In [1]:

```
# importing python libraries
import numpy as np
import pandas as pd
import matplotlib.pyplot as plt # visualizing data
%matplotlib inline
import seaborn as sns
```

In [2]:

```
# import csv file
df = pd.read_csv('Amazon Sales data.csv', encoding= 'unicode_escape')
```

In [3]:

```
df.head()
```

Out[3]:

	Region	Country	Item Type	Sales Channel	Order Priority	Order Date	Order ID	Ship Date	Units Sold	Unit Price	Unit Cost	Total Revenue	Total Cost	Total Profit
0	Australia and Oceania	Tuvalu	Baby Food	Offline	H	5/28/2010	669165933	6/27/2010	9925	255.28	159.42	2533654.00	1582243.50	951410.50
1	Central America and the Caribbean	Grenada	Cereal	Online	C	8/22/2012	963881480	9/15/2012	2804	205.70	117.11	576782.80	328376.44	248406.36
2	Europe	Russia	Office Supplies	Offline	L	5/2/2014	341417157	5/8/2014	1779	651.21	524.96	1158502.59	933903.84	224598.75
3	Sub-Saharan Africa	Sao Tome and Principe	Fruits	Online	C	6/20/2014	514321792	7/5/2014	8102	9.33	6.92	75591.66	56065.84	19525.82
4	Sub-Saharan Africa	Rwanda	Office Supplies	Offline	L	2/1/2013	115456712	2/6/2013	5062	651.21	524.96	3296425.02	2657347.52	639077.50

In [4]:

```
#Check Duplicates
df['Order ID'].duplicated()
```

Out[4]:

```
0     False
1     False
2     False
3     False
4     False
      ...  
95    False
96    False
97    False
98    False
99    False
Name: Order ID, Length: 100, dtype: bool
```

In [5]:

```
df.drop_duplicates('Order ID')
```

Out[5]:

	Region	Country	Item Type	Sales Channel	Order Priority	Order Date	Order ID	Ship Date	Units Sold	Unit Price	Unit Cost	Total Revenue	Total Cost	Total Profit
0	Australia and Oceania	Tuvalu	Baby Food	Offline	H	5/28/2010	669165933	6/27/2010	9925	255.28	159.42	2533654.00	1582243.50	951410.50
1	Central America and the Caribbean	Grenada	Cereal	Online	C	8/22/2012	963881480	9/15/2012	2804	205.70	117.11	576782.80	328376.44	248406.36
2	Europe	Russia	Office Supplies	Offline	L	5/2/2014	341417157	5/8/2014	1779	651.21	524.96	1158502.59	933903.84	224598.75
3	Sub-Saharan Africa	Sao Tome and Principe	Fruits	Online	C	6/20/2014	514321792	7/5/2014	8102	9.33	6.92	75591.66	56065.84	19525.82
4	Sub-Saharan Africa	Rwanda	Office Supplies	Offline	L	2/1/2013	115456712	2/6/2013	5062	651.21	524.96	3296425.02	2657347.52	639077.50
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
95	Sub-Saharan Africa	Mali	Clothes	Online	M	7/26/2011	512878119	9/3/2011	888	109.28	35.84	97040.64	31825.92	65214.72
96	Asia	Malaysia	Fruits	Offline	L	11/11/2011	810711038	12/28/2011	6267	9.33	6.92	58471.11	43367.64	15103.47
97	Sub-Saharan Africa	Sierra Leone	Vegetables	Offline	C	6/1/2016	728815257	6/29/2016	1485	154.06	90.93	228779.10	135031.05	93748.05
98	North America	Mexico	Personal Care	Offline	M	7/30/2015	559427106	8/8/2015	5767	81.73	56.67	471336.91	326815.89	144521.02
99	Sub-Saharan Africa	Mozambique	Household	Offline	L	2/10/2012	665095412	2/15/2012	5367	668.27	502.54	3586605.09	2697132.18	889472.91

100 rows × 14 columns

In [6]:

```
df.describe()
```

Out[6]:

	Order ID	Units Sold	Unit Price	Unit Cost	Total Revenue	Total Cost	Total Profit
count	1.000000e+02	100.000000	100.000000	100.000000	1.000000e+02	1.000000e+02	1.000000e+02
mean	5.550204e+08	5128.710000	276.761300	191.048000	1.373488e+06	9.318057e+05	4.416820e+05
std	2.606153e+08	2794.484562	235.592241	188.208181	1.460029e+06	1.083938e+06	4.385379e+05
min	1.146066e+08	124.000000	9.330000	6.920000	4.870260e+03	3.612240e+03	1.258020e+03
25%	3.389225e+08	2836.250000	81.730000	35.840000	2.687212e+05	1.688680e+05	1.214436e+05
50%	5.577086e+08	5382.500000	179.880000	107.275000	7.523144e+05	3.635664e+05	2.907680e+05
75%	7.907551e+08	7369.000000	437.200000	263.330000	2.212045e+06	1.613870e+06	6.358288e+05
max	9.940222e+08	9925.000000	668.270000	524.960000	5.997055e+06	4.509794e+06	1.719922e+06

In [7]:

```
# Assuming 'df' to  DataFrame
df['Order Date'] = pd.to_datetime(df['Order Date'])
```

In [8]:

```
df.describe()
```

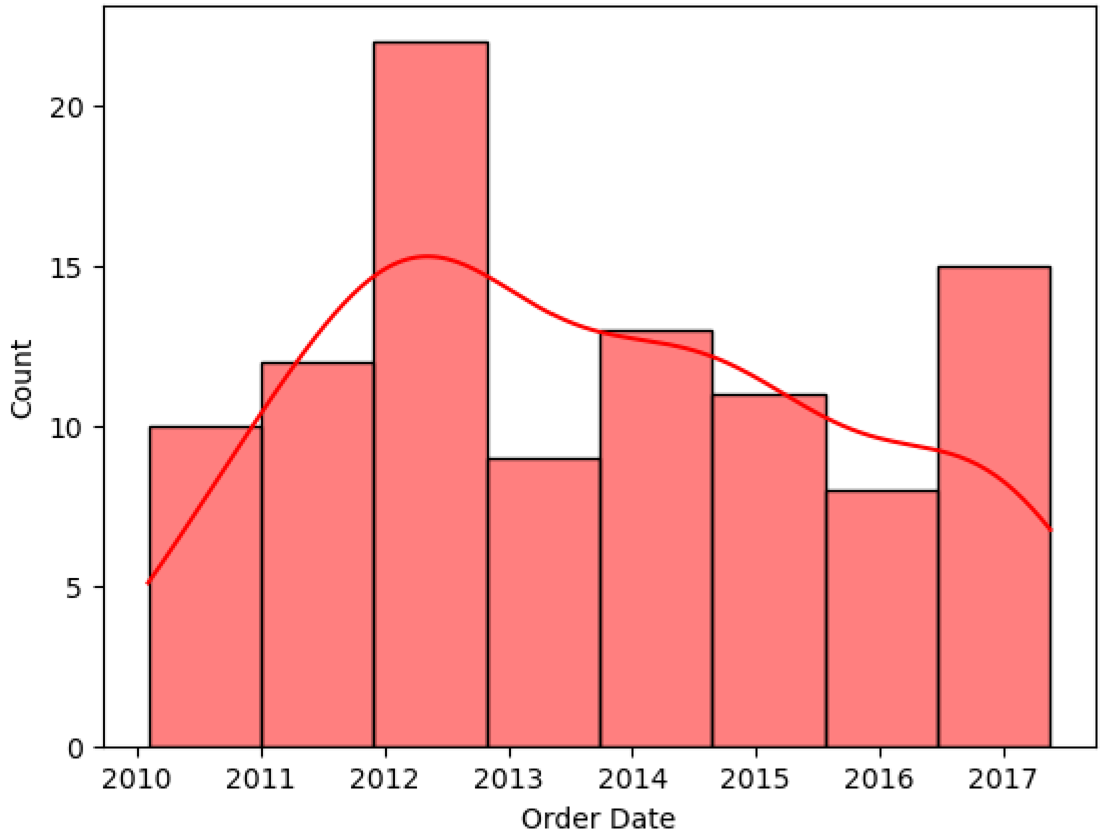
Out[8]:

	Order Date	Order ID	Units Sold	Unit Price	Unit Cost	Total Revenue	Total Cost	Total Profit
count	100	1.000000e+02	100.000000	100.000000	100.000000	1.000000e+02	1.000000e+02	1.000000e+02
mean	2013-09-16 14:09:36	5.550204e+08	5128.710000	276.761300	191.048000	1.373488e+06	9.318057e+05	4.416820e+05
min	2010-02-02 00:00:00	1.146066e+08	124.000000	9.330000	6.920000	4.870260e+03	3.612240e+03	1.258020e+03
25%	2012-02-14 12:00:00	3.389225e+08	2836.250000	81.730000	35.840000	2.687212e+05	1.688680e+05	1.214436e+05
50%	2013-07-12 12:00:00	5.577086e+08	5382.500000	179.880000	107.275000	7.523144e+05	3.635664e+05	2.907680e+05
75%	2015-04-07 00:00:00	7.907551e+08	7369.000000	437.200000	263.330000	2.212045e+06	1.613870e+06	6.358288e+05
max	2017-05-22 00:00:00	9.940222e+08	9925.000000	668.270000	524.960000	5.997055e+06	4.509794e+06	1.719922e+06
std	NaN	2.606153e+08	2794.484562	235.592241	188.208181	1.460029e+06	1.083938e+06	4.385379e+05

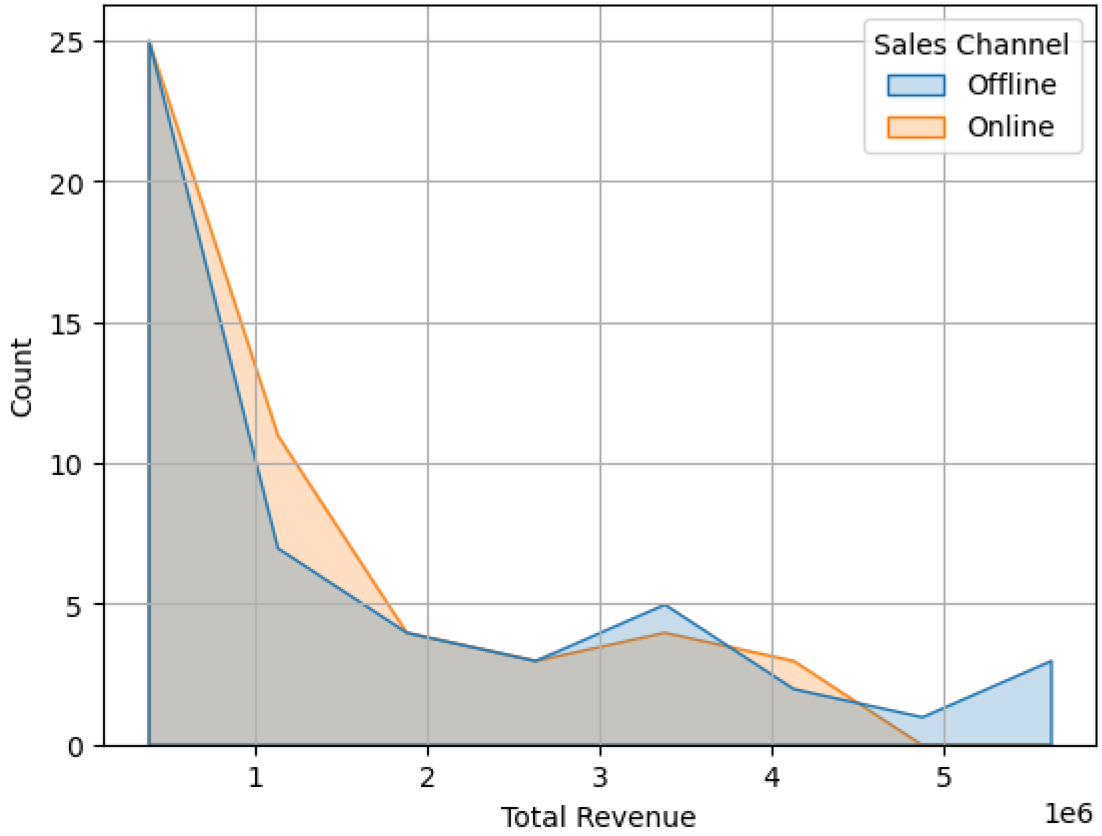
Exploratory Data Analysis

In [9]:
```
#Most of The Order Place Year
sns.histplot(x='Order Date',data=df,kde=True,color='Red')
```

Out[9]:
```
<Axes: xlabel='Order Date', ylabel='Count'>
```



In [11]:
```
#Online and offline Total Revenue
sns.histplot(x='Total Revenue',data=df,hue='Sales Channel',element='poly')
plt.grid()
```

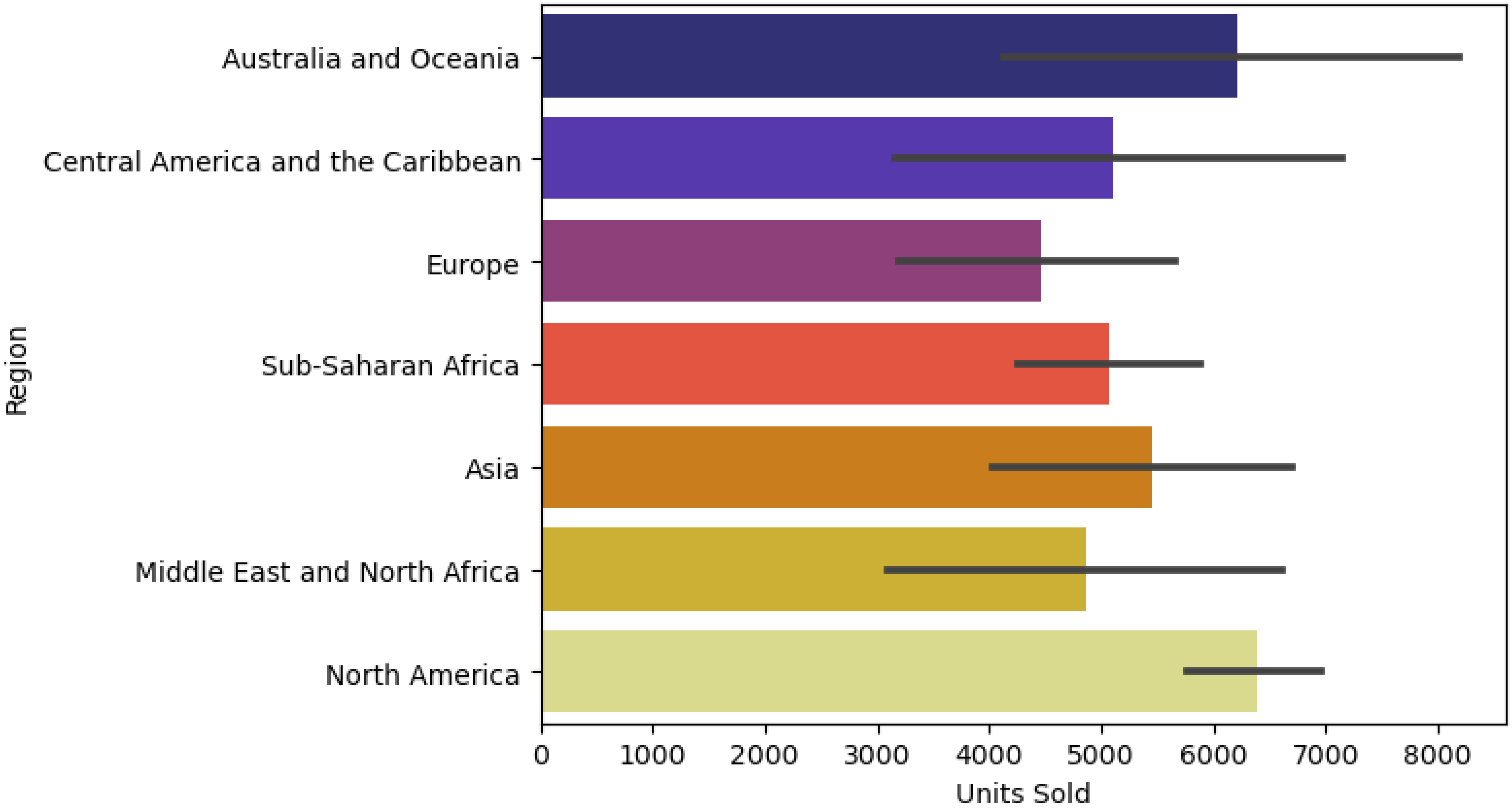


In [12]:

```
#Total Units Sold By Region
sns.barplot(y='Region',x='Units Sold',data=df,palette='CMRmap')
```

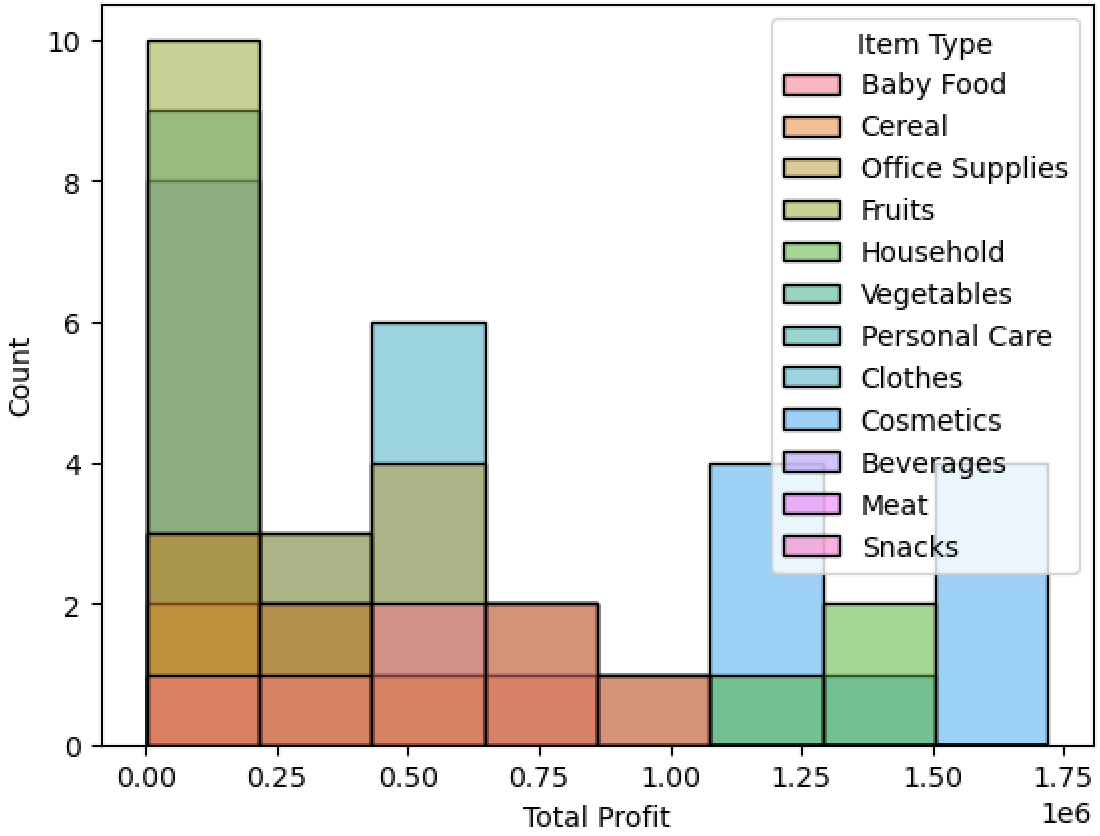
Out[12]:

```
<Axes: xlabel='Units Sold', ylabel='Region'>
```



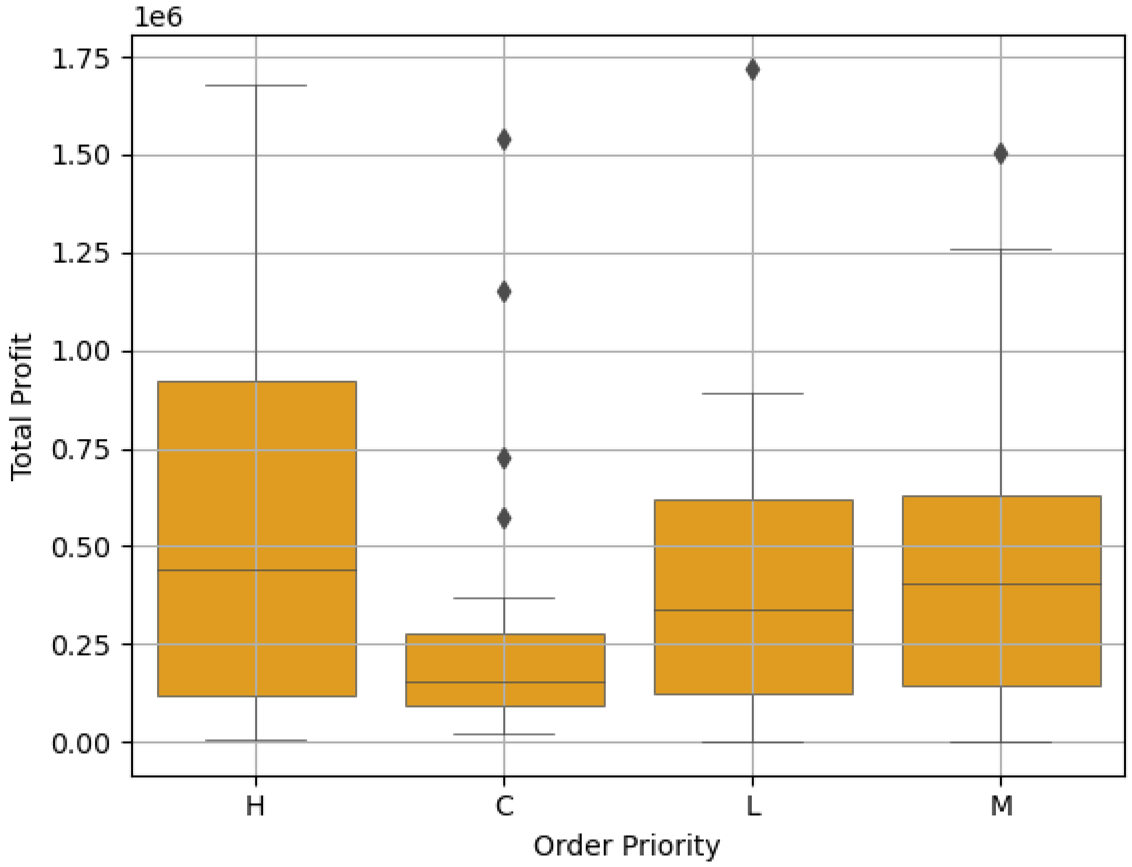
```
In [13]: #Total Profit By Item Type
         sns.histplot(x='Total Profit',hue='Item Type',data=df)
```

```
Out[13]: <Axes: xlabel='Total Profit', ylabel='Count'>
```



In [14]:

```
#Total Profit by Order Priority
sns.boxplot(y='Total Profit',data=df,x='Order Priority',color='Orange',linewidth=0.5)
plt.grid()
```

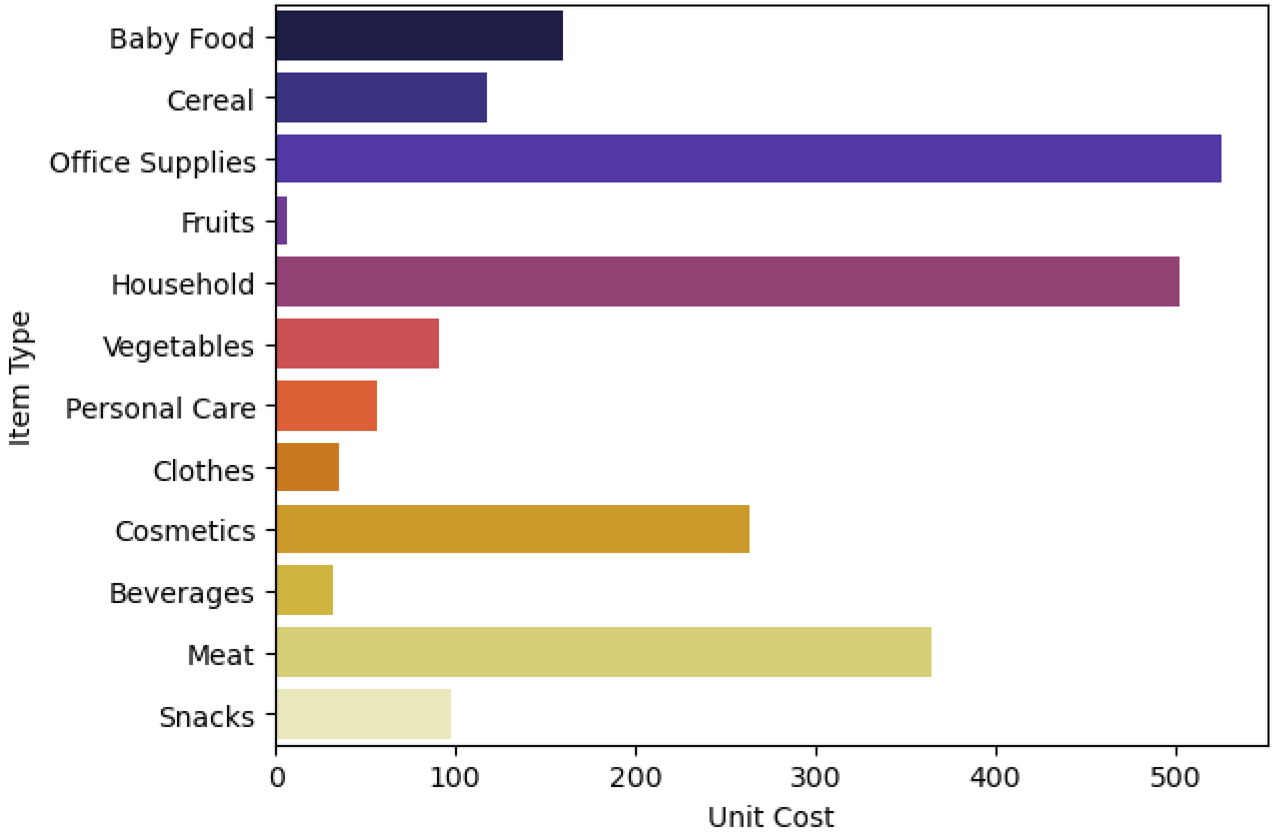


In [16]:

```
#Unit Cost By Item Type
sns.barplot(x='Unit Cost',y='Item Type',data=df,palette='CMRmap')
```

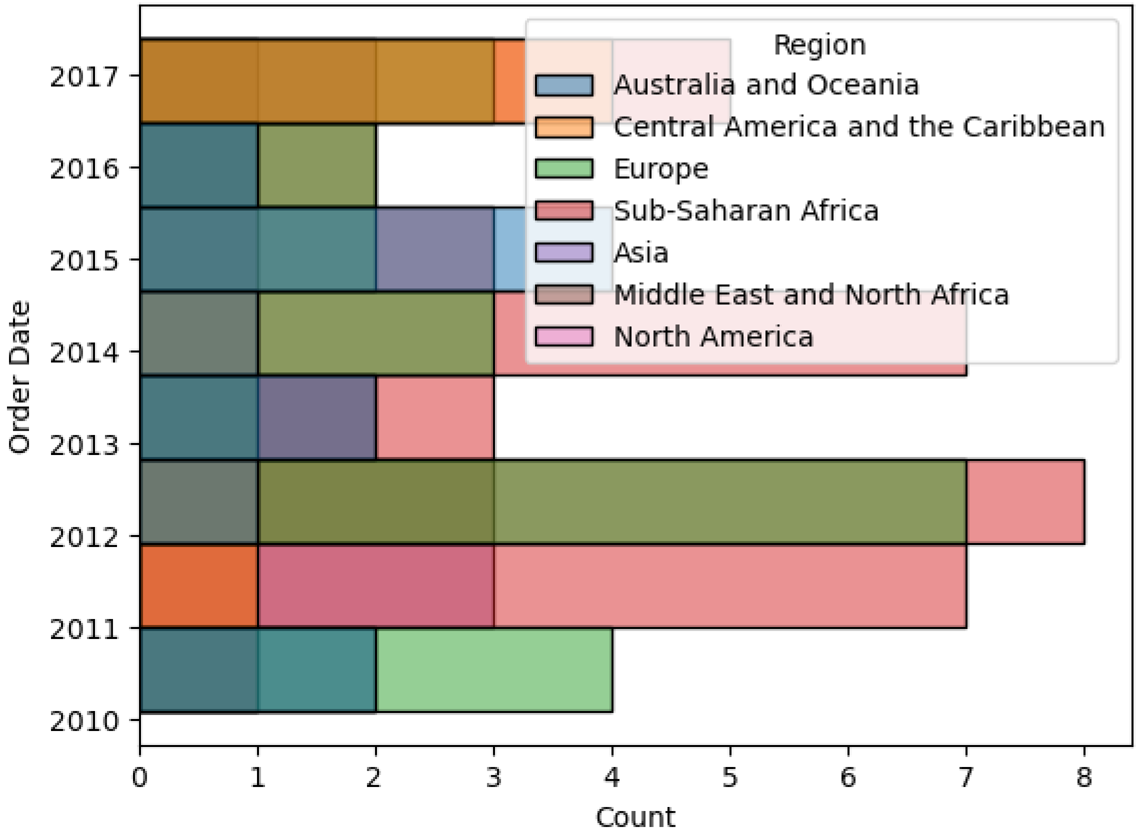
Out[16]:

```
<Axes: xlabel='Unit Cost', ylabel='Item Type'>
```



```
In [17]: #Order Year By Region
         sns.histplot(y='Order Date',hue='Region',data=df)
```

```
Out[17]: <Axes: xlabel='Count', ylabel='Order Date'>
```

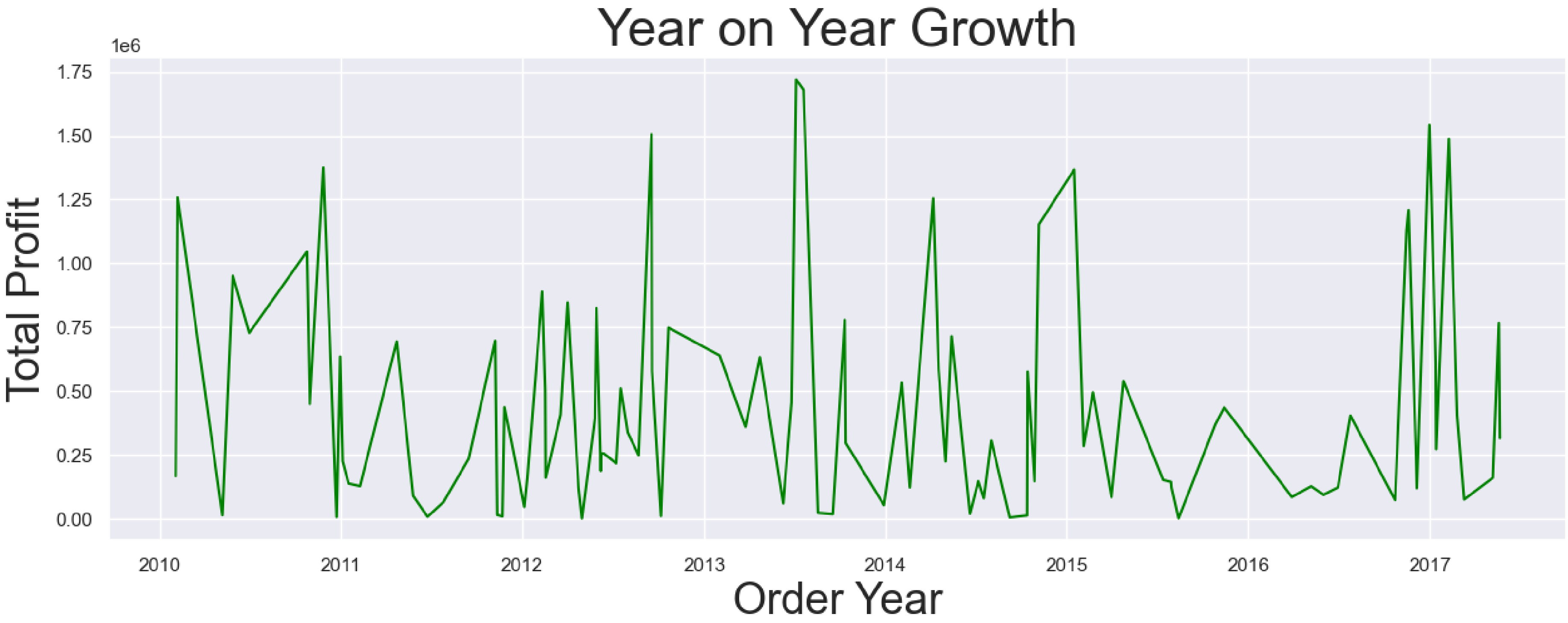


In [23]:

```
#Year on Year Growth 2010 to 2017
sns.lineplot(x='Order Date',y='Total Profit',data=df,color='Green')
sns.set(rc={'figure.figsize':(15,5)})
plt.xlabel('Order Year',fontsize=25)
plt.ylabel('Total Profit',fontsize=25)
plt.title('Year on Year Growth',fontsize=30)
```

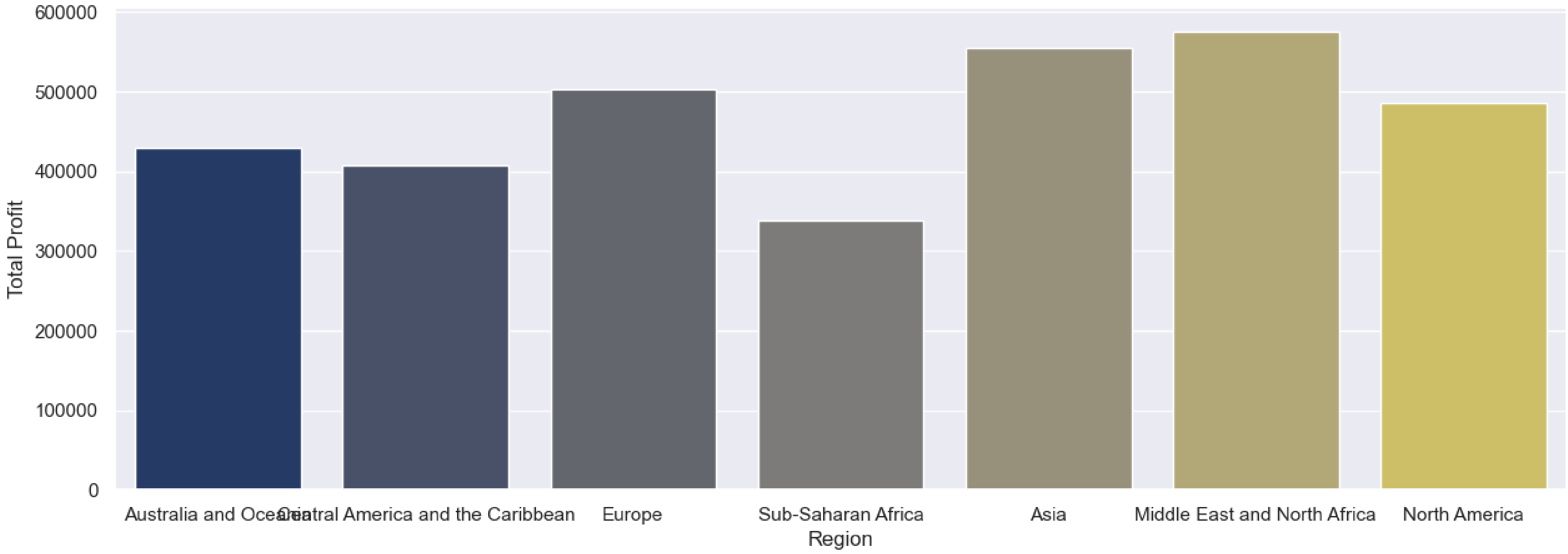
Out[23]:

```
Text(0.5, 1.0, 'Year on Year Growth')
```



```
In [24]: #Total Profit By Region
         sns.barplot(y='Total Profit',x='Region',data=df,palette='cividis',errorbar=None)
```

```
Out[24]: <Axes: xlabel='Region', ylabel='Total Profit'>
```



Deepak Mengal.

This is ending of the project, Thank you...