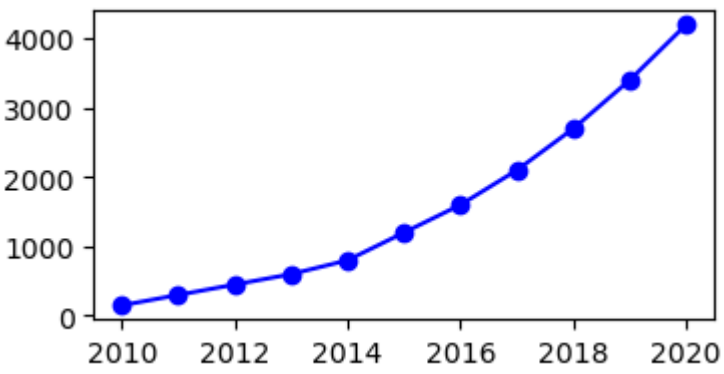
1. The following table shows the number of Data Science job postings recorded over the years 2010 to 2020. Year Job Postings 2010 150 2011 300 2012 450 2013 600 2014 800 2015 1200 2016 1600 2017 2100 2018 2700 2019 3400 2020 4200 Using Python (Pandas and Matplotlib): Create a DataFrame for the given data

In [4]:
```
import pandas as pd
data={
    'Year':[2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020],
'Job Postings': [150, 300, 450, 600, 800, 1200, 1600, 2100, 2700, 3400, 4200]
}
df=pd.DataFrame(data)
print(df)
```

```
    Year  Job Postings
0   2010           150
1   2011           300
2   2012           450
3   2013           600
4   2014           800
5   2015          1200
6   2016          1600
7   2017          2100
8   2018          2700
9   2019          3400
10  2020          4200
```

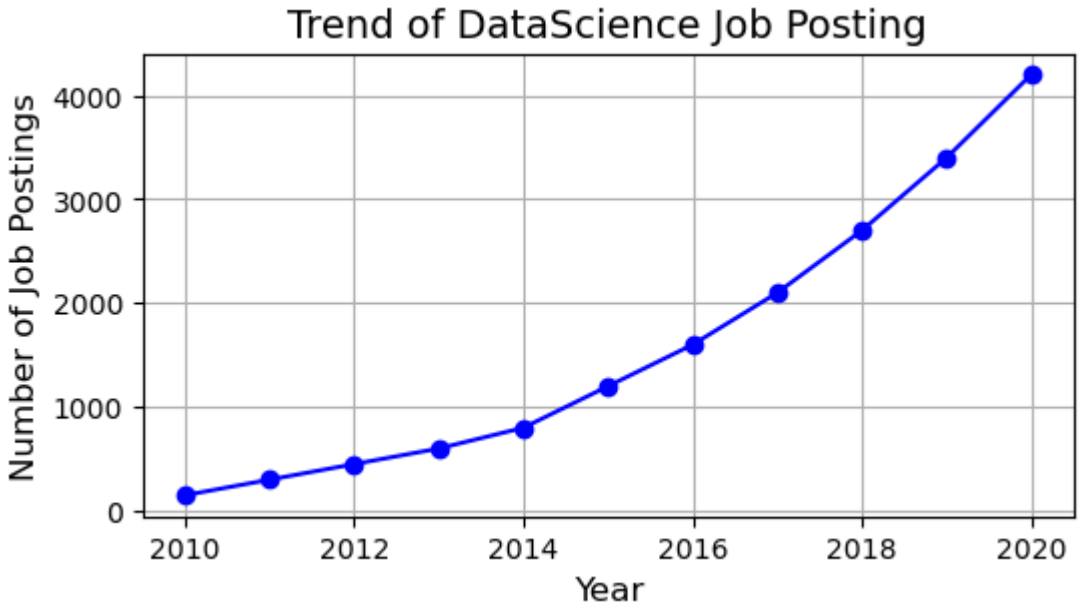
In [ ]:
```
2. Plot a line graph showing the trend of Data Science job postings over the yea
on data points.
```

In [3]:
```
import pandas as pd
import matplotlib.pyplot as plt
data={
    'Year':[2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020],
'Job Postings': [150, 300, 450, 600, 800, 1200, 1600, 2100, 2700, 3400, 4200]
}
df=pd.DataFrame(data)
plt.figure(figsize=(4,2))
plt.plot(df['Year'],df['Job Postings'],marker='o',linestyle='-',color='b')
plt.show()
```



In [ ]:
```
3. Add suitable title and axis labels to the graph.
Expected Output: A line chart showing steady growth of job postings from 2010 to
```

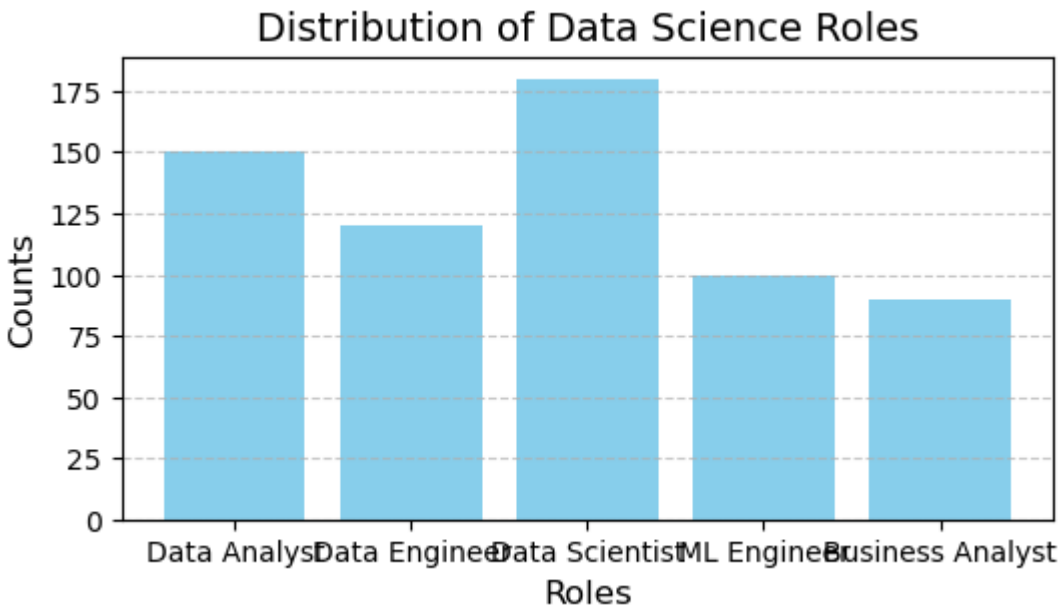
In [8]:
```
import pandas as pd
import matplotlib.pyplot as plt
data={
    'Year':[2010, 2011, 2012, 2013, 2014, 2015, 2016, 2017, 2018, 2019, 2020],
'Job Postings': [150, 300, 450, 600, 800, 1200, 1600, 2100, 2700, 3400, 4200]
}
df=pd.DataFrame(data)
plt.figure(figsize=(6,3))
plt.plot(df['Year'],df['Job Postings'],marker='o',linestyle='-',color='b')
plt.title('Trend of DataScience Job Posting',fontsize=14)
plt.xlabel('Year',fontsize=12)
plt.ylabel('Number of Job Postings',fontsize=12)
plt.grid(True)
plt.show()
```



4.Write a Python program using Matplotlib to create a bar chart that shows the distribution of different Data Science roles (Data Analyst, Data Engineer, Data Scientist, ML Engineer, and

Business Analyst) with their respective counts. Add appropriate axis labels and a title to the chart."

In [8]:
```
import pandas as pd
import matplotlib.pyplot as plt
data={
    'roles' : ['Data Analyst', 'Data Engineer', 'Data Scientist', 'ML Engineer',
    'counts' : [150, 120, 180, 100, 90]
}
df=pd.DataFrame(data)
plt.figure(figsize=(6,3))
plt.bar(df['roles'],df['counts'],color='skyblue')
plt.title('Distribution of Data Science Roles',fontsize=14)
plt.xlabel('Roles',fontsize=12)
plt.ylabel('Counts',fontsize=12)
plt.grid(axis='y', linestyle='--', alpha=0.7)
plt.show()
```



4."Write a Python program to demonstrate the three main types of data: Structured, Unstructured, and Semi-structured. Use a Pandas DataFrame for structured data, a text string for unstructured data, and a dictionary for semi-structured data. Print each type of data with clear labels."

In [14]:
```
import pandas as pd
data ={
    'name':['kaushika','kaviya','manisha'],
    'age':[19,17,18],
    'city':['kpm','chennai','chennai']
}
df=pd.DataFrame(data)
print("___structured_data___")
print(df,"\n")
unstructured_data="we three are studying in rajalakshmi college"
print("___unstructured_data___")
print(unstructured_data,"\n")
semi_structured_data = {
    "student": {
        "id": 101,
        "name": "John Doe",
        "courses": ["Math", "Science", "History"],
        "address": {
            "city": "New York",
            "zipcode": "10001"
        }
    }
}
print("___semi_structured_data___")
print(semi_structured_data)
```

```
___structured_data___
       name  age     city
0  kaushika   19      kpm
1    kaviya   17  chennai
2   manisha   18  chennai

___unstructured_data___
we three are studying in rajalakshmi college

___semi_structured_data___
{'student': {'id': 101, 'name': 'John Doe', 'courses': ['Math', 'Science', 'Histo
ry'], 'address': {'city': 'New York', 'zipcode': '10001'}}}
```

5."Write a Python program using the cryptography.fernet module to demonstrate symmetric key encryption and decryption. Encrypt the text 'Rajalakshmi Engineering College' using a generated key, display the encrypted ciphertext, and then decrypt it back to the original text. Print the original, encrypted, and decrypted data."

In [16]:
```python
from cryptography.fernet import Fernet
key = Fernet.generate_key()
f = Fernet(key)
token = f.encrypt(b"Rajalakshmi Engineering College")
token
f.decrypt(token)
key = Fernet.generate_key()
cipher_suite = Fernet(key)
plain_text = b"Rajalakshmi Engineering College."
cipher_text = cipher_suite.encrypt(plain_text)
decrypted_text = cipher_suite.decrypt(cipher_text)
print("Original Data:", plain_text)
print("Encrypted Data:", cipher_text)
print("Decrypted Data:", decrypted_text)
```

```
Original Data: b'Rajalakshmi Engineering College.'
Encrypted Data: b'gAAAAABotqMVLSHIvEqKOCsKwcTrBobp3mODH6UcPgNqfIfN7gbF0a05Lwfum5Z
UW3368Lmi4Be5E0c1NcA-fePsP-FKX6qbz64pFdx6gHnOqrwERwUaDW6FNYgcf8M1v8qUM0XiqOEp'
Decrypted Data: b'Rajalakshmi Engineering College.'
```

In [ ]:
```
```