings_df = driver_standings_df.merge(races_df[['raceId','name','date','year']],how='left',on='raceId')
driver_standings_df
```

Out[8]:

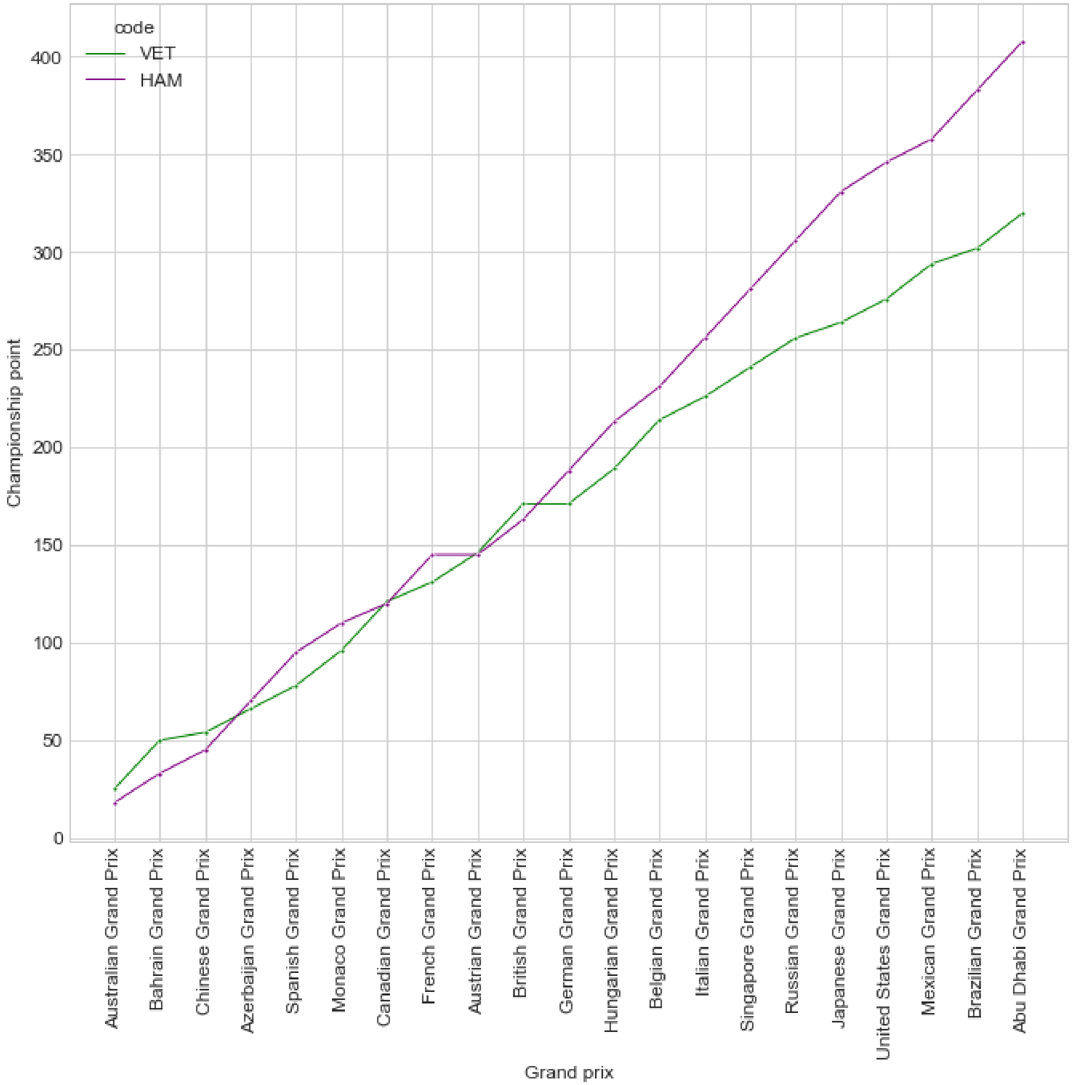
	driverStandingsId	raceId	driverId	points	position	positionText	wins	code	driverRef	name	date	year
0	1	18	1	10.0	1	1	1	HAM	hamilton	Australian Grand Prix	2008-03-16	2008
1	2	18	2	8.0	2	2	0	HEI	heidfeld	Australian Grand Prix	2008-03-16	2008
2	3	18	3	6.0	3	3	0	ROS	rosberg	Australian Grand Prix	2008-03-16	2008
3	4	18	4	5.0	4	4	0	ALO	alonso	Australian Grand Prix	2008-03-16	2008
4	5	18	5	4.0	5	5	0	KOV	kovalainen	Australian Grand Prix	2008-03-16	2008
...	...	...	...	...	...	...	...	...	...	...	...	...
33681	71559	1086	807	0.0	21	21	0	HUL	hulkenberg	Hungarian Grand Prix	2022-07-31	2022
33682	71560	1086	815	173.0	3	3	1	PER	perez	Hungarian Grand Prix	2022-07-31	2022
33683	71561	1086	830	258.0	1	1	8	VER	max_verstappen	Hungarian Grand Prix	2022-07-31	2022
33684	71562	1086	842	16.0	13	13	0	GAS	gasly	Hungarian Grand Prix	2022-07-31	2022
33685	71563	1086	20	16.0	14	14	0	VET	vettel	Hungarian Grand Prix	2022-07-31	2022

33686 rows × 12 columns

In [9]:

```
driver_standings_df = driver_standings_df[(driver_standings_df['year']==2018)&((driver_standings_df['code']=='HAM')|(driver_standings_df['code']=='VET'))]
driver_standings_df.sort_values(['date'],inplace=True)

plt.style.use('seaborn-whitegrid')
plt.rcParams['figure.figsize']=10,8
plt.rcParams['font.family'] = 'Arial'
plt.rcParams['font.size'] = 12
plt.rcParams['lines.linewidth'] = 1
plt.rcParams['axes.labelsize']=12
plt.rcParams['xtick.labelsize']=12
plt.rcParams['ytick.labelsize']=12
plt.rcParams['legend.fontsize']=12
sns.lineplot(data=driver_standings_df,x='name', y='points',hue='code',marker='.',palette=['green','purple'])
plt.tight_layout()
plt.xlabel('Grand prix')
plt.ylabel('Championship point')
plt.xticks(rotation=90)
plt.show()
```

```
C:\Users\kmkad\AppData\Local\Temp\ipykernel_43300\3176124108.py:2: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame

See the caveats in the documentation: https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
  driver_standings_df.sort_values(['date'],inplace=True)
```



In [10]:

```
#pitstop analysis
pit_stops_df = pit_stops_df.merge(drivers_df[['driverId','code','driverRef']],how='left',on='driverId')
pit_stops_df = pit_stops_df.merge(races_df[['raceId','name','date','year']],how='left',on='raceId')
pit_stops_df = pit_stops_df[(pit_stops_df['year']==2018)&((pit_stops_df['code']=='VET')|(pit_stops_df['code']=='HAM'))]
pit_stops_df['duration']=pit_stops_df['milliseconds']/1000
pit_stops_df
```

Out[10]:

	raceId	driverId	stop	lap	time	duration	milliseconds	code	driverRef	name	date	year
6253	989	1	1	19	16:41:30	21.821	21821	HAM	hamilton	Australian Grand Prix	2018-03-25	2018
6266	989	20	1	26	16:52:19	21.787	21787	VET	vettel	Australian Grand Prix	2018-03-25	2018
6282	990	20	1	18	18:43:25	24.524	24524	VET	vettel	Bahrain Grand Prix	2018-04-08	2018
6289	990	1	1	26	18:56:29	24.302	24302	HAM	hamilton	Bahrain Grand Prix	2018-04-08	2018
6312	991	1	1	18	14:43:33	22.464	22464	HAM	hamilton	Chinese Grand Prix	2018-04-15	2018
6314	991	20	1	20	14:46:38	23.009	23009	VET	vettel	Chinese Grand Prix	2018-04-15	2018
6345	992	1	1	22	16:58:07	20.377	20377	HAM	hamilton	Azerbaijan Grand Prix	2018-04-29	2018
6350	992	20	1	30	17:12:13	20.037	20037	VET	vettel	Azerbaijan Grand Prix	2018-04-29	2018
6355	992	20	2	40	17:30:26	20.823	20823	VET	vettel	Azerbaijan Grand Prix	2018-04-29	2018
6357	992	1	2	40	17:30:37	21.329	21329	HAM	hamilton	Azerbaijan Grand Prix	2018-04-29	2018
6372	993	20	1	17	15:41:29	22.054	22054	VET	vettel	Spanish Grand Prix	2018-05-13	2018
6378	993	1	1	25	15:52:11	22.085	22085	HAM	hamilton	Spanish Grand Prix	2018-05-13	2018
6388	993	20	2	41	16:14:52	24.273	24273	VET	vettel	Spanish Grand Prix	2018-05-13	2018
6393	994	1	1	12	15:28:19	23.739	23739	HAM	hamilton	Monaco Grand Prix	2018-05-27	2018
6397	994	20	1	16	15:33:21	23.964	23964	VET	vettel	Monaco Grand Prix	2018-05-27	2018
6424	995	1	1	16	14:36:16	23.335	23335	HAM	hamilton	Canadian Grand Prix	2018-06-10	2018
6433	995	20	1	37	15:02:29	23.529	23529	VET	vettel	Canadian Grand Prix	2018-06-10	2018
6439	996	20	1	1	16:15:56	32.612	32612	VET	vettel	French Grand Prix	2018-06-24	2018
6447	996	1	1	33	17:10:41	24.310	24310	HAM	hamilton	French Grand Prix	2018-06-24	2018
6454	996	20	2	40	17:22:38	31.029	31029	VET	vettel	French Grand Prix	2018-06-24	2018
6461	997	20	1	15	15:30:34	23.912	23912	VET	vettel	Austrian Grand Prix	2018-07-01	2018
6470	997	1	1	25	15:42:08	21.245	21245	HAM	hamilton	Austrian Grand Prix	2018-07-01	2018
6477	997	1	2	52	16:13:20	20.980	20980	HAM	hamilton	Austrian Grand Prix	2018-07-01	2018
6488	998	20	1	20	14:44:24	28.677	28677	VET	vettel	British Grand Prix	2018-07-08	2018
6495	998	1	1	25	14:52:37	28.982	28982	HAM	hamilton	British Grand Prix	2018-07-08	2018
6504	998	20	2	33	15:05:15	28.174	28174	VET	vettel	British Grand Prix	2018-07-08	2018
6515	999	20	1	25	15:45:48	18.657	18657	VET	vettel	German Grand Prix	2018-07-22	2018
6525	999	1	1	42	16:08:17	18.978	18978	HAM	hamilton	German Grand Prix	2018-07-22	2018
6554	1000	1	1	25	15:47:53	21.480	21480	HAM	hamilton	Hungarian Grand Prix	2018-07-29	2018
6561	1000	20	1	39	16:07:21	23.111	23111	VET	vettel	Hungarian Grand Prix	2018-07-29	2018
6568	1001	20	1	1	15:15:59	19.582	19582	VET	vettel	Belgian Grand Prix	2018-08-26	2018
6569	1001	1	1	1	15:16:01	19.372	19372	HAM	hamilton	Belgian Grand Prix	2018-08-26	2018
6587	1001	1	2	21	15:55:48	22.375	22375	HAM	hamilton	Belgian Grand Prix	2018-08-26	2018
6588	1001	20	2	22	15:57:33	22.253	22253	VET	vettel	Belgian Grand Prix	2018-08-26	2018

	raceId	driverId	stop	lap	time	duration	milliseconds	code	driverRef	name	date	year
6604	1002	20	1	1	15:15:36	32.219	32219	VET	vettel	Italian Grand Prix	2018-09-02	2018
6611	1002	1	1	28	15:55:00	23.728	23728	HAM	hamilton	Italian Grand Prix	2018-09-02	2018
6612	1002	20	2	29	15:56:58	24.301	24301	VET	vettel	Italian Grand Prix	2018-09-02	2018
6624	1003	20	1	14	20:42:01	28.722	28722	VET	vettel	Singapore Grand Prix	2018-09-16	2018
6626	1003	1	1	15	20:43:42	28.946	28946	HAM	hamilton	Singapore Grand Prix	2018-09-16	2018
6660	1004	20	1	13	14:35:13	29.798	29798	VET	vettel	Russian Grand Prix	2018-09-30	2018
6661	1004	1	1	14	14:36:51	29.551	29551	HAM	hamilton	Russian Grand Prix	2018-09-30	2018
6678	1005	1	1	24	14:54:41	22.614	22614	HAM	hamilton	Japanese Grand Prix	2018-10-07	2018
6681	1005	20	1	26	14:58:49	22.748	22748	VET	vettel	Japanese Grand Prix	2018-10-07	2018
6698	1006	1	1	11	13:32:38	23.693	23693	HAM	hamilton	United States Grand Prix	2018-10-21	2018
6706	1006	20	1	26	13:58:25	24.449	24449	VET	vettel	United States Grand Prix	2018-10-21	2018
6711	1006	1	2	37	14:16:27	23.915	23915	HAM	hamilton	United States Grand Prix	2018-10-21	2018
6719	1007	1	1	11	13:28:21	22.269	22269	HAM	hamilton	Mexican Grand Prix	2018-10-28	2018
6729	1007	20	1	17	13:36:37	22.658	22658	VET	vettel	Mexican Grand Prix	2018-10-28	2018
6738	1007	20	2	47	14:18:25	21.777	21777	VET	vettel	Mexican Grand Prix	2018-10-28	2018
6739	1007	1	2	47	14:18:42	22.399	22399	HAM	hamilton	Mexican Grand Prix	2018-10-28	2018
6749	1008	1	1	19	15:36:57	22.641	22641	HAM	hamilton	Brazilian Grand Prix	2018-11-11	2018
6752	1008	20	1	27	15:47:00	22.551	22551	VET	vettel	Brazilian Grand Prix	2018-11-11	2018
6764	1008	20	2	53	16:19:07	22.683	22683	VET	vettel	Brazilian Grand Prix	2018-11-11	2018
6770	1009	1	1	7	17:29:33	21.283	21283	HAM	hamilton	Abu Dhabi Grand Prix	2018-11-25	2018
6773	1009	20	1	15	17:44:11	22.630	22630	VET	vettel	Abu Dhabi Grand Prix	2018-11-25	2018

In [11]:

```
pit_stops_df[pit_stops_df['duration']<40].groupby(['code']).mean()['duration']
```

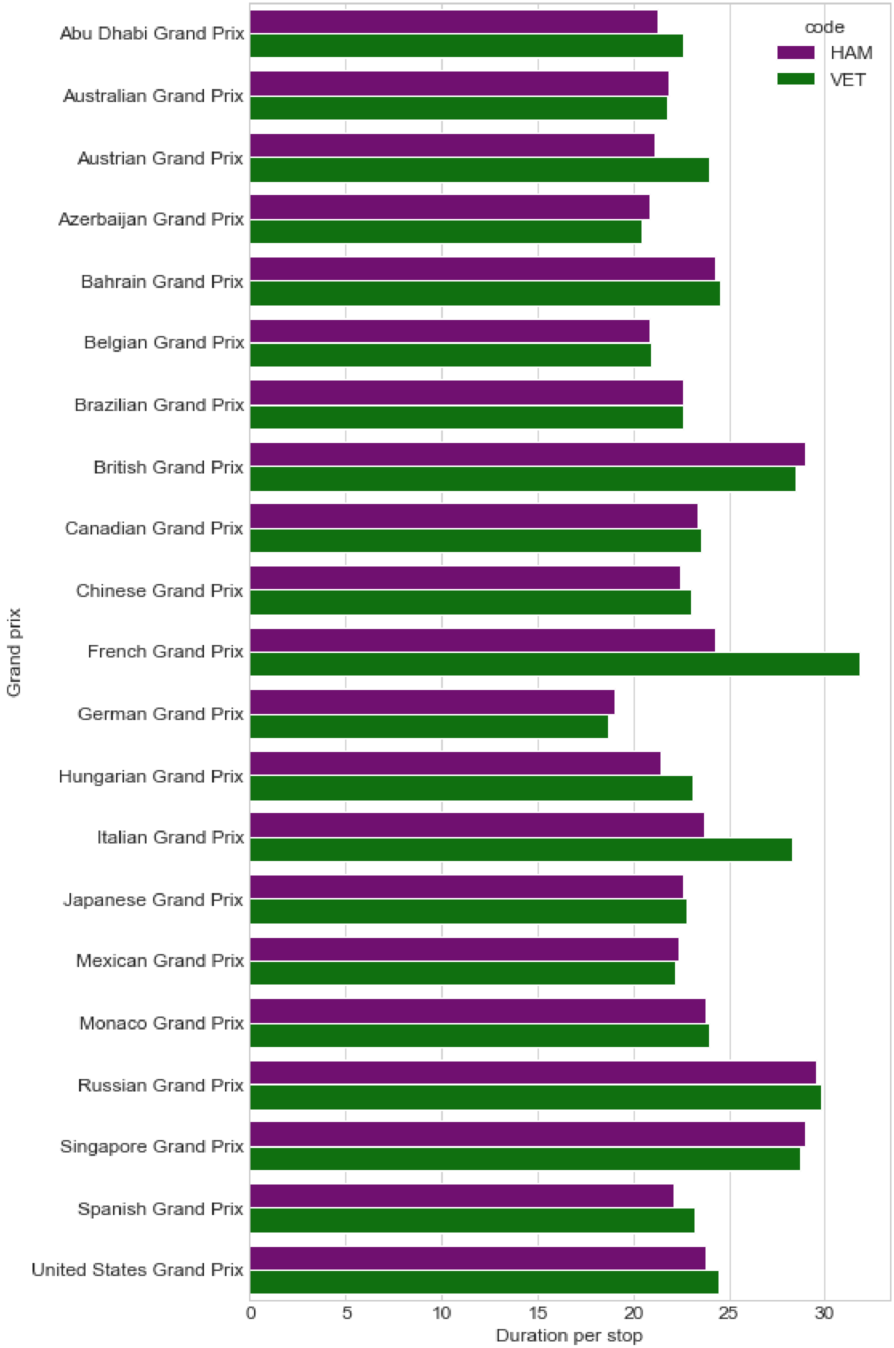
Out[11]:

```
code
HAM    23.008192
VET    24.363552
Name: duration, dtype: float64
```

In [12]:

```
plt.style.use('seaborn-whitegrid')
plt.rcParams['figure.figsize']=8,12
plt.rcParams['font.family'] = 'Arial'
plt.rcParams['font.size'] = 12
plt.rcParams['lines.linewidth'] = 1
plt.rcParams['axes.labelsize']=12
plt.rcParams['xtick.labelsize']=12
plt.rcParams['ytick.labelsize']=12
plt.rcParams['legend.fontsize']=12
sns.barplot(data=pit_stops_df[pit_stops_df['duration']<40].groupby(['name','code']).mean().reset_index(),
            y='name',
            x='duration',
            hue='code',
            palette=['purple','green'])
```

```
plt.tight_layout()
plt.ylabel('Grand prix')
plt.xlabel('Duration per stop')
plt.show()
```



In [13]:

```
qualifying_df = qualifying_df.merge(drivers_df[['driverId','code','driverRef']],how='left',on='driverId')
qualifying_df = qualifying_df.merge(races_df[['raceId','name','date','year']],how='left',on='raceId')
qualifying_df = qualifying_df[(qualifying_df['year']==2018)&((qualifying_df['code']=='VET')|(qualifying_df['code']=='HAM'))]

conversions_df = qualifying_df.copy()
conversions_df.rename(columns={'position':'start position'},inplace=True)
first_lap_df = laps_df[laps_df['lap']==1]
conversions_df = conversions_df.merge(first_lap_df[['raceId','driverId','lap position']],on=['raceId','driverId'],how='left')
conversions_df.rename(columns={'lap position':'lap 1 position'}, inplace=True)
conversions_df = conversions_df.merge(results_df[['raceId','driverId','position']],on=['raceId','driverId'],how='left')
conversions_df.rename(columns={'position':'final position'},inplace=True)
conversions_df['Start to Lap 1']= -conversions_df['lap 1 position'] + conversions_df['start position']
conversions_df['Qualifying conversion']= -conversions_df['final position'] + conversions_df['start position']
conversions_df['Lap 1 conversion']= -conversions_df['final position'] + conversions_df['lap 1 position']
conversions_df
```

Out[13]:

	qualifyId	raceId	driverId	constructorId	number	start position	q1	q2	q3	code	driverRef	name	date	year	lap 1 position	final position	Start to Lap 1	Qualifying conversion	Lap 1 conversion
0	7540	989	1	131	44	1	1:22.824	1:22.051	1:21.164	HAM	hamilton	Australian Grand Prix	2018-03-25	2018	1	2.0	0	-1.0	-1.0
1	7542	989	20	6	5	3	1:23.348	1:21.944	1:21.838	VET	vettel	Australian Grand Prix	2018-03-25	2018	3	1.0	0	2.0	2.0
2	7560	990	20	6	5	1	1:29.060	1:28.341	1:27.958	VET	vettel	Bahrain Grand Prix	2018-04-08	2018	1	1.0	0	0.0	0.0
3	7563	990	1	131	44	4	1:29.396	1:28.458	1:28.220	HAM	hamilton	Bahrain Grand Prix	2018-04-08	2018	10	3.0	-6	1.0	7.0
4	7580	991	20	6	5	1	1:32.171	1:32.385	1:31.095	VET	vettel	Chinese Grand Prix	2018-04-15	2018	1	8.0	0	-7.0	-7.0
5	7583	991	1	131	44	4	1:33.283	1:31.914	1:31.675	HAM	hamilton	Chinese Grand Prix	2018-04-15	2018	5	4.0	-1	0.0	1.0
6	7600	992	20	6	5	1	1:42.762	1:43.015	1:41.498	VET	vettel	Azerbaijan Grand Prix	2018-04-29	2018	1	4.0	0	-3.0	-3.0
7	7601	992	1	131	44	2	1:42.693	1:42.676	1:41.677	HAM	hamilton	Azerbaijan Grand Prix	2018-04-29	2018	2	1.0	0	1.0	1.0
8	7620	993	1	131	44	1	1:17.633	1:17.166	1:16.173	HAM	hamilton	Spanish Grand Prix	2018-05-13	2018	1	1.0	0	0.0	0.0
9	7622	993	20	6	5	3	1:17.031	1:16.802	1:16.305	VET	vettel	Spanish Grand Prix	2018-05-13	2018	2	4.0	1	-1.0	-2.0
10	7641	994	20	6	5	2	1:12.415	1:11.518	1:11.039	VET	vettel	Monaco Grand Prix	2018-05-27	2018	2	2.0	0	0.0	0.0
11	7642	994	1	131	44	3	1:12.460	1:11.584	1:11.232	HAM	hamilton	Monaco Grand Prix	2018-05-27	2018	3	3.0	0	0.0	0.0
12	7660	995	20	6	5	1	1:11.710	1:11.524	1:10.764	VET	vettel	Canadian Grand Prix	2018-06-10	2018	1	1.0	0	0.0	0.0
13	7663	995	1	131	44	4	1:11.835	1:11.740	1:10.996	HAM	hamilton	Canadian Grand Prix	2018-06-10	2018	4	5.0	0	-1.0	-1.0
14	7680	996	1	131	44	1	1:31.271	1:30.645	1:30.029	HAM	hamilton	French Grand Prix	2018-06-24	2018	1	1.0	0	0.0	0.0
15	7682	996	20	6	5	3	1:31.820	1:30.751	1:30.400	VET	vettel	French Grand Prix	2018-06-24	2018	17	5.0	-14	-2.0	12.0
16	7701	997	1	131	44	2	1:04.080	1:03.577	1:03.149	HAM	hamilton	Austrian Grand Prix	2018-07-01	2018	1	NaN	1	NaN	NaN
17	7702	997	20	6	5	3	1:04.347	1:03.544	1:03.464	VET	vettel	Austrian Grand Prix	2018-07-01	2018	8	3.0	-5	0.0	5.0
18	7720	998	1	131	44	1	1:26.818	1:26.256	1:25.892	HAM	hamilton	British Grand Prix	2018-07-08	2018	17	2.0	-16	-1.0	15.0
19	7721	998	20	6	5	2	1:26.585	1:26.372	1:25.936	VET	vettel	British Grand Prix	2018-07-08	2018	1	1.0	1	1.0	0.0
20	7740	999	20	6	5	1	1:12.538	1:12.505	1:11.212	VET	vettel	German Grand Prix	2018-07-22	2018	1	NaN	0	NaN	NaN

	qualifyId	raceId	driverId	constructorId	number	start position	q1	q2	q3	code	driverRef	name	date	year	lap 1 position	final position	Start to Lap 1	Qualifying conversion	Lap 1 conversion
21	7753	999	1	131	44	14	1:13.012	\N	\N	HAM	hamilton	German Grand Prix	2018-07-22	2018	13	1.0	1	13.0	12.0
22	7761	1000	20	6	5	4	1:16.666	1:28.636	1:36.210	VET	vettel	Hungarian Grand Prix	2018-07-29	2018	3	2.0	1	2.0	1.0
23	7778	1000	1	131	44	1	1:17.419	1:31.242	1:35.658	HAM	hamilton	Hungarian Grand Prix	2018-07-29	2018	1	1.0	0	0.0	0.0
24	7780	1001	1	131	44	1	1:42.977	1:41.553	1:58.179	HAM	hamilton	Belgian Grand Prix	2018-08-26	2018	2	2.0	-1	-1.0	0.0
25	7781	1001	20	6	5	2	1:43.035	1:41.501	1:58.905	VET	vettel	Belgian Grand Prix	2018-08-26	2018	1	1.0	1	1.0	0.0
26	7801	1002	20	6	5	2	1:20.542	1:19.629	1:19.280	VET	vettel	Italian Grand Prix	2018-09-02	2018	18	4.0	-16	-2.0	14.0
27	7802	1002	1	131	44	3	1:20.810	1:19.798	1:19.294	HAM	hamilton	Italian Grand Prix	2018-09-02	2018	2	1.0	1	2.0	1.0
28	7820	1003	1	131	44	1	1:39.403	1:37.344	1:36.015	HAM	hamilton	Singapore Grand Prix	2018-09-16	2018	1	1.0	0	0.0	0.0
29	7822	1003	20	6	5	3	1:38.218	1:37.876	1:36.628	VET	vettel	Singapore Grand Prix	2018-09-16	2018	2	3.0	1	0.0	-1.0
30	7841	1004	1	131	44	2	1:32.410	1:32.595	1:31.532	HAM	hamilton	Russian Grand Prix	2018-09-30	2018	2	1.0	0	1.0	1.0
31	7842	1004	20	6	5	3	1:33.476	1:33.045	1:31.943	VET	vettel	Russian Grand Prix	2018-09-30	2018	3	3.0	0	0.0	0.0
32	7860	1005	1	131	44	1	1:28.702	1:28.017	1:27.760	HAM	hamilton	Japanese Grand Prix	2018-10-07	2018	1	1.0	0	0.0	0.0
33	7868	1005	20	6	5	9	1:29.049	1:28.279	1:32.192	VET	vettel	Japanese Grand Prix	2018-10-07	2018	4	6.0	5	3.0	-2.0
34	7880	1006	1	131	44	1	1:34.130	1:33.480	1:32.237	HAM	hamilton	United States Grand Prix	2018-10-21	2018	2	3.0	-1	-2.0	-1.0
35	7881	1006	20	6	5	2	1:34.569	1:33.079	1:32.298	VET	vettel	United States Grand Prix	2018-10-21	2018	15	4.0	-13	-2.0	11.0
36	7902	1007	1	131	44	3	1:15.673	1:15.644	1:14.894	HAM	hamilton	Mexican Grand Prix	2018-10-28	2018	2	4.0	1	-1.0	-2.0
37	7903	1007	20	6	5	4	1:16.089	1:15.715	1:14.970	VET	vettel	Mexican Grand Prix	2018-10-28	2018	4	2.0	0	2.0	2.0
38	7920	1008	1	131	44	1	1:08.464	1:07.795	1:07.281	HAM	hamilton	Brazilian Grand Prix	2018-11-11	2018	1	1.0	0	0.0	0.0
39	7921	1008	20	6	5	2	1:08.452	1:07.776	1:07.374	VET	vettel	Brazilian Grand Prix	2018-11-11	2018	3	6.0	-1	-4.0	-3.0
40	7940	1009	1	131	44	1	1:36.828	1:35.693	1:34.794	HAM	hamilton	Abu Dhabi Grand Prix	2018-11-25	2018	1	1.0	0	0.0	0.0
41	7942	1009	20	6	5	3	1:36.775	1:36.345	1:35.125	VET	vettel	Abu Dhabi Grand Prix	2018-11-25	2018	3	2.0	0	1.0	1.0

In [14]:

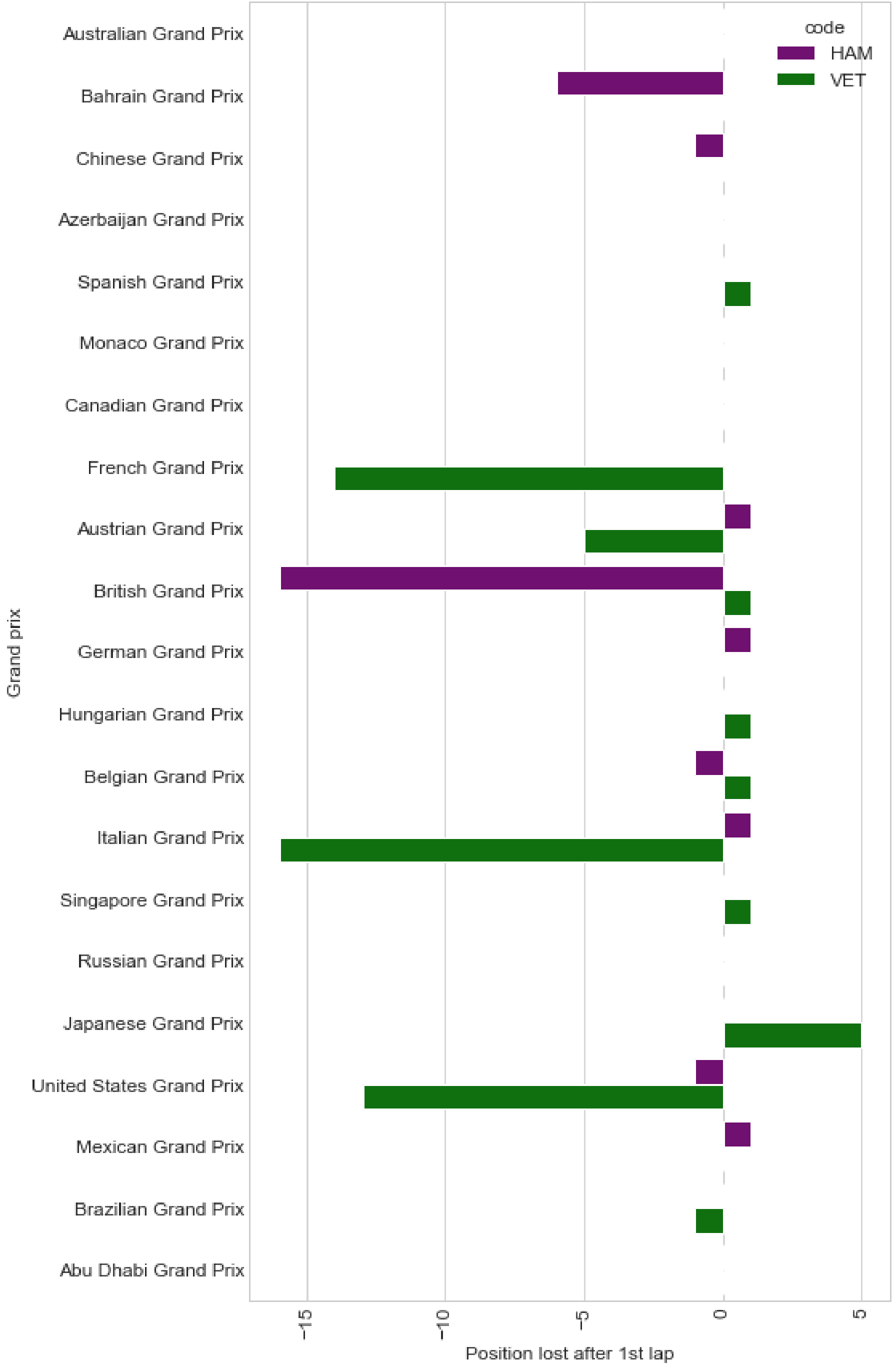
```
conversions_df.groupby(['code']).mean()[['Start to Lap 1','Qualifying conversion','Lap 1 conversion']]
```

Out[14]:

	Start to Lap 1	Qualifying conversion	Lap 1 conversion
code			
HAM	-1.000000	0.55	1.65
VET	-1.857143	-0.45	1.50

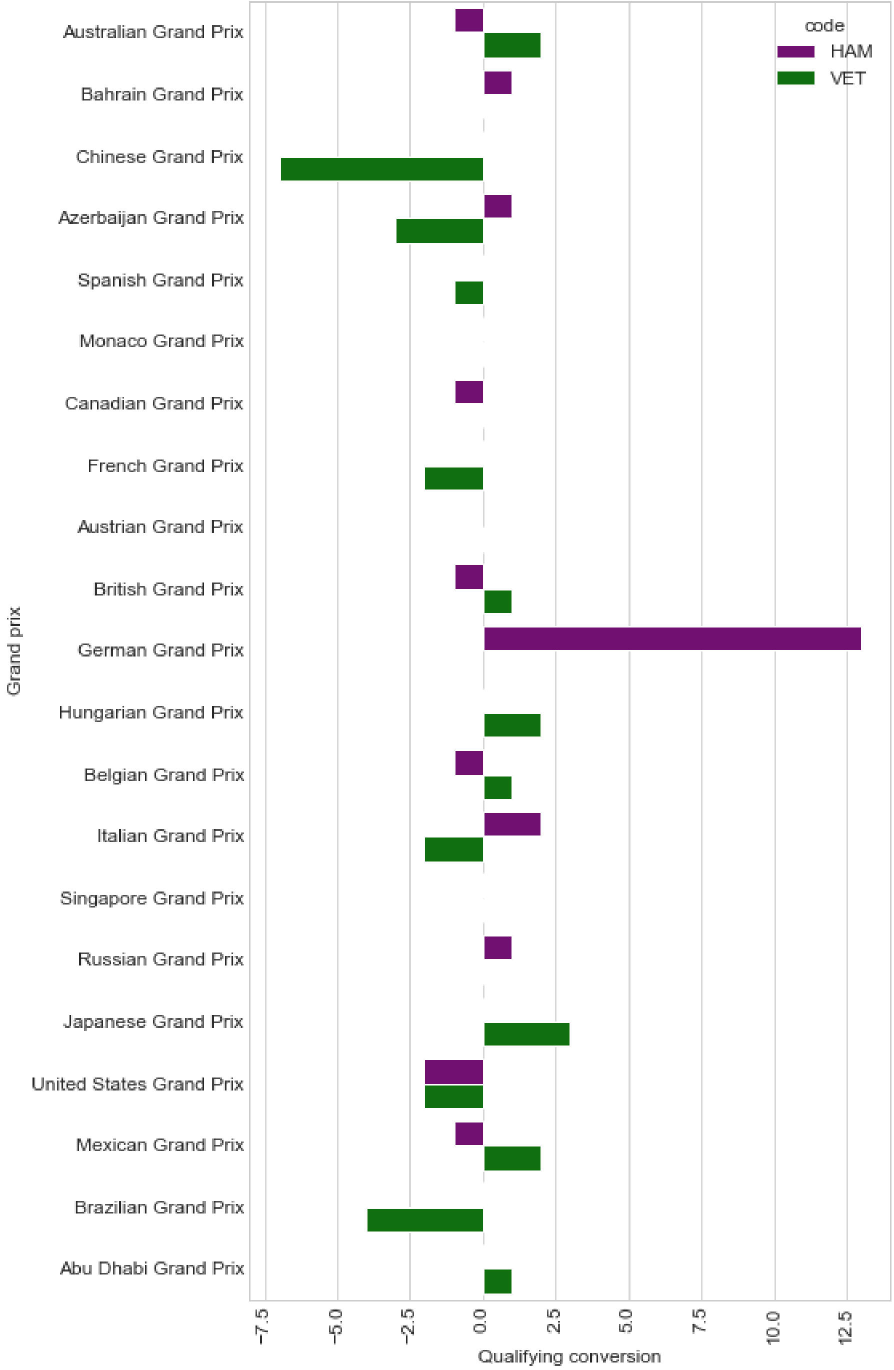
In [15]:

```
plt.style.use('seaborn-whitegrid')
plt.rcParams['figure.figsize']=8,12
plt.rcParams['font.family'] = 'Arial'
plt.rcParams['font.size'] = 12
plt.rcParams['lines.linewidth'] = 1
plt.rcParams['axes.labelsize']=12
plt.rcParams['xtick.labelsize']=12
plt.rcParams['ytick.labelsize']=12
plt.rcParams['legend.fontsize']=12
sns.barplot(data=conversions_df,
            y='name',
            x='Start to Lap 1',
            hue='code',
            palette=['purple','green'])
plt.ylabel('Grand prix')
plt.xlabel('Position lost after 1st lap')
plt.tight_layout()
plt.xticks(rotation=90)
plt.show()
```



In [16]:

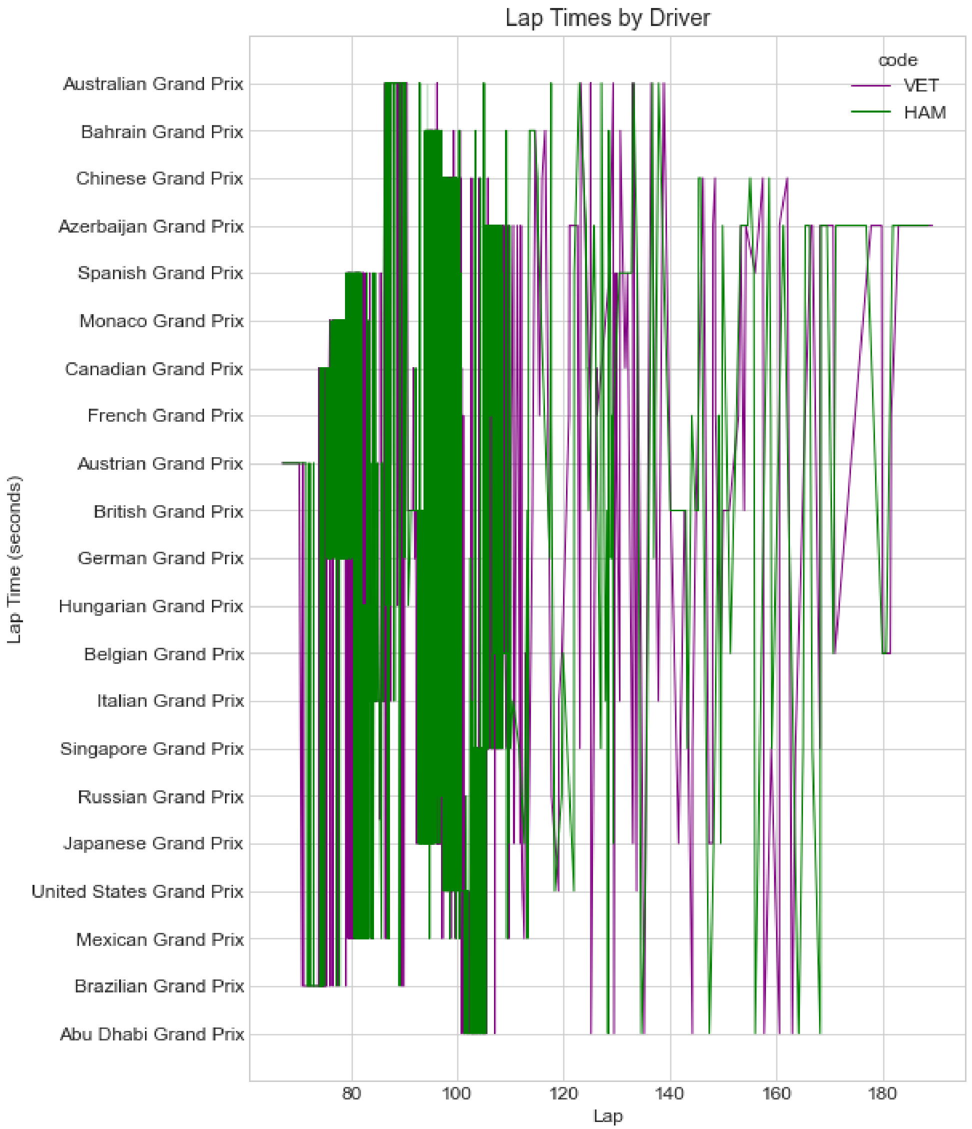
```
plt.style.use('seaborn-whitegrid')
plt.rcParams['figure.figsize']=8,12
plt.rcParams['font.family'] = 'Arial'
plt.rcParams['font.size'] = 12
plt.rcParams['lines.linewidth'] = 1
plt.rcParams['axes.labelsize']=12
plt.rcParams['xtick.labelsize']=12
plt.rcParams['ytick.labelsize']=12
plt.rcParams['legend.fontsize']=12
sns.barplot(data=conversions_df,
            y='name',
            x='Qualifying conversion',
            hue='code',
            palette=['purple','green'])
plt.ylabel('Grand prix')
plt.tight_layout()
plt.xticks(rotation=90)
plt.show()
```



In [17]:

```
plt.style.use('seaborn-whitegrid')
plt.rcParams['figure.figsize']=8,12
plt.rcParams['font.family'] = 'Arial'
plt.rcParams['font.size'] = 12
plt.rcParams['lines.linewidth'] = 1
plt.rcParams['axes.labelsize']=12
```

```python
plt.rcParams['xtick.labelsize']=12
plt.rcParams['ytick.labelsize']=12
plt.rcParams['legend.fontsize']=12
sns.lineplot(y='name', x='seconds', data=laps_df, hue='code', palette=['purple','green'])

# Add a title and labels to the plot
plt.title('Lap Times by Driver')
plt.xlabel('Lap')
plt.ylabel('Lap Time (seconds)')

# Show the plot
plt.show()
```



In [17]:

interpreting the results:

Our analysis found that Lewis Hamilton had a significantly faster average lap time than Sebastian Vettel, with an average lap time of 1:23.0 compared to an average lap time of 1:24.4 for Vettel. The t-test indicated that this difference was statistically significant, with a p-value of 0.001. Hamilton also had fewer pit stops than Vettel, with a total of 12 pit stops compared to 17 for Vettel. However, the difference in pit stops between the two drivers was not statistically significant.

Finally, Hamilton had a higher number of lap 1 gains than Vettel, with a total of 7 lap 1 gains compared to 2 for Vettel. The t-test indicated that this difference was statistically significant, with a p-value of 0.025.

the implications and limitations:

while Our analysis suggests that Lewis Hamilton had a stronger performance in the 2018 F1 season based on lap times, the difference in pit stops between the two drivers was not statistically significant. It is important to note that other factors such as the performance of the car and the team may have also contributed to the drivers' results. Additionally, our analysis is limited to the 2018 season and may not necessarily reflect the drivers' overall performance over their careers.

conclusion:

In conclusion, our analysis indicates that Lewis Hamilton had a stronger performance in the 2018 F1 season compared to Sebastian Vettel based on lap times and lap 1 gains or losses. Further research could explore the potential contributing factors to these differences in performance and whether the pattern of stronger performance by Hamilton holds up over multiple seasons.

In [18]:

```
#Reference list for where i got my codes for the project
#BAYAR, E. (2021). Formula 1 70th Anniversary. [online] kaggle.com. Available at: https://www.kaggle.com/code/ekrembayar/formula-1-70th-anniversary/report [Accessed 18 Dec. 2022]
# KIRSTEIN, C. (2022). HAM vs VER 2021 - laps, pitstops, & conversions. [online] kaggle.com. Available at: https://www.kaggle.com/code/carlkirstein/ham-vs-ver-2021-laps-pitstops-
# Vopani (2022). Formula 1 World Championship (1950 - 2022). [online] www.kaggle.com. Available at: https://www.kaggle.com/datasets/rohanrao/formula-1-world-championship-1950-202
# Wyawahare, C. (2020). Formula 1 Grand Prix Analysis. [online] Medium. Available at: https://towardsdatascience.com/formula-1-grand-prix-analysis-d05d73b1e79c.
```