

# REPORTE 2 - Synergy Logistics

---

Héctor Gerardo Paredes Castillo

5 de Marzo de 2022

# ÍNDICE

1. Introducción
2. Código
3. Resultados
4. Estrategia
5. Conclusiones

# INTRODUCCIÓN

Synergy Logistics es una empresa intermediaria de importación y exportación de productos para diferentes empresas alrededor del mundo. Mediante un análisis de información de la empresa, se busca conocer lo siguiente:

- Principales rutas de importación y exportación
- Resultados de importación y exportación
- Resultados por medio de transporte utilizado
- Principales países para importación y exportación
- Resultados por año
- Principales clientes

# CÓDIGO

---

Link Github: [hectorgpcas/ANALISIS\\_o2\\_-PAREDES\\_HECTOR \(github.com\)](https://github.com/hectorgpcas/ANALISIS_o2_-PAREDES_HECTOR)

Link Colab: [ANALISIS\\_o2\\_PAREDES\\_HECTOR.ipynb - Colaboratory \(google.com\)](https://colab.research.google.com/notebooks/ANALISIS_o2_PAREDES_HECTOR.ipynb)

# Importar librerías y archivo csv

- Se importan la librería pandas para análisis de datos.
- Se importan la librería matplotlib para gráficos.
- Se importa el archivo csv de información y se convierte a DataFrame para su lectura.

```
[ ] #Se importa librería pandas
import pandas as pd

#Se importan librerías de gráficos
import matplotlib.pyplot as plt
import matplotlib as mpl

# Convertimos CSV a DataFrame
df = pd.read_csv('synergy_logistics_database.csv')
df.head()
```

	register_id	direction	origin	destination	year	date	product	transport_mode	company_name	total_value
0	1	Exports	Japan	China	2015	31/01/15	Cars	Sea	Honda	33000000
1	2	Exports	Japan	China	2015	01/02/15	Cars	Sea	Honda	16000000
2	3	Exports	Japan	China	2015	02/02/15	Cars	Sea	Honda	29000000
3	4	Exports	Japan	China	2015	03/02/15	Cars	Sea	Honda	14000000
4	5	Exports	Japan	China	2015	04/02/15	Cars	Sea	Honda	17000000

# Análisis de rutas de importación y exportación

- Del DataFrame se extraen registros de importación o exportación según el caso, mediante el uso de corchetes y se generan nuevas listas.
- Se contabilizan por origen, destino y modo de transportación con la función `size()` y `ascending=False`.
- Se ordenan las rutas de mayor a menor según su frecuencia con la función `sort_values`.
- Se filtran las primeras 10 rutas con más frecuencias con la función `.head(10)` y se imprime el resultado final de importación y exportación.

```
#Obtenemos datos de exportaciones
df_exp = df[df['direction']=='Exports']

#Se contabilizan exportaciones por rutas
exp = df_exp.groupby(['origin', 'destination', 'transport_mode']).size()

#Se convierten datos a DataFrame
exp=pd.DataFrame(exp)

#Se ordenan los datos de mayor a menor según la cantidad de exportaciones
exp=exp.sort_values(by=[0], ascending=False)

#Se renombra columna index y se convierte a columna
exp.index.name="Exportaciones"
exp.reset_index(inplace=True)

#Se renombra la nueva columna index para indicar la posición cardinal de cada ruta
exp.index.name="Lugar"

#Se renumera la columna de posición de la ruta a partir del 1
exp.index += 1

#Se renombran las columnas de origen, destino y cantidad de exportaciones
exp.columns = ['Origen', 'Destino', 'Medio', 'Exportaciones']

#Se obtienen únicamente las primeras 10 rutas con más exportaciones
exp = exp.head(10)

#Título de la tabla
print('Top 10 Rutas de Exportaciones')
print('')

#Se presenta resultado final del top 10 de exportaciones
exp
```

# Total de importaciones y exportaciones

- Se genera variable del valor total de importaciones y exportaciones por medio de las listas de importación y exportación realizados en el paso anterior.
- Se genera variable de total de importaciones y exportaciones por medio de suma de las dos variables anteriores.
- Se genera Dataframe con valores y porcentajes de importación y exportación.
- Se exponen resultados finales.

```
#Se calcula valor total de importaciones
total_imp = df_imp['total_value'].sum()

#Se calcula valor total de exportaciones
total_exp = df_exp['total_value'].sum()

#Se declara valor total de operaciones
total = total_imp + total_exp

#Se genera tabla de resultados finales de exportación e importación
resultados = pd.DataFrame(columns=['Tipo', 'Valor (Bill $)', '%'])
resultados = resultados.append({'Tipo': 'Importaciones', 'Valor (Bill $)': round(total_imp/1000000000, 2), '%':round(total_imp/total*100, 2)}, ignore_index=True)
resultados = resultados.append({'Tipo': 'Exportaciones', 'Valor (Bill $)': round(total_exp/1000000000, 2), '%':round(total_exp/total*100, 2)}, ignore_index=True)
resultados = resultados.append({'Tipo': 'Total', 'Valor (Bill $)': round(total/1000000000,2), '%':round(total/total*100, 2)}, ignore_index=True)
resultados.set_index(['Tipo'], inplace=True)
```

sultados Totales de Importación y Exportación

```
#Título
print('Resultados de Importación y Exportación')
print('')
```

resultados

Resultados de Importación y Exportación

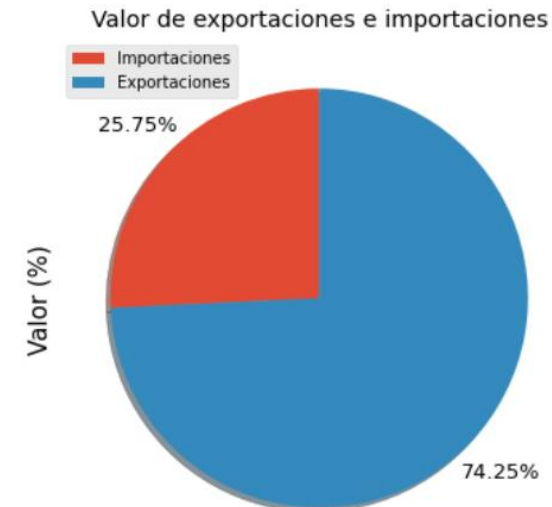
	Valor (Bill \$)	%
Tipo		
Importaciones	55.53	25.74
Exportaciones	160.16	74.26
Total	215.69	100.00

# Gráfico de importaciones y exportaciones

- Se extraen valores de importación y exportación.
- Se genera un gráfico de pie con la librería de matplotlib a partir de los valores de importación y exportación.

```
#Se obtienen los valores de importaciones y exportaciones
o = resultados[:2]

# Grafico de pastel
o['Valor (Bill $)'].plot(kind='pie', title= 'Valor de exportaciones e importaciones', autopct='%1.2f%%', figsize=(15, 6), startangle=90, shadow=True, fontsize=13, pctdi
plt.legend(labels=['Importaciones', 'Exportaciones'], loc='upper left')
plt.ylabel('Valor (%)', color='black', fontsize=15)
plt.show()
```





# Análisis de rutas de importación y exportación

- De la lista de importación y exportación, se agrupan con groupby los registros por medio de transporte, y se suman los valores de cada medio con .sum().
- Se ordenan los valores con sort\_values de mayor a menor con ascending=False.
- Se agrega una columna de porcentaje, dividiendo el valor de cada medio entre la variable total de importaciones o de exportaciones generada previamente, según el caso, se multiplica por 100 y se redondea a 2 decimales con round.
- Para facilitar la lectura de números, se convierten los valores totales de cada medio a billones, por medio de una división y redondeando a 2 decimales con round.
- Se exponen resultados finales.

```
#Se filtran los registros de importaciones por medio de transporte y se suma el valor total de los viajes según el medio de transporte
transp_imp = df_imp.groupby('transport_mode')['total_value'].sum()

#Se convierten datos a DataFrame
transp_imp = pd.DataFrame(transp_imp)

#Se ordenan los medios de transporte de mayor a menor valor total generado
transp_imp = transp_imp.sort_values(by=['total_value'], ascending=False)

#Se resetea index para colocar la posición cardinal de cada medio de transporte
transp_imp.reset_index(inplace=True)
transp_imp.index.name="Lugar"

#Se renumera la columna de posición del medio de transporte a partir del 1
transp_imp.index += 1

#Se renombran columnas de medio de transporte y valor total
transp_imp.columns = ['Medio de transporte', 'Valor Total Imp (Bill $)']

#Se obtiene porcentaje
transp_imp['%'] = round((transp_imp['Valor Total Imp (Bill $)']/total_imp)*100, 2)

#Se convierte valor del medio de transporte a billones
transp_imp['Valor Total Imp (Bill $)'] = round(transp_imp['Valor Total Imp (Bill $)']/1000000000, 2)

#Título de la tabla
print('Valor Total de Importaciones por Medio de Transporte')

#Total en billones
print(f'Total Importaciones: ${round(total_imp/1000000000, 2)} Billones')
print('')
```

Valor Total de Importaciones por Medio de Transporte  
Total Importaciones: \$55.53 Billones

	Medio de transporte	Valor Total Imp (Bill \$)	%
Lugar			
1	Sea	34.94	62.92
2	Rail	9.12	16.43
3	Road	5.99	10.79
4	Air	5.48	9.86

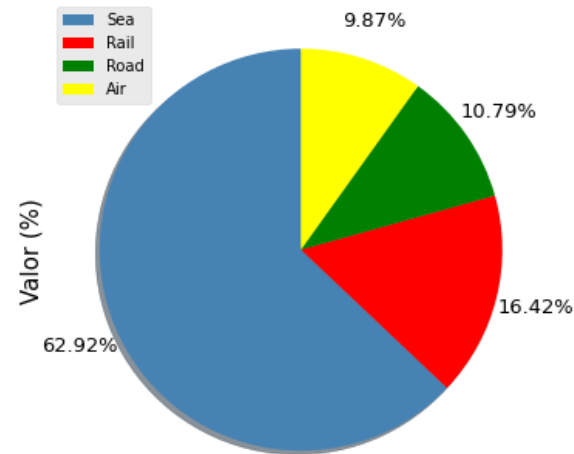
# Gráfico de medios de transporte

- Para los resultados de importación y exportación por medio de transporte, se genera un gráfico de pie con matplotlib.
- Se predeterminan previamente los colores con una lista para sea el mismo código de colores en cada gráfica.
- Se generan los resultados a partir del valor generado por cada medio de transporte.

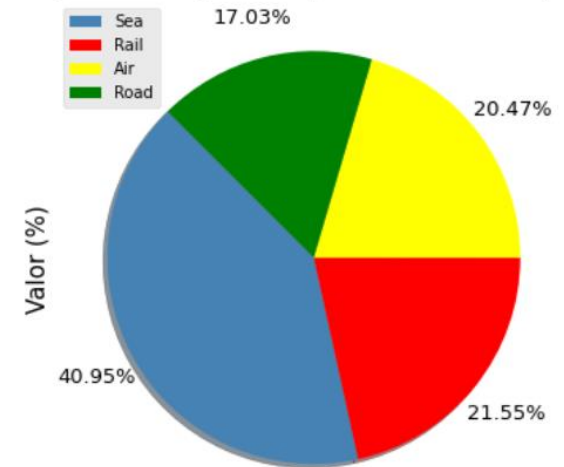
```
#Predeterminar colores
colors=['steelblue', 'red', 'yellow', 'green']

# Grafico de pastel
transp_exp['Valor Total Exp (Bill $)'].plot(kind='pie', title= 'Exportaciones generado por medios de transporte', autopct='%1.2f%%', figsize=(15, 6), startangle=135, sh
plt.legend(labels=transp_exp['Medio de transporte'], loc='upper left')
plt.ylabel('Valor (%)', color='black', fontsize=15)
plt.show()
```

Importaciones generadas por medios de transporte (%)



Exportaciones generado por medios de transporte



# Análisis de importación y exportación por país

- Tanto de la lista de importación como de exportación, se agrupan con groupby los registros por país de origen. Su suman las registros de total\_value para cada país.
- Se ordenan los resultados por país con sort\_values y ascending=False, de mayor a menor.
- Se calculan porcentajes para cada país dividiendo su valor entre la variable del total de importación y exportación respectivamente, redondeados a dos decimales.
- Por medio de la función cumsum, se suman los porcentajes acumulados por país para conocer qué países conforman el 80% con más valor generado.
- Se convierten valores de cada país a billones para facilitar su lectura, redondeados a 3 decimales.

```
#Se suman valores totales según país de destino de importación
value_imp=df_imp.groupby('origin')['total_value'].sum()

#Se conviertan datos a formato DataFrame
value_imp=pd.DataFrame(value_imp)

#Se ordenan países según sus valores de importación de mayor a menor
value_imp=value_imp.sort_values(by=['total_value'], ascending=False)

#Se resetea index para colocar la posición cardinal de cada país
value_imp.reset_index(inplace=True)
value_imp.index.name="Lugar"

#Se renumera la columna de posición del país a partir del 1
value_imp.index += 1

#Se renombran columnas de país y valor
value_imp.columns = ['País', 'Valor Imp (Bill $)']

#Se calcula columna de porcentaje del total de cada país
value_imp['% del total'] = (value_imp['Valor Imp (Bill $)']/total_imp)*100

#Se calcula porcentaje acumulado para determinar el 80% del valor
value_imp['% acum'] = value_imp['% del total'].cumsum()

#Se convierte valor total de cada país a billones
value_imp['Valor Imp (Bill $)'] = round(value_imp['Valor Imp (Bill $)']/1000000000, 3)

#Se redondea porcentaje del valor total de cada país
value_imp['% del total'] = round(value_imp['% del total'], 2)

#Se redondea porcentaje acumulado
value_imp['% acum'] = round(value_imp['% acum'], 2)

#Título
print('Valor total de Importaciones por país')

#Se imprime valor total de Importaciones (en billones)
print(f'Total Importaciones: ${round(total_imp/1000000000, 2)} Billones')
print('')

#Se presenta valores finales por país
value_imp
```

# Análisis de importación y exportación por país

- Se calcula el 80% de los países por medio de la columna de porcentaje acumulado generada en el paso anterior.
- El criterio a utilizar es contar los países con porcentaje menor a 80 y sumar uno, es decir el primero que rebase el 80% acumulado. Con ello, se genera una variable que contabiliza los países que conforman el 80% de importación y exportación respectivamente.
- Se genera una nueva lista de países que cumplen el criterio del 80%. De la lista de importación y exportación respectivamente, con el uso del corchete [:c\_países\_80\_imp] o [:c\_países\_80\_exp], se le pide que arroje los primeros países en la lista hasta llegar al primero que rebasa el 80%.
- Se imprimen los resultados de esos países y su valor en billones.

```
#Se calcula con len la cantidad de países que suman el 80% del valor total de importaciones,
#todos los menores al 80% acumulado más el primero que reúna 80% o más de acumulado para cumplir el criterio
c_países_80_imp=len(value_imp[value_imp['% acum']<80])+1

#Se establece filtro que arroje solamente los países que cumplan con el criterio del 80% acumulado
t_países_80_imp=value_imp[:c_países_80_imp]


#De los países que cumplan con el criterio del 80% acumulado, se obtiene País y Valor
países_80_imp=t_países_80_imp[['País', 'Valor Imp (Bill $)']]
```

ses que conforman el 80% de las Importaciones

```
#Título
print('Países que conforman el 80% de las Importaciones')

#Se exponen resultados
países_80_imp
```

Países que conforman el 80% de las Importaciones

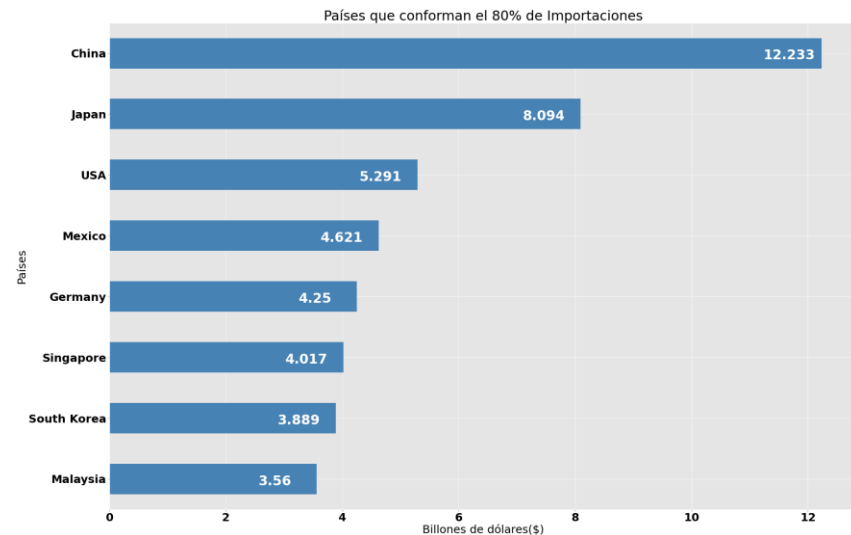
País Valor Imp (Bill \$) 		
Lugar		
1	Thailand	13.745
2	Mexico	8.894
3	United Arab Emirates	7.848
4	Japan	5.828
5	Germany	5.109
6	USA	3.758

# Gráfico de importación y exportación por país

- Se ordena la tabla del 80% de países de importación y exportación de forma invertida, dado que matplotlib lee de forma inversa a la colocación de la tabla.
- Se genera gráfico de barras horizontal.
- Se importa ggplot para obtener ordenadas en el fondo del gráfico.
- Se da formato a las etiquetas de datos.
- Con un ciclo for para cada país, se colocan las etiquetas de valores de los países en su respectiva ubicación con `plt.annotate`, con coordenadas `xy`.

```
#Obtener tabla con los países indexados y ordenados por valor de forma inversa para generar gráfico
o=países_80_imp.sort_values(by=['Valor Imp (Bill $)'])
o=o.set_index(['País'])

#Gráfico de barra horizontal
mpl.style.use('ggplot')
o.plot(kind='barh', figsize=(30,20), color='steelblue')
plt.xlabel('Billones de dólares($)', size=25, color='black')
plt.ylabel('Países', size=25, color='black')
plt.title('Países que conforman el 80% de Importaciones', size=30)
plt.xticks(fontsize=25, color='black', fontweight='bold')
plt.yticks(fontsize=25, color='black', fontweight='bold')
plt.legend([])
for i in range(0, len(o)):
    plt.annotate(o.iloc[i,0], xy=(o.iloc[i,0]-1.2, i-0.1), color='white', fontsize=30, fontweight='bold')
```



# Análisis de importación y exportación por año

- Se agrupan los registros de importación y exportación por año, y se suman los valores totales.
- Se ordenan los resultados por año del más antiguo al más reciente.
- Se convierten los resultados de cada año a billones y se redondean a 2 decimales para facilitar su lectura.
- Se imprimen resultados finales.

```
#Se suman valores totales por año de la lista de importaciones
value_year_imp = df_imp.groupby('year')['total_value'].sum()

#Se conviertan datos a formato DataFrame
value_year_imp = pd.DataFrame(value_year_imp)

#Se ordenan años del más antiguo al más reciente
value_year_imp = value_year_imp.sort_values(by=['year'], ascending=True)

#Se renombran columnas de año y valor total
value_year_imp.columns = ['Valor (Bill $)']

#Se convierte valor total de cada año a billones
value_year_imp['Valor (Bill $)'] = round(value_year_imp['Valor (Bill $)']/1000000000, 2)
```

```
#Título
print('Importaciones por año')
print('')

#Se exponen resultados de importaciones por año
value_year_imp
```

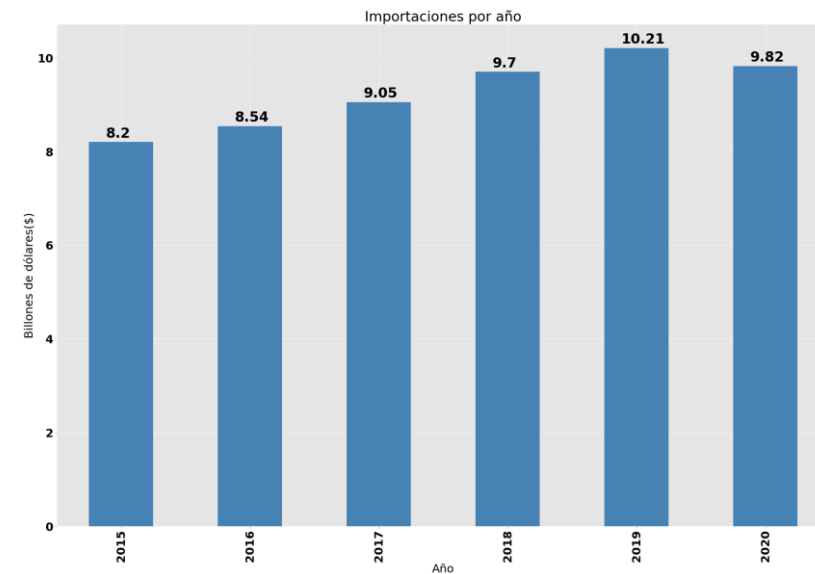
Importaciones por año

	Año	Valor (Bill \$)
Lugar		
1	2015	8.20
2	2016	8.54
3	2017	9.05
4	2018	9.70
5	2019	10.21
6	2020	9.82

# Gráfico de importación y exportación por año

- A partir de la tabla de importaciones y exportaciones por año, se genera una gráfica de barras vertical que muestra los resultados por año para cada rubro.
- Se importa ggplot para obtener ordenadas en el fondo del gráfico.
- Se da formato a las etiquetas de datos.
- Con un ciclo for para cada año, se colocan las etiquetas de valores de cada año en su respectiva ubicación con `plt.annotate`, con coordenadas `xy`.

```
#Gráfico de barra horizontal
mpl.style.use('ggplot')
value_year_imp['Valor (Bill $)'].plot(kind='bar', figsize=(30,20), color='steelblue')
plt.xlabel('Año', size=25, color='black')
plt.ylabel('Billones de dólares($)', size=25, color='black')
plt.title('Importaciones por año', size=30)
plt.xticks(fontsize=25, color='black', fontweight="bold")
plt.yticks(fontsize=25, color='black', fontweight="bold")
plt.legend([])
for i in range(0, len(value_year_imp)):
    plt.annotate(value_year_imp.iloc[i,0], xy=(i-0.12, value_year_imp.iloc[i,0]+0.1), color='black', fontsize=30, fontweight="bold")
plt.show()
```



# Análisis por compañía cliente

- Con una lógica similar a la utilizada para analizar los resultados por país, se realiza el análisis de resultados por cliente, a partir de la columna company\_name.
- En este caso, se contemplan resultados en general, sin distinguir si son importación y exportación. Entonces, los datos se extraen del df original generado a partir del csv proporcionado.
- Finalmente, se filtran los primeros 10 clientes con mayor valor total.

```
#Se suman valores totales por cliente de la lista general
value_comp = df.groupby('company_name')['total_value'].sum()

#Se convierten datos a formato DataFrame
value_comp = pd.DataFrame(value_comp)

#Se ordenan clientes según sus valores de importación de mayor a menor
value_comp = value_comp.sort_values(by=['total_value'], ascending=False)

#Se resetea index para colocar la posición cardinal de cada cliente
value_comp.reset_index(inplace=True)
value_comp.index.name="Lugar"

#Se renumera la columna de posición del cliente a partir del 1
value_comp.index += 1

#Se renombran columnas de cliente y valor
value_comp.columns = ['Compañía', 'Valor (Bill $)']

#Se calcula columna de porcentaje del total de cada cliente
value_comp['% del total'] = (value_comp['Valor (Bill $)']/total)*100

#Se calcula porcentaje acumulado para determinar el 80% del valor
value_comp['% acum'] = value_comp['% del total'].cumsum()

#Se convierte valor total de cada cliente a billones
value_comp['Valor (Bill $)'] = round(value_comp['Valor (Bill $)']/1000000000, 3)

#Se redondea porcentaje del valor total de cada cliente
value_comp['% del total'] = round(value_comp['% del total'], 2)

#Se redondea porcentaje acumulado
value_comp['% acum'] = round(value_comp['% acum'], 2)

#Se filtran los diez principales clientes
top_10_comp = value_comp.head(10)

#Total ventas top 10
total_top_10 = top_10_comp['Valor (Bill $)'].sum()
```

```
#Título
print('Top 10 Clientes')
#Se calcula total de clientes de la compañía
print(f"Total ventas top 10: ${total_top_10} Billones")
print(f"Cantidad de clientes: {len(value_comp)}")
print('')
```

```
#Se presenta valores finales por cliente (en billones)
top_10_comp
```

Top 10 Clientes  
Total ventas top 10: \$90.977 Billones  
Cantidad de clientes: 77

	Compañía	Valor (Bill \$)	% del total	% acum
Lugar				
1	Lenovo Group Ltd	16.203	7.51	7.51
2	Boeing Company	10.215	4.74	12.25
3	Cubist Pharma F	10.163	4.71	16.96
4	PAO Gazprom	8.731	4.05	21.01
5	Honda	8.528	3.95	24.96
6	LG Electronics Inc	8.203	3.80	28.76
7	Technology & Innovation Co	7.848	3.64	32.40
8	Xiaomi Corp	7.666	3.55	35.96
9	Caterpillar Inc	7.282	3.38	39.33
10	L Oreal Co	6.138	2.85	42.18

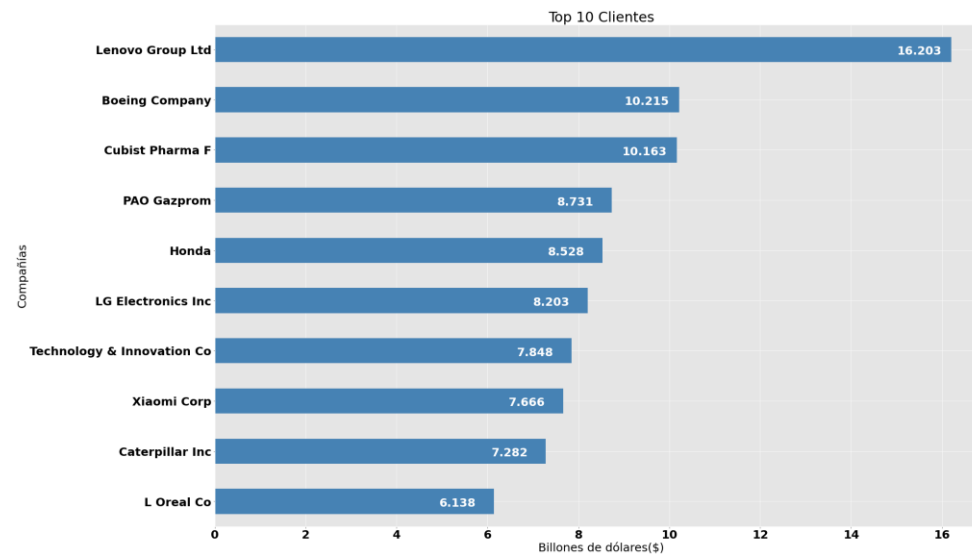


# Gráfico de importación y exportación por cliente

- A partir de la tabla de resultados del top 10 de clientes, se genera una gráfica de barras horizontal que muestra los resultados de cada uno.
- Se importa ggplot para obtener ordenadas en el fondo del gráfico.
- Se da formato a las etiquetas de datos.
- Con un ciclo for para cada cliente, se colocan las etiquetas de valores de cada cliente en su respectiva ubicación con plt.annotate, con coordenadas xy.

```
#Obtener tabla con los clientes indexados y ordenados por valor de forma inversa para generar gráfico
o=top_10_comp.sort_values(by=['Valor (Bill $)'])
o=o.set_index(['Compañía'])

#Gráfico de barra horizontal
mpl.style.use('ggplot')
o['Valor (Bill $)'].plot(kind='barh', figsize=(30,20), color='steelblue')
plt.xlabel('Billones de dólares($)', size=25, color='black')
plt.ylabel('Compañías', size=25, color='black')
plt.title('Top 10 Clientes', size=30)
plt.xticks(fontsize=25, color='black', fontweight='bold')
plt.yticks(fontsize=25, color='black', fontweight='bold')
plt.legend([])
for i in range(0, len(o)):
    plt.annotate(o.iloc[i,0], xy=(o.iloc[i,0]-1.2, i-0.1), color='white', fontsize=25, fontweight='bold')
plt.show()
```



# RESULTADOS

---

# Top 10 Rutas de Importaciones

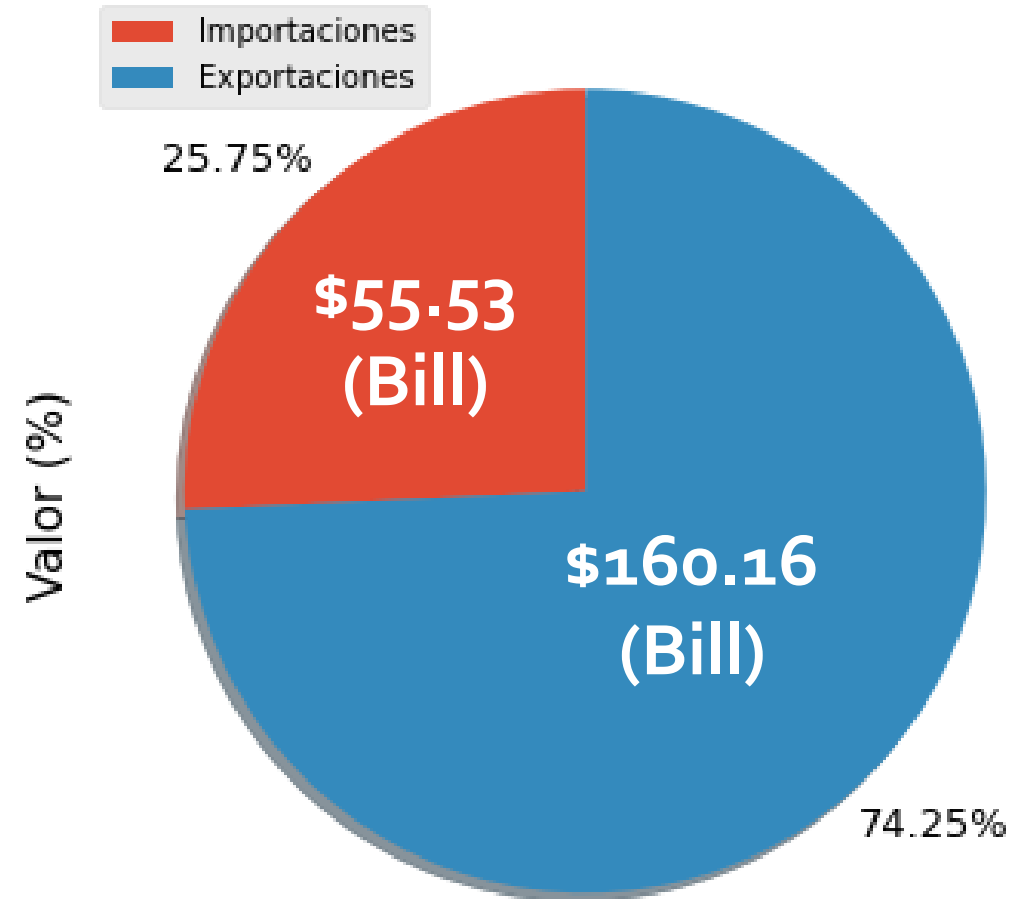
	Origen	Destino	Medio	Importaciones
Lugar				
1	Singapore	Thailand	Sea	273
2	Germany	China	Sea	233
3	China	Japan	Air	210
4	Japan	Mexico	Sea	206
5	Malaysia	Thailand	Rail	195
6	China	Thailand	Road	145
7	Spain	Germany	Road	142
8	China	United Arab Emirates	Sea	114
9	Brazil	China	Sea	113
10	USA	Thailand	Sea	109

# Top 10 Rutas de Exportaciones

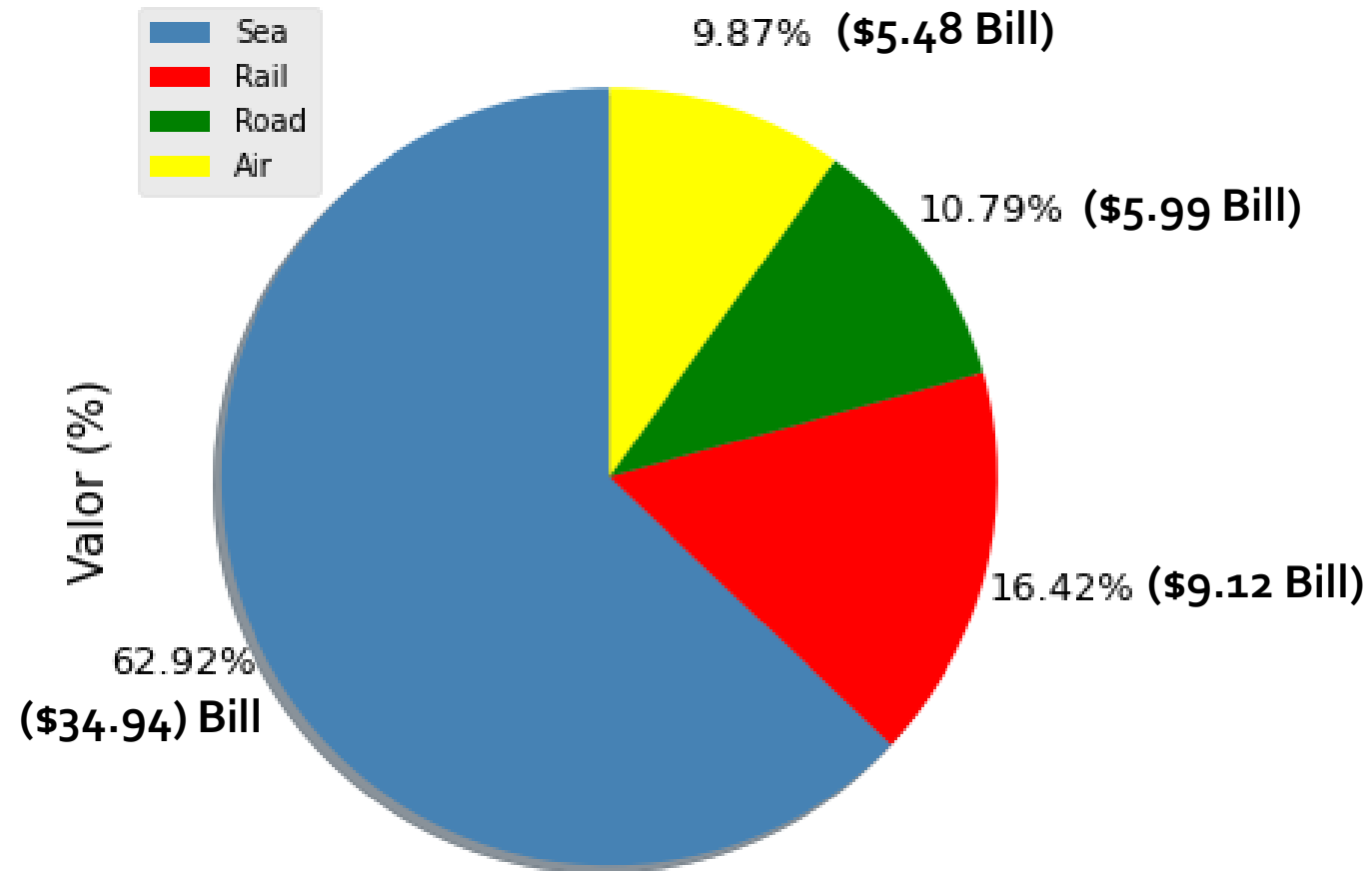
	Origen	Destino	Medio	Exportaciones
Lugar				
1	South Korea	Vietnam	Sea	497
2	USA	Netherlands	Sea	436
3	Netherlands	Belgium	Road	374
4	China	Mexico	Air	330
5	Japan	Brazil	Sea	306
6	Germany	France	Road	299
7	South Korea	Japan	Sea	279
8	Australia	Singapore	Sea	273
9	Canada	Mexico	Rail	261
10	China	Spain	Air	250

# Valor de exportaciones e importaciones

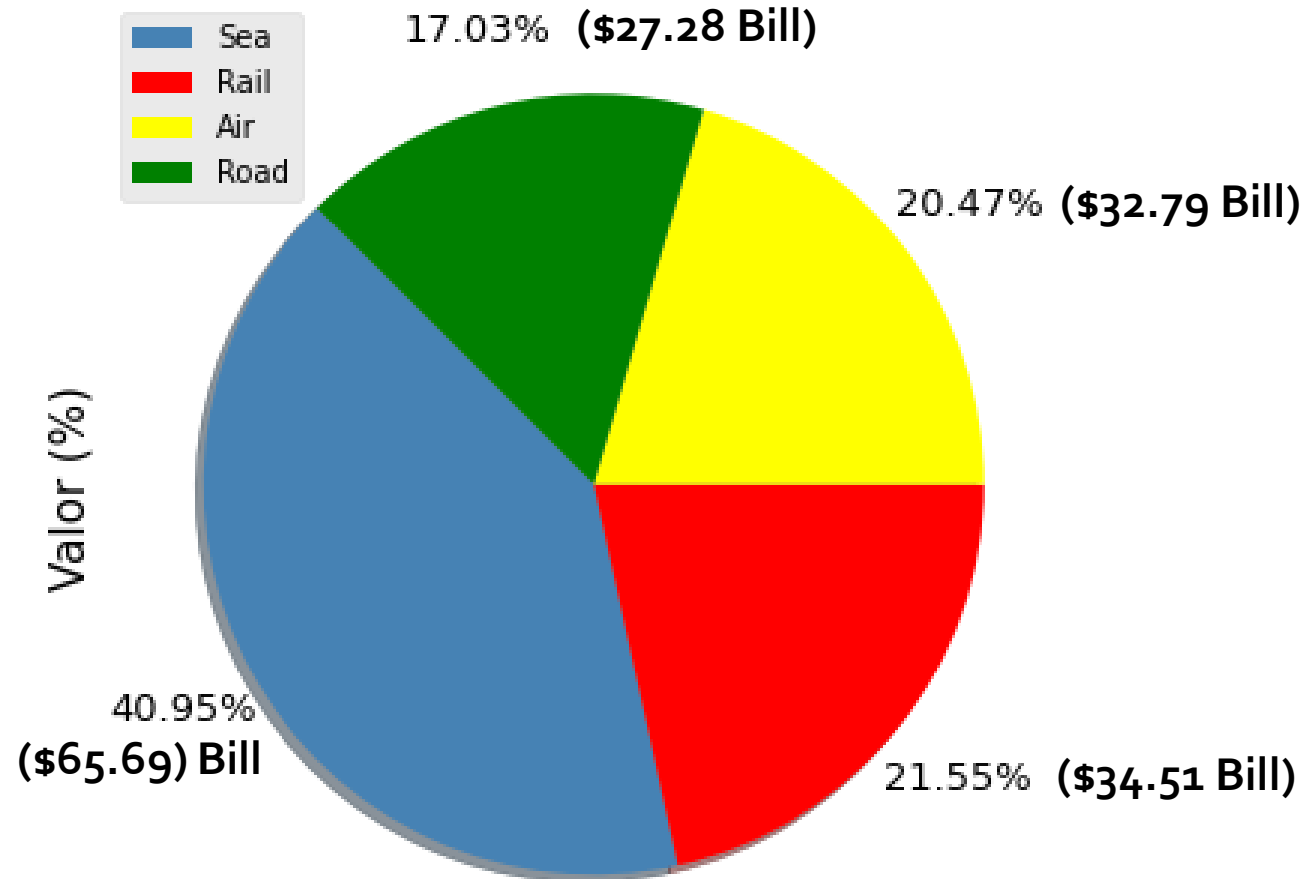
**\$215.69**  
Billones



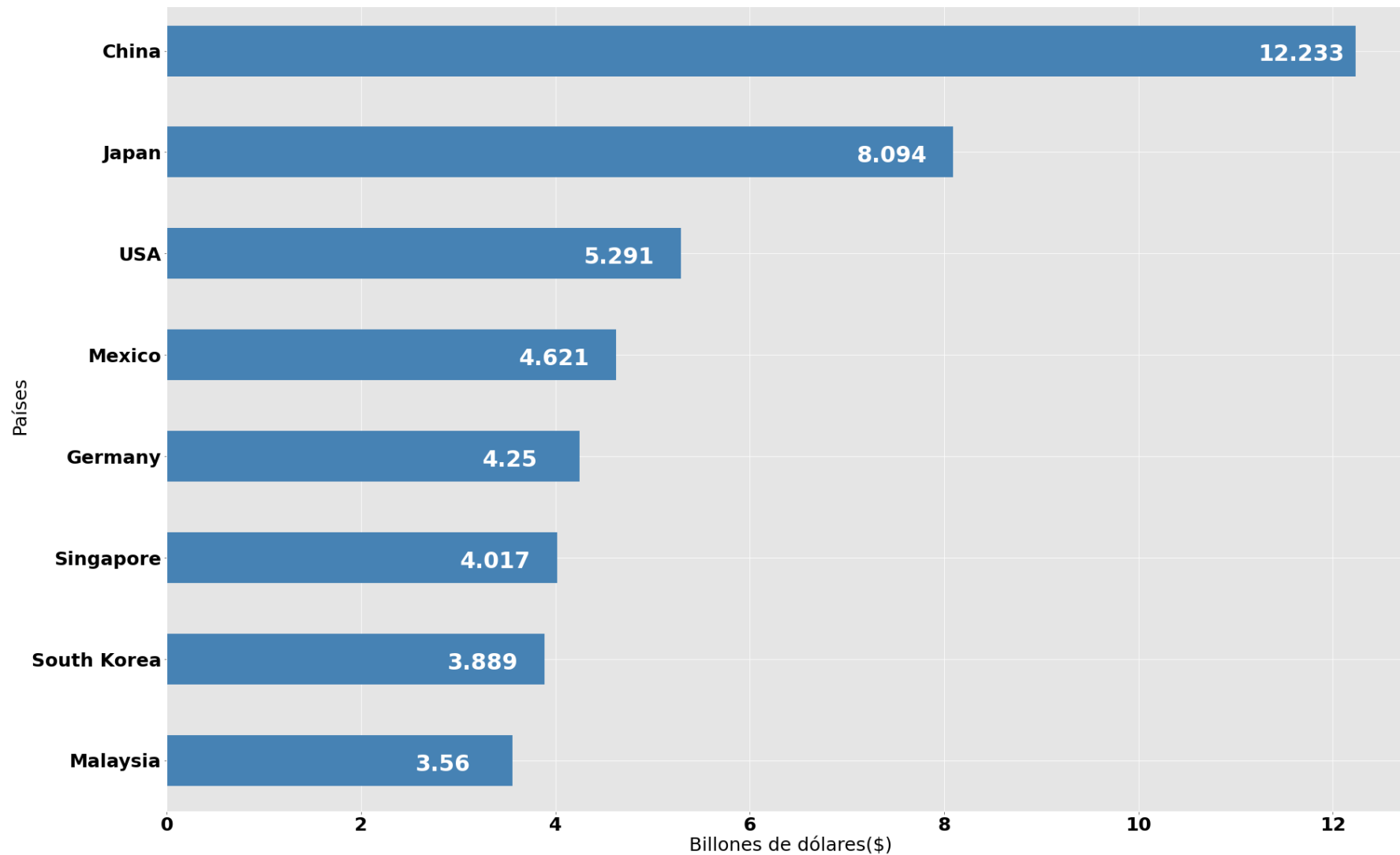
# Importaciones por Medio de Transporte



# Exportaciones por Medio de Transporte

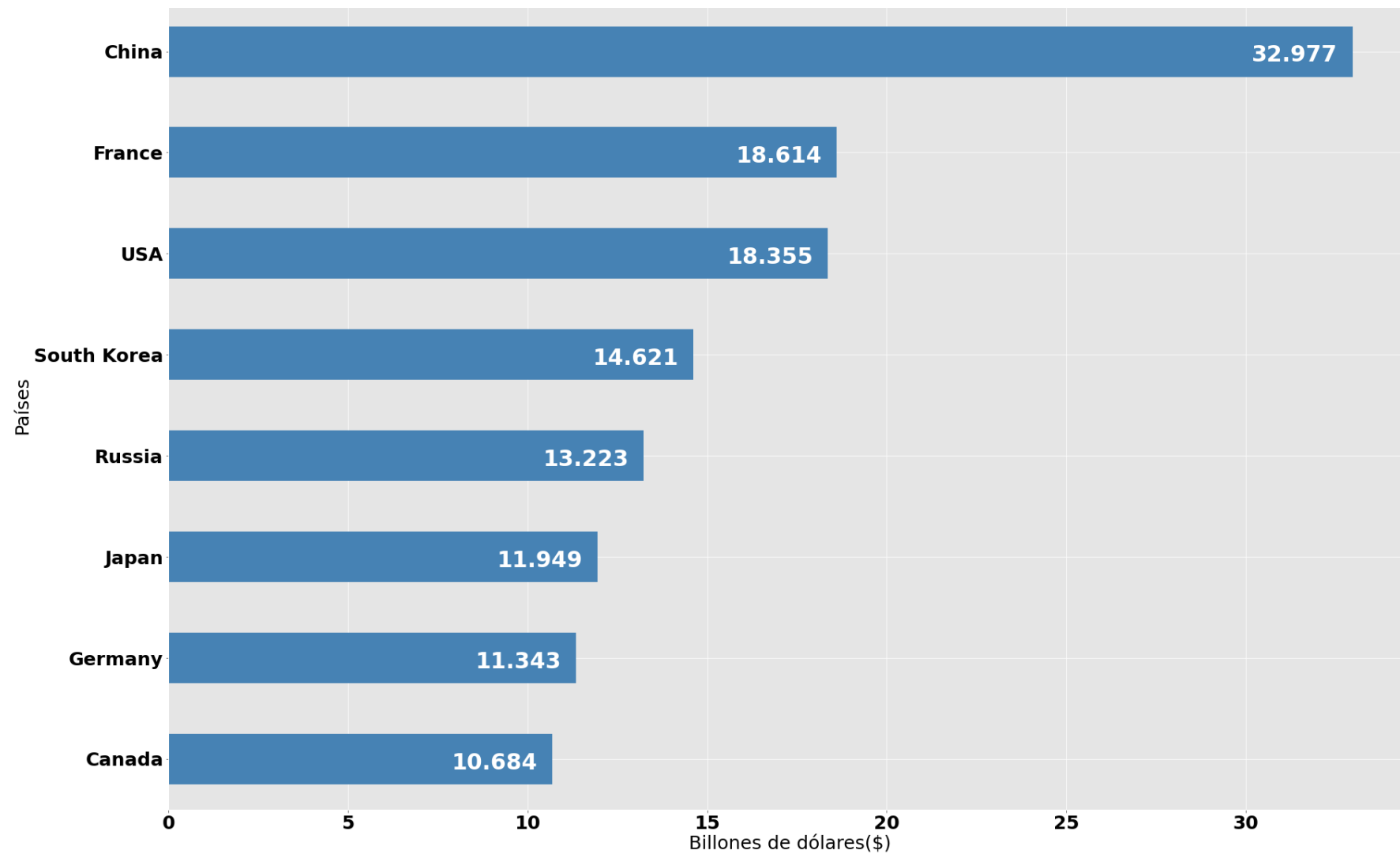


# Países que conforman el 80% de Importaciones

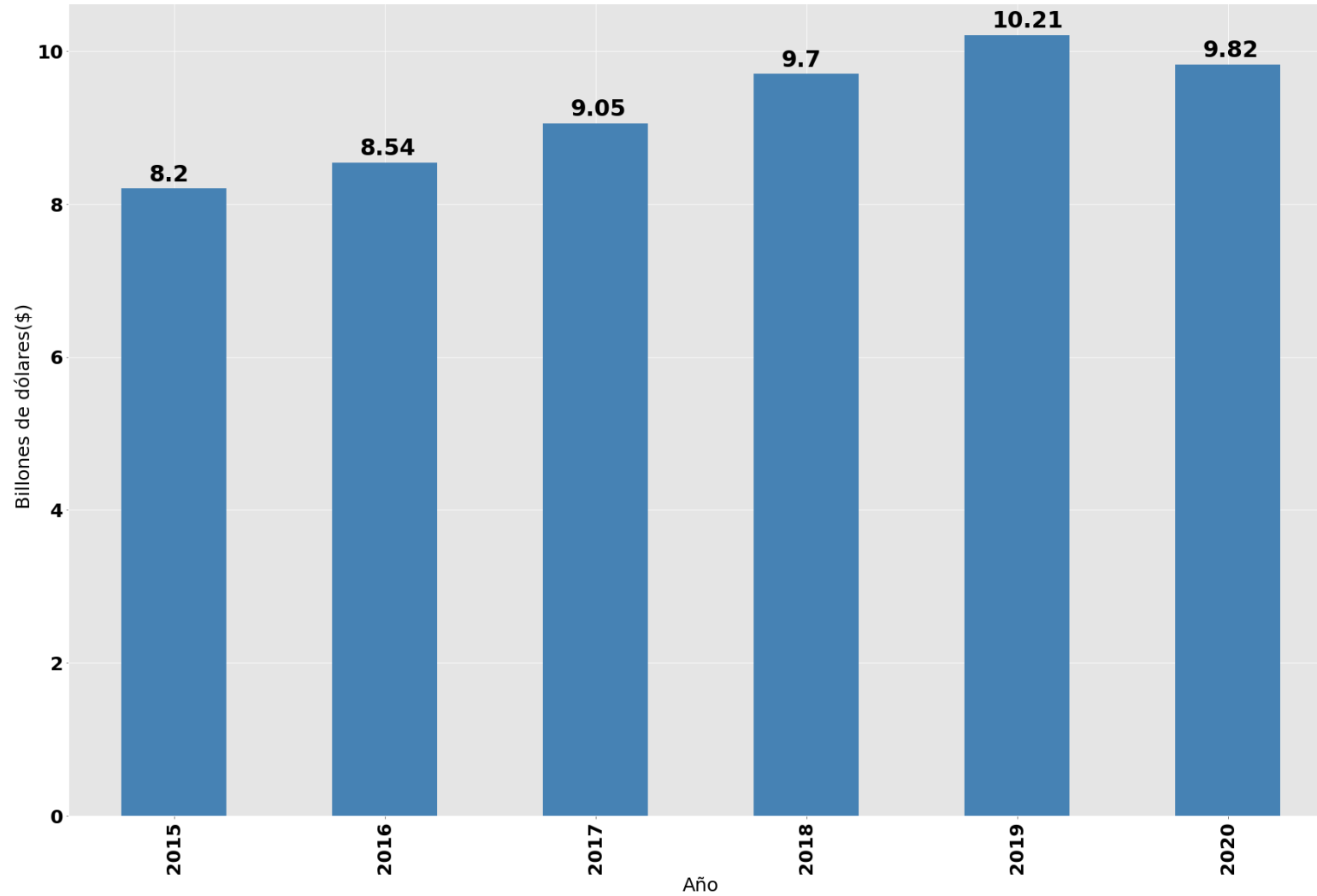




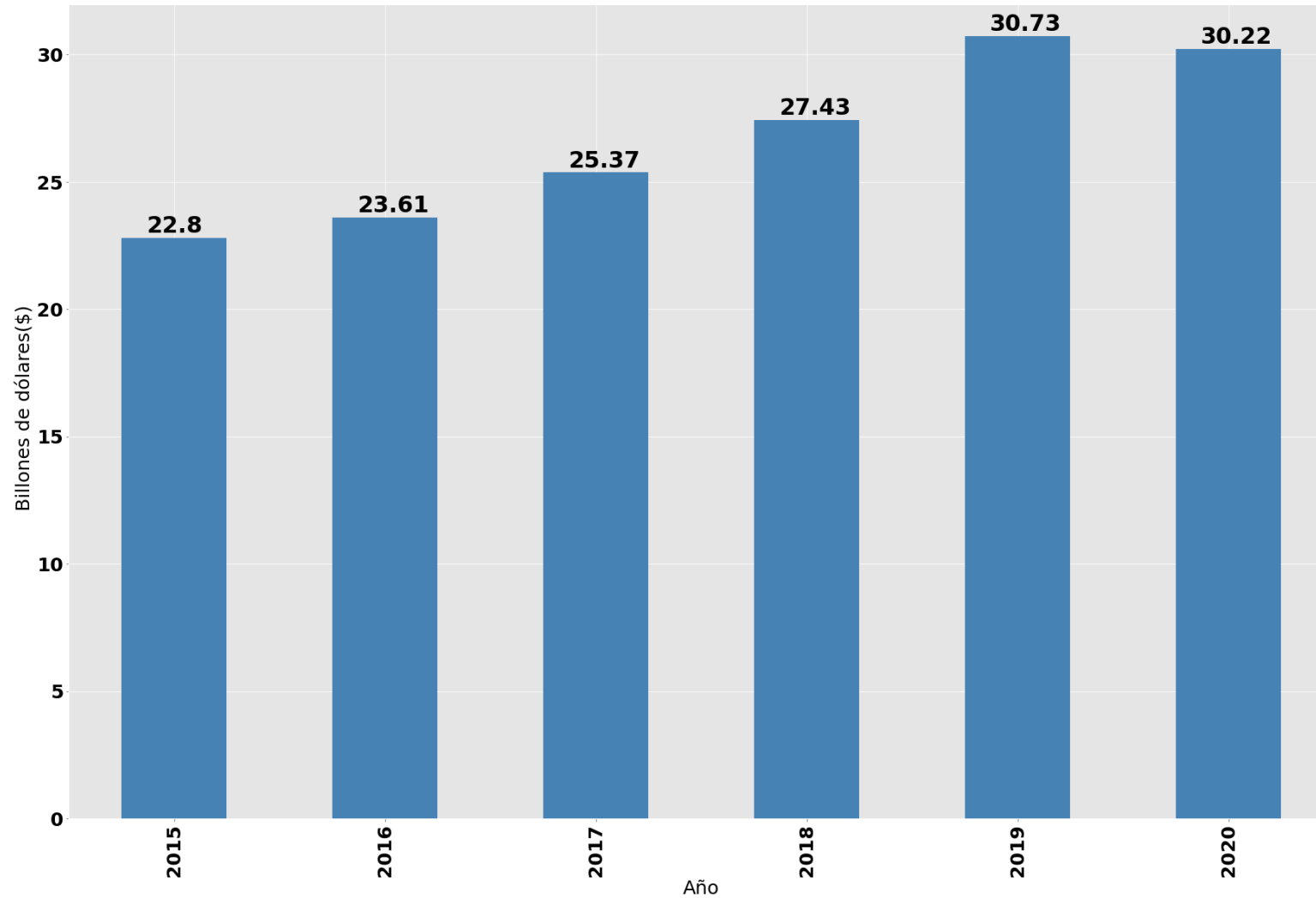
# Países que conforman el 80% de Exportaciones



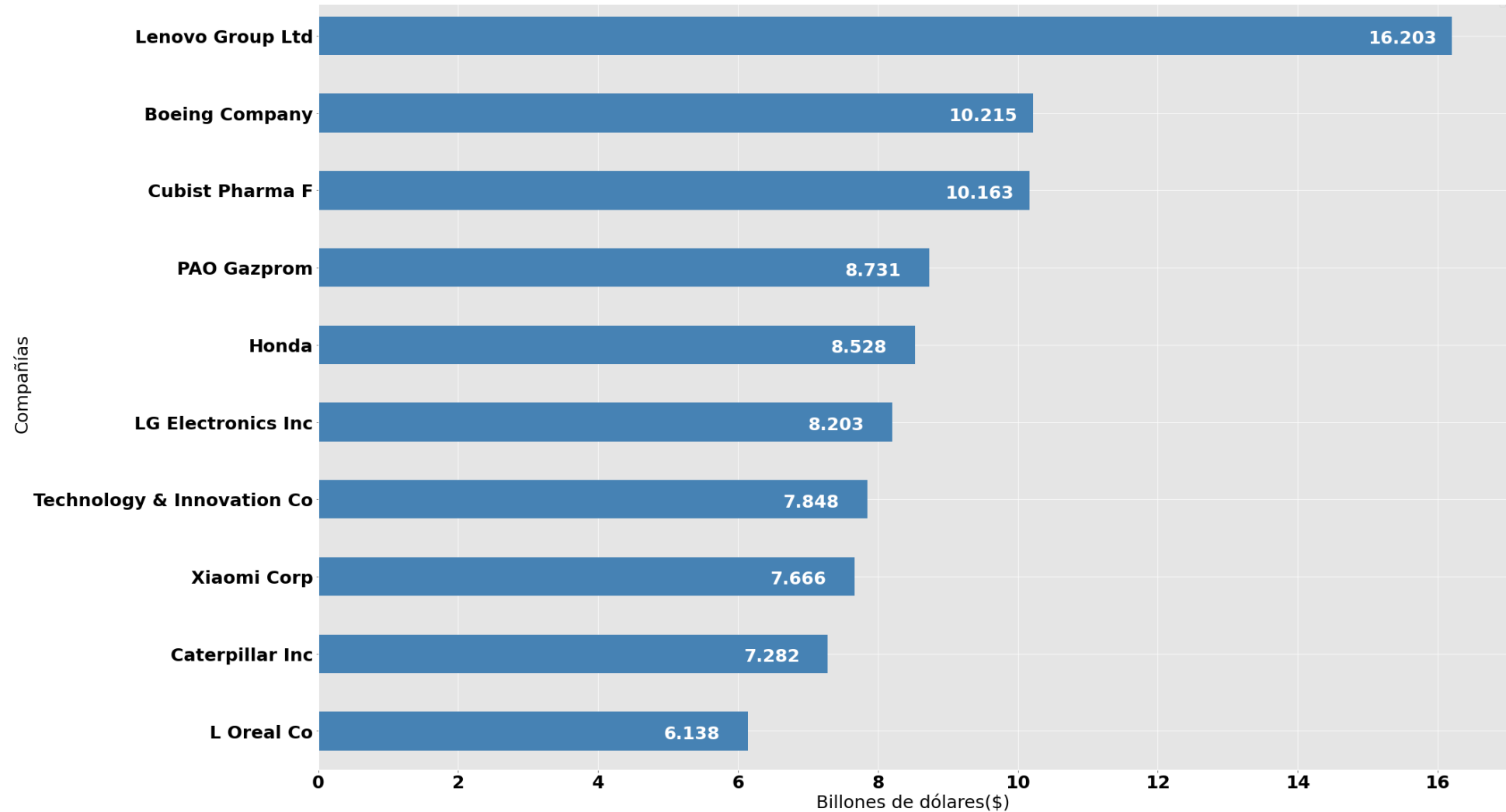
# Importaciones por año



# Exportaciones por año



# Top 10 Clientes

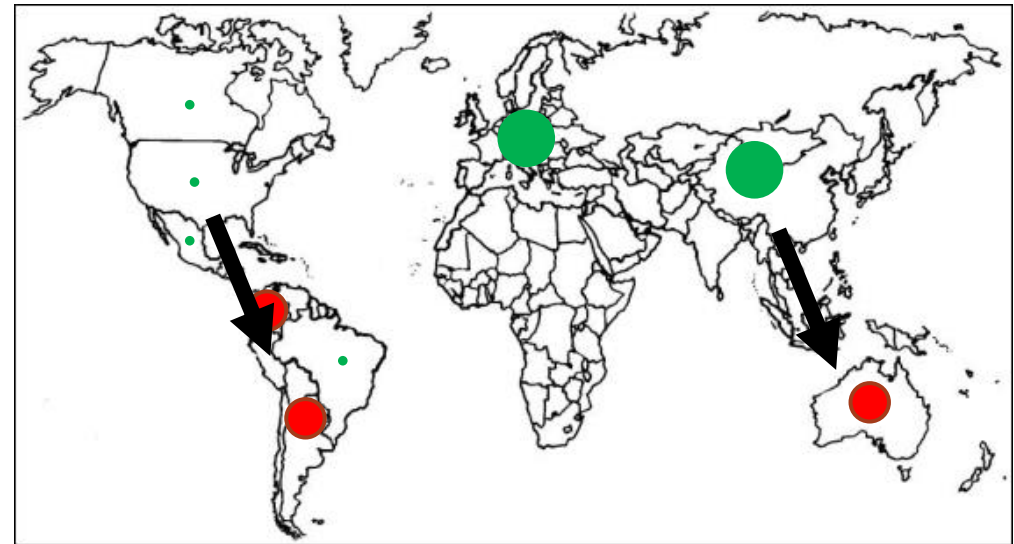


# ESTRATEGIA

---

# Estrategia por países

- Synergy tiene la oportunidad de explotar los mercados de América Latina y Oceanía (Puntos Rojos).
- Puede llegar a América Latina a través de Norte América.
- Puede llegar a Oceanía a través de Asia.



# Estrategia de medios de transporte

- La empresa debe minimizar los servicios de transporte terrestre:
  - Generan pocos beneficios a la compañía.
  - Las distancias entre los principales países de operación son largas.
  - Solo deben utilizarlo en distancias cortas, dentro del mismo continente.



# Estrategia de importación

- Synergy puede brindar más facilidades para importación de productos y elevar sus ingresos en ese rubro.
  - Brindar asesoría a clientes nuevos que deseen importar productos.
  - Reducir precios u ofrecer promociones según el volumen de importación.





# CONCLUSIONES

- La actividad que genera mayores ingresos a Synergy es la exportación de productos.
- Los principales medios de transporte para realizar sus operaciones son el barco y el ferrocarril.
- Las zonas donde más se demandan los servicios de la empresa son Norte América, Asia y Europa.
- Los ingresos de la compañía tanto en importación como exportación han crecido constantemente de 2015 a 2019, con una pequeña disminución en 2020. Posiblemente como un efecto de la pandemia de COVID-19.
- Los principales clientes de la compañía figuran en el ramo electrónico, automotriz e industrial.