

## 第一次作品：Python 函數的繪製的觀念與技巧¶

本作品藉著繪製 11 個函數（如網站講義所列），學習 Python 的繪圖觀念與指令的運用，並透過函數繪製的過程，更深入了解每個函數的特色與精彩之處，譬如，函數的極值位置（local extrema and (or) global extrema）、函數是否通過  $y=0$ （即  $f(x)=0$  是否有實數根）、是否有漸進線？函數的範圍... 等。

以下針對每個函數，除程式碼與結果之外，另加註「注意事項與討論」，補充對函數的分析與程式技巧的探討，做為未來 Python 程式寫作的參考與資源庫。

備註：對每一份作品，同學可以依據自己的時間與興趣，將作品表達得更豐富。譬如，可以再找一些有趣或有難度的函數試試看。但是，切記不要從其他地方直接抄襲過來。

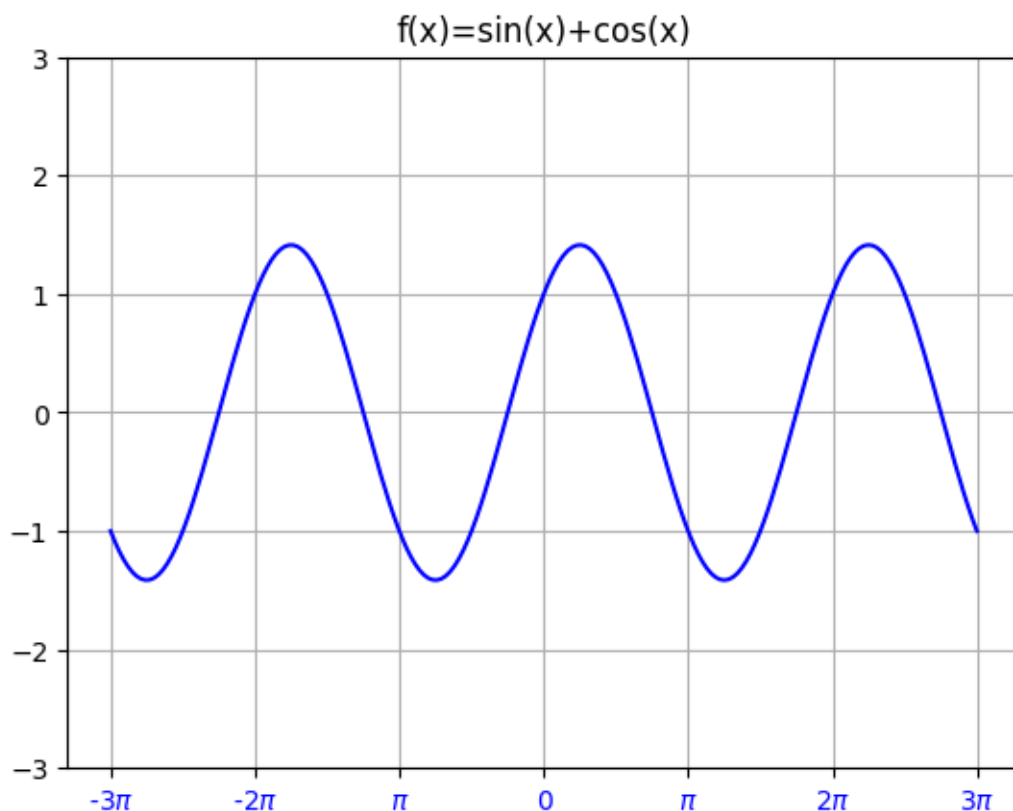
### 1. 繪製函數¶

$f(x) = \sin(x) + \cos(x)$

注意事項與討論：

- 畫出下列函數，X 軸範圍自訂，盡量畫出最完整的函數圖，能呈現隱藏在函數的資訊。
- 將上述範例介紹的繪圖技巧充分地應用，讓自己熟悉各種繪圖的表現，包括文字、刻度、顏色、legend、grid、座標軸的位置... 等。不是只將函數畫出來而已，還要畫得好。有些函數需要的指令不再前面的範例，請自行上網查詢手冊。

```
In [ ]:
import numpy as np
import matplotlib.pyplot as plt
x = np.linspace(-3*np.pi, 3*np.pi, 200)
y = np.sin(x) + np.cos(x)
fig, ax = plt.subplots(1)
ax.plot(x, y, color='b')
ax.set_xticks(np.array([-3, -2, -1, 0, 1, 2, 3])*np.pi)
ax.set_xticklabels(['-3 $\pi$ ', '-2 $\pi$ ', ' $\pi$ ', '0', ' $\pi$ ', '2 $\pi$ ', '3 $\pi$ '], \
                    fontsize=10, color = 'b')
ax.set_title('f(x)=sin(x)+cos(x)')
ax.set_ylim([-3, 3])
ax.grid(True)
plt.show()
```



## 2. 繪製函數¶

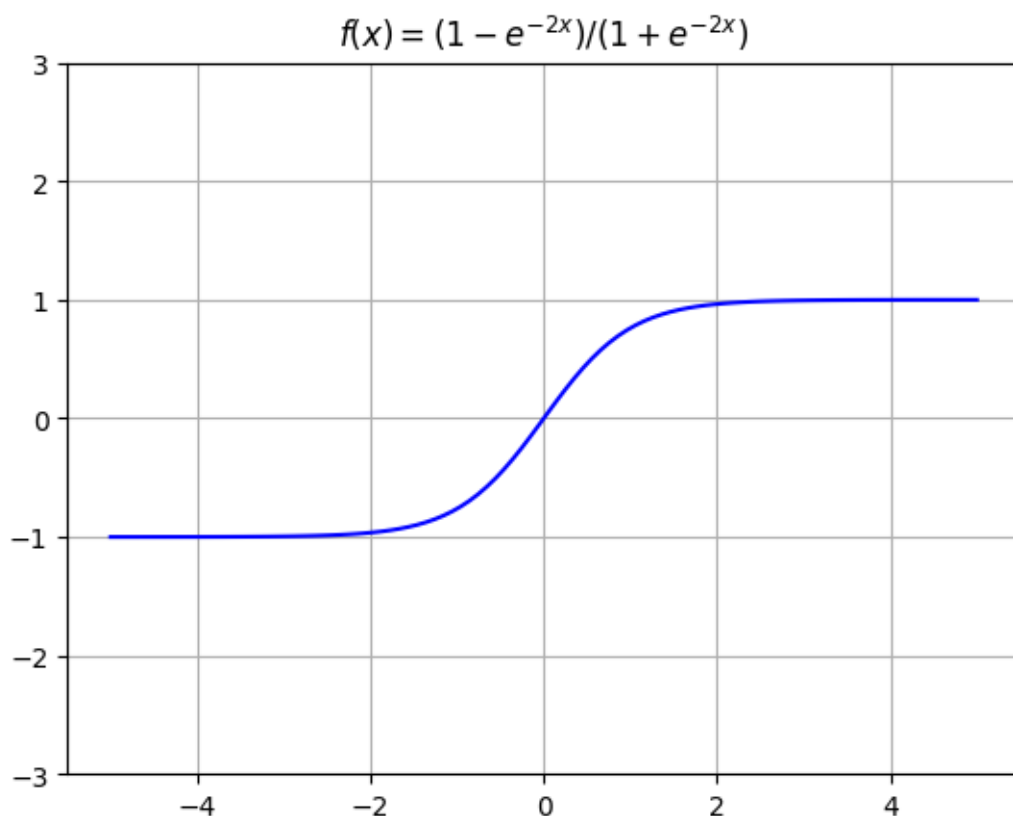
$$f(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}$$

注意事項與討論：

- 畫出下列函數，X 軸範圍自訂，盡量畫出最完整的函數圖，能呈現隱藏在函數的資訊。
- 將上述範例介紹的繪圖技巧充分地應用，讓自己熟悉各種繪圖的表現，包括文字、刻度、顏色、legend、grid、座標軸的位置... 等。不是只將函數畫出來而已，還要畫得好。有些函數需要的指令不再前面的範例，請自行上網查詢手冊。

In [ ]:

```
import numpy as np
import matplotlib.pyplot as plt
xmin, xmax = -5, 5
x = np.linspace(xmin, xmax, 100)
y = (1 - np.exp(-2*x)) / (1 + np.exp(-2*x))
fig, ax = plt.subplots(1)
ax.plot(x, y, color='b')
ax.set_ylim([-3, 3])
ax.set_title('$f(x) = (1 - e^{-2x}) / (1 + e^{-2x})$')
ax.grid(True)
plt.show()
```



## 3. 繪製函數¶

$$f(x) = \sqrt[3]{\frac{4 - x^3}{1 + x^2}}$$

注意事項與討論：

- 畫出下列函數，X 軸範圍自訂，盡量畫出最完整的函數圖，能呈現隱藏在函數的資訊。
- 將上述範例介紹的繪圖技巧充分地應用，讓自己熟悉各種繪圖的表現，包括文字、刻度、顏色、legend、grid、座標軸的位置... 等。不是只將函數畫出來而已，還要畫得好。有些函數需要的指令不再前面的範例，請自行上網查詢手冊。

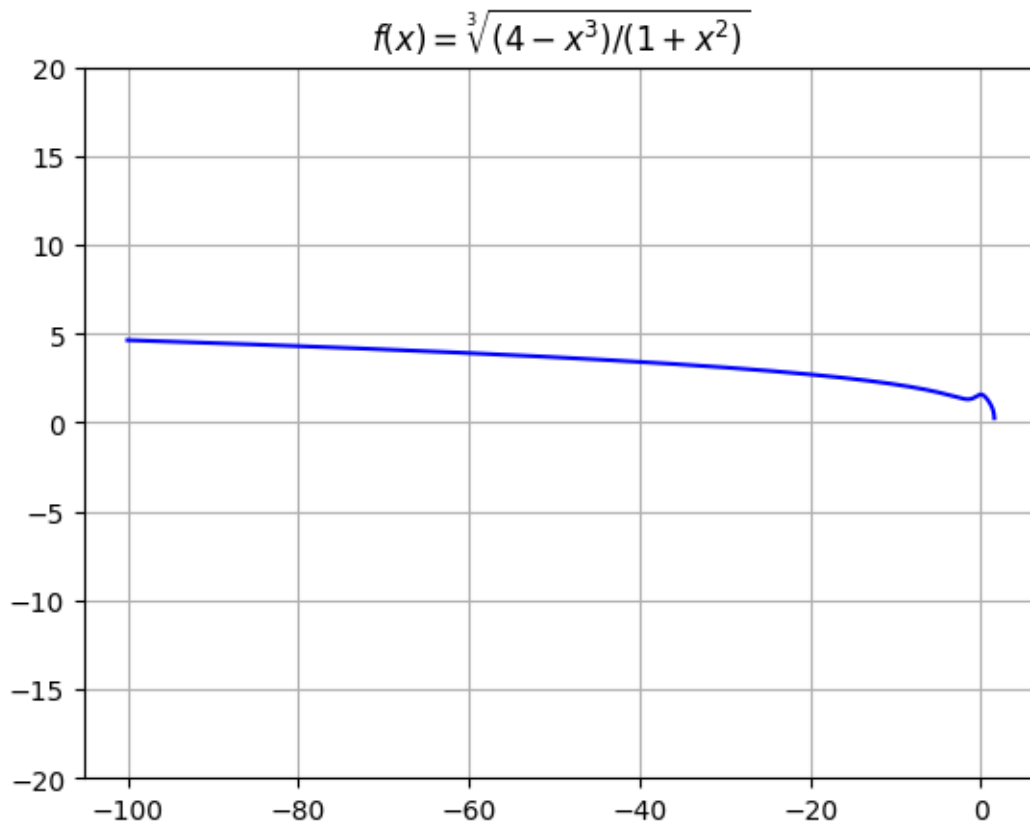
In [ ]:

```
import numpy as np
import matplotlib.pyplot as plt
x = np.arange(-100, 100, 0.01)
f = lambda x : ((4 - x**3) / (1 + x**2))**(1/3)
```

```
fig, ax = plt.subplots(1)
ax.plot(x, f(x), color='b')
ax.set_ylim([-20, 20])
ax.set_title('$f(x) = \sqrt[3]{(4-x^3)/(1+x^2)}$')
ax.grid(True)
plt.show()
```

C:\Users\3hhsi\AppData\Local\Temp\ipykernel\_1124\1713147981.py:4: RuntimeWarning:  
invalid value encountered in power

```
f = lambda x : ((4-x**3)/(1+x**2))**(1/3)
```



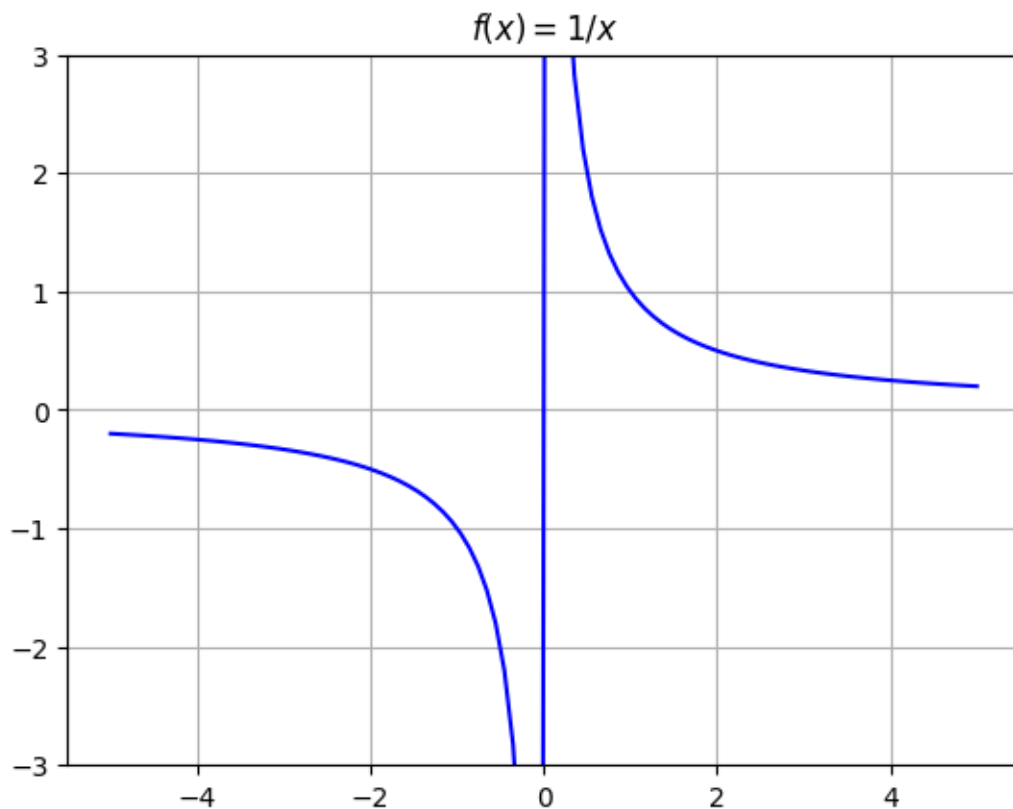
#### 4. 繪製函數

$f(x) = \frac{1}{x}$

注意事項與討論：

- 畫出下列函數，X 軸範圍自訂，盡量畫出最完整的函數圖，能呈現隱藏在函數的資訊。
- 將上述範例介紹的繪圖技巧充分地應用，讓自己熟悉各種繪圖的表現，包括文字、刻度、顏色、legend、grid、座標軸的位置... 等。不是只將函數畫出來而已，還要畫得好。有些函數需要的指令不再前面的範例，請自行上網查詢手冊。

```
In [ ]:
import numpy as np
import matplotlib.pyplot as plt
xmin, xmax = -5, 5
x = np.linspace(xmin, xmax, 100)
y = 1/x
fig, ax = plt.subplots(1)
ax.plot(x, y, color='b')
ax.set_ylim([-3, 3])
ax.grid(True)
ax.set_title('$f(x) = 1/x$')
plt.show()
```



## 5. 繪製函數

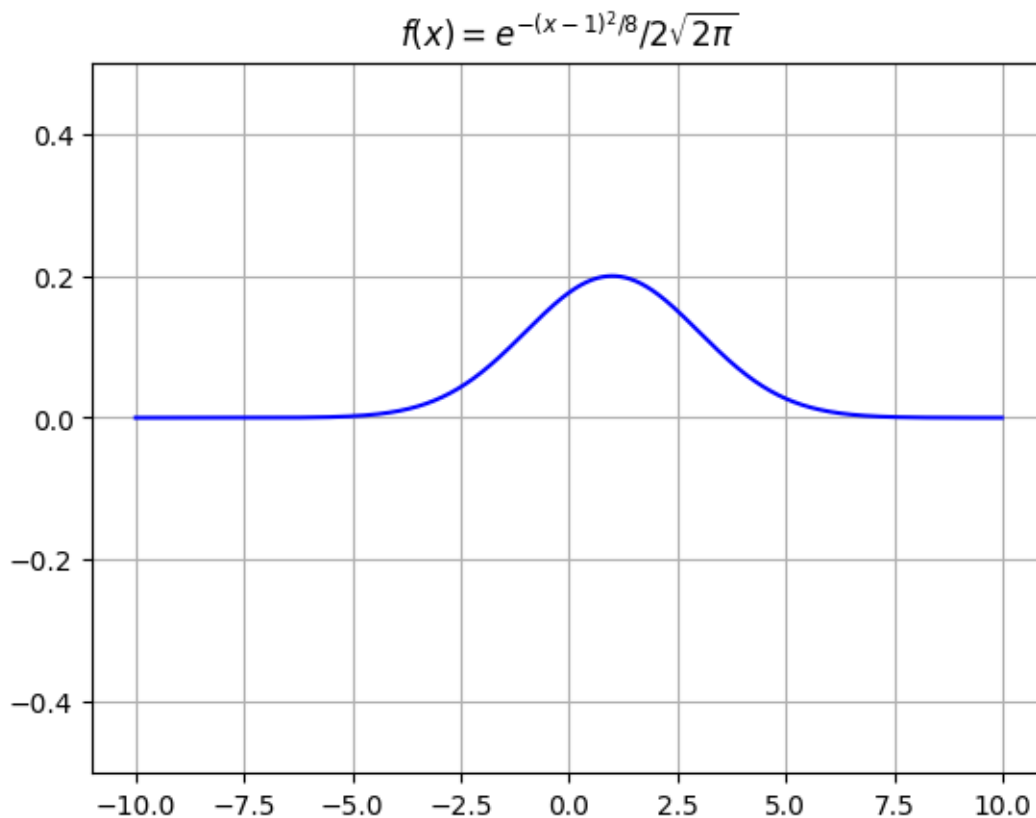
$f(x) = \frac{1}{\sqrt{2\pi}} e^{-\frac{(x-1)^2}{8}}$

注意事項與討論：

- 畫出下列函數，X 軸範圍自訂，盡量畫出最完整的函數圖，能呈現隱藏在函數的資訊。
- 將上述範例介紹的繪圖技巧充分地應用，讓自己熟悉各種繪圖的表現，包括文字、刻度、顏色、legend、grid、座標軸的位置... 等。不是只將函數畫出來而已，還要畫得好。有些函數需要的指令不再前面的範例，請自行上網查詢手冊。

In [ ]:

```
import numpy as np
import matplotlib.pyplot as plt
x = np.arange(-10, 10, 0.01)
f = lambda x : 1/(2*((2*np.pi)**(1/2)))*np.exp(-((x-1)**2)/8)
fig, ax = plt.subplots(1)
ax.plot(x, f(x), color='b')
ax.set_ylim([-0.5, 0.5])
ax.grid(True)
ax.set_title('$f(x) = e^{\{-(x-1)^2\}/8}/2\sqrt{2\pi}$')
plt.show()
```



## 6. 繪製函數¶

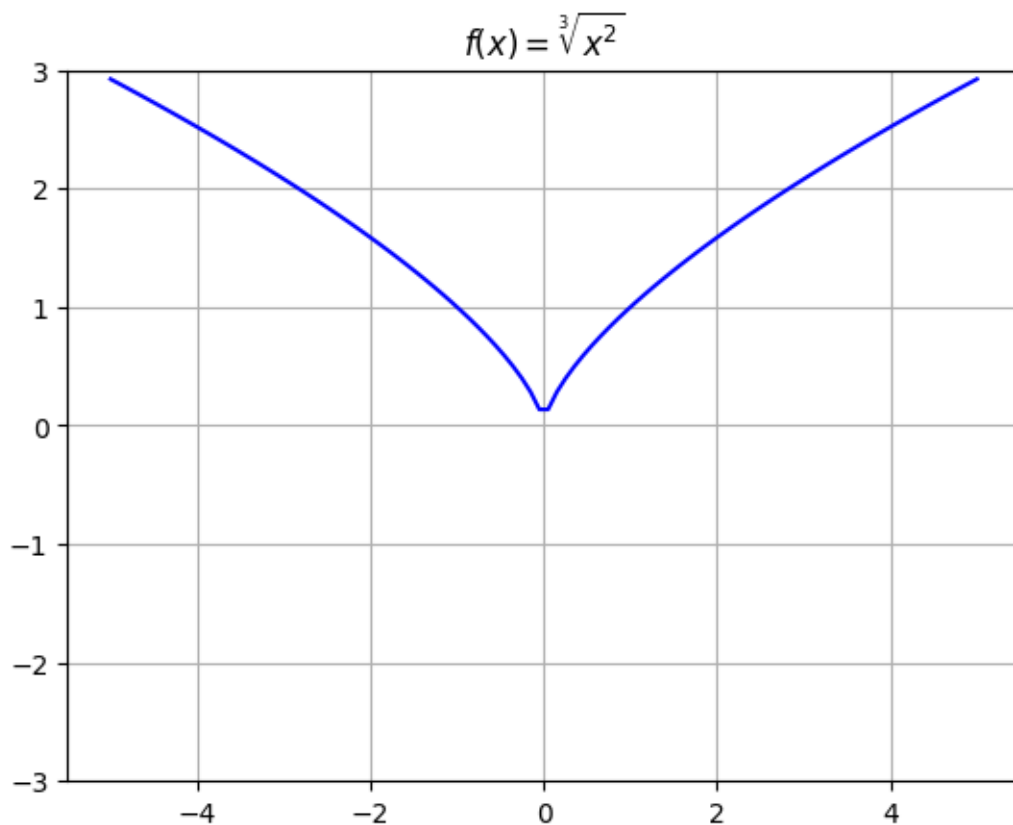
$f(x) = \sqrt[3]{x^2}$

注意事項與討論：

- 畫出下列函數，X 軸範圍自訂，盡量畫出最完整的函數圖，能呈現隱藏在函數的資訊。
- 將上述範例介紹的繪圖技巧充分地應用，讓自己熟悉各種繪圖的表現，包括文字、刻度、顏色、legend、grid、座標軸的位置... 等。不是只將函數畫出來而已，還要畫得好。有些函數需要的指令不再前面的範例，請自行上網查詢手冊。

In [ ]:

```
import numpy as np
import matplotlib.pyplot as plt
xmin, xmax = -5, 5
x = np.linspace(xmin, xmax, 100)
y = (x**(2))**(1/3)
fig, ax = plt.subplots(1)
ax.plot(x, y, color='b')
ax.set_ylim([-3, 3])
ax.grid(True)
ax.set_title('$f(x) = \sqrt[3]{x^2}$')
plt.show()
```



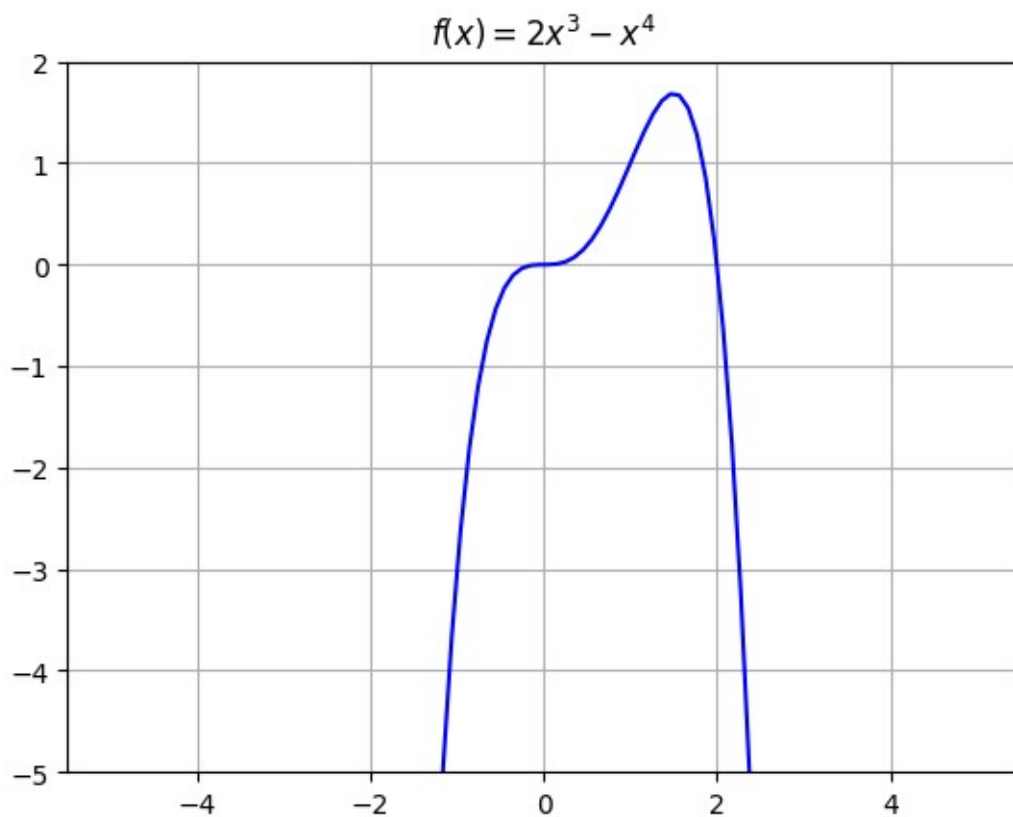
## 7. 繪製函數

$$f(x) = 2x^3 - x^4$$

注意事項與討論：

- 畫出下列函數，X 軸範圍自訂，盡量畫出最完整的函數圖，能呈現隱藏在函數的資訊。
- 將上述範例介紹的繪圖技巧充分地應用，讓自己熟悉各種繪圖的表現，包括文字、刻度、顏色、legend、grid、座標軸的位置... 等。不是只將函數畫出來而已，還要畫得好。有些函數需要的指令不再前面的範例，請自行上網查詢手冊。

```
In [ ]:
import numpy as np
import matplotlib.pyplot as plt
xmin, xmax = -5, 5
x = np.linspace(xmin, xmax, 100)
y = 2*x**3 - x**4
fig, ax = plt.subplots(1)
ax.plot(x, y, color='b')
ax.set_ylim([-5, 2])
ax.grid(True)
ax.set_title('$f(x) = 2x^3 - x^4$')
plt.show()
```



## 8. 繪製函數

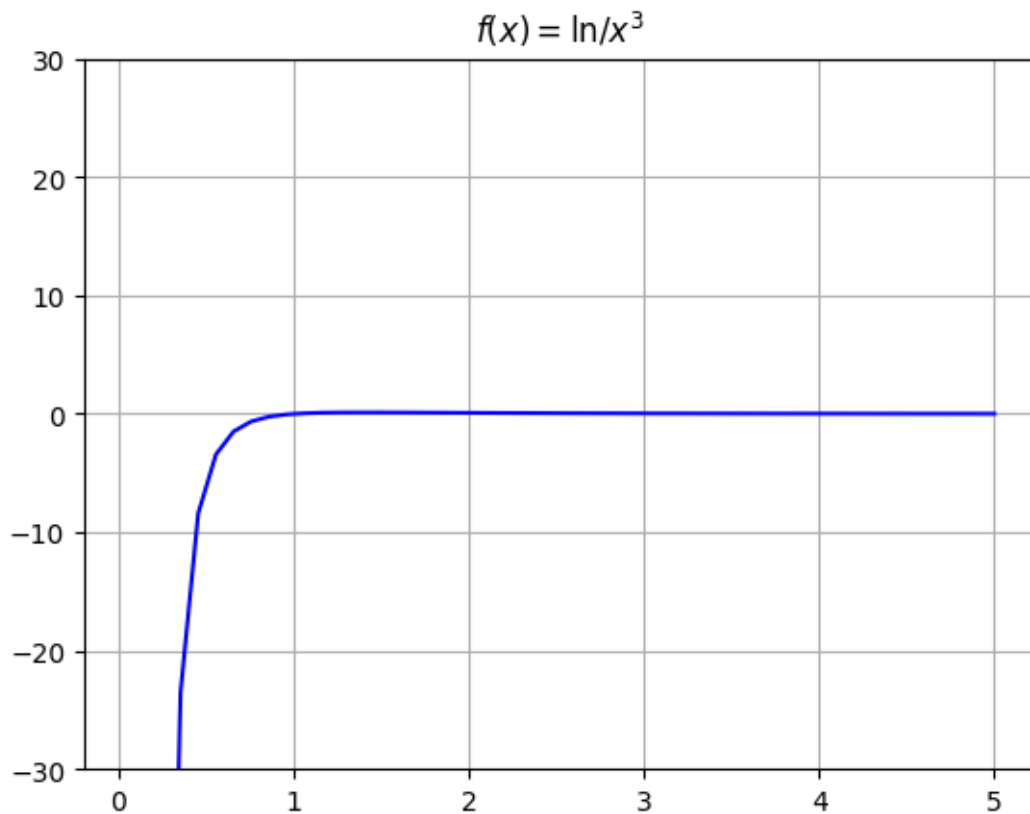
$f(x) = \frac{\ln}{x^3}$

注意事項與討論：

- 畫出下列函數，X 軸範圍自訂，盡量畫出最完整的函數圖，能呈現隱藏在函數的資訊。
- 將上述範例介紹的繪圖技巧充分地應用，讓自己熟悉各種繪圖的表現，包括文字、刻度、顏色、legend、grid、座標軸的位置... 等。不是只將函數畫出來而已，還要畫得好。有些函數需要的指令不再前面的範例，請自行上網查詢手冊。

```
In [ ]:
import numpy as np
import matplotlib.pyplot as plt
xmin, xmax = -5, 5
x = np.linspace(xmin, xmax, 100)
y = np.log(x) / x**3
fig, ax = plt.subplots(1)
ax.plot(x, y, color='b')
ax.set_ylim([-30, 30])
ax.grid(True)
ax.set_title('$f(x) = \ln/x^3$')
plt.show()
```

C:\Users\3hhsi\AppData\Local\Temp\ipykernel\_1124\919468924.py:5: RuntimeWarning:  
invalid value encountered in log  
y = np.log(x) / x\*\*3



## 9. 繪製函數

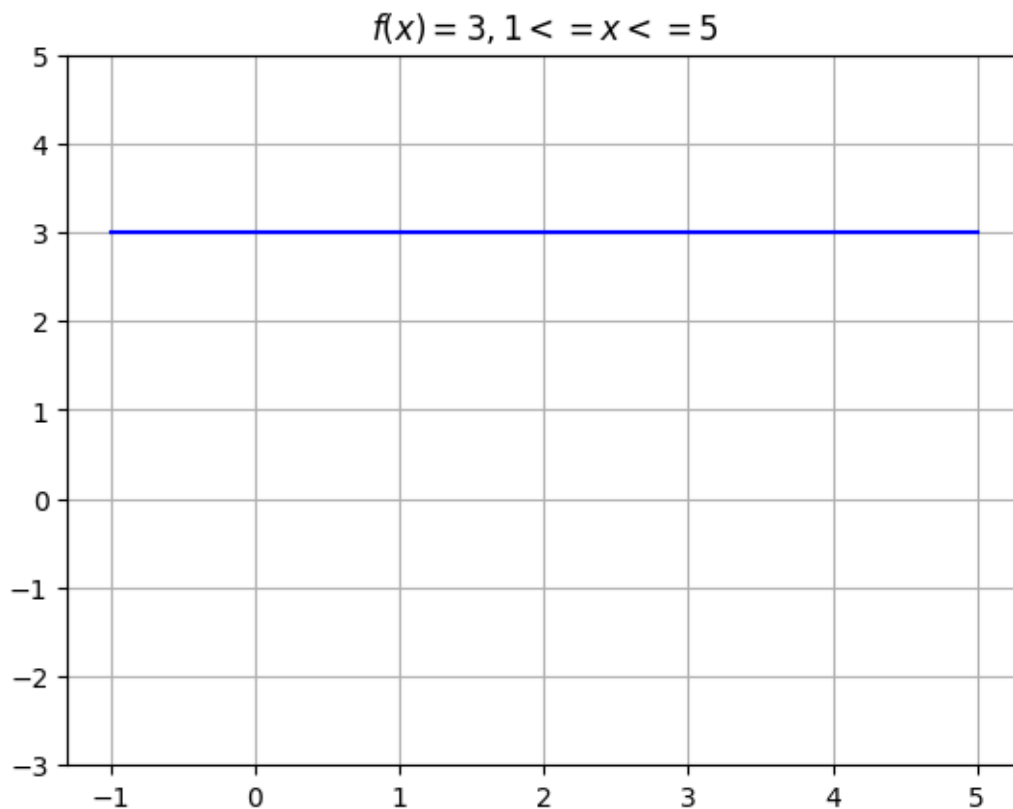
$f(x) = 3, 1 \leq x \leq 5$

注意事項與討論：

- 畫出下列函數，X 軸範圍自訂，盡量畫出最完整的函數圖，能呈現隱藏在函數的資訊。
- 將上述範例介紹的繪圖技巧充分地應用，讓自己熟悉各種繪圖的表現，包括文字、刻度、顏色、legend、grid、座標軸的位置... 等。不是只將函數畫出來而已，還要畫得好。有些函數需要的指令不再前面的範例，請自行上網查詢手冊。

```
In [ ]:
import numpy as np
import matplotlib.pyplot as plt
xmin, xmax = -1, 5
x = np.linspace(xmin, xmax, 100)
y = 3*x**0
fig, ax = plt.subplots(1)
ax.plot(x, y, color='b')
ax.set_ylim([-3, 5])
ax.grid(True)
ax.set_title('$f(x) = 3, 1 \leq x \leq 5$')
plt.show()
```





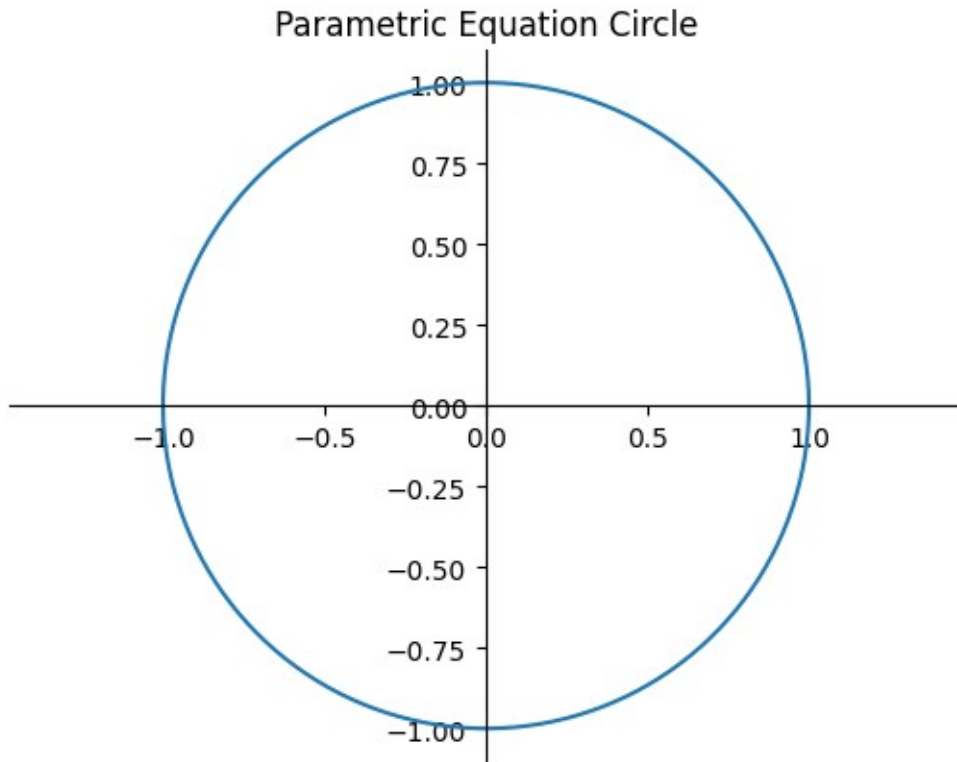
## 10. 繪製函數

$$x^2 + y^2 = 1$$

注意事項與討論：

- 畫出下列函數，X 軸範圍自訂，盡量畫出最完整的函數圖，能呈現隱藏在函數的資訊。
- 將上述範例介紹的繪圖技巧充分地應用，讓自己熟悉各種繪圖的表現，包括文字、刻度、顏色、legend、grid、座標軸的位置... 等。不是只將函數畫出來而已，還要畫得好。有些函數需要的指令不再前面的範例，請自行上網查詢手冊。

```
In [ ]:
import numpy as np
import matplotlib.pyplot as plt
r = 1.0
a, b = (0, 0)
theta = np.arange(0, 2*np.pi, 0.01)
x = a + r * np.cos(theta)
y = b + r * np.sin(theta)
fig = plt.figure()
axes = fig.add_subplot(111)
axes.plot(x, y)
axes.axis('equal')
axes.spines['top'].set_visible(False)
axes.spines['right'].set_visible(False)
axes.spines['left'].set_position(('data',0))
axes.spines['bottom'].set_position(('data',0))
plt.title('Parametric Equation Circle')
plt.show()
```



## 11. 繪製函數

- 繪製如下右圖的正方形。可以展示多種繪圖方式。

注意事項與討論：

- 畫出下列函數，X 軸範圍自訂，盡量畫出最完整的函數圖，能呈現隱藏在函數的資訊。
- 將上述範例介紹的繪圖技巧充分地應用，讓自己熟悉各種繪圖的表現，包括文字、刻度、顏色、legend、grid、座標軸的位置... 等。不是只將函數畫出來而已，還要畫得好。有些函數需要的指令不再前面的範例，請自行上網查詢手冊。

In [ ]:

```
import matplotlib.pyplot as plt
```

```
figure, ax = plt.subplots()
```

```
x=[-0.5,0.5,0.5,-0.5,-0.5]
y=[-0.5,-0.5,0.5,0.5,-0.5]
ax.set_xticks(np.array([-1, -0.5, 0, 0.5, 1]))
ax.set_xticklabels(['-1', '-0.5', '0', '0.5', '1'], fontsize=10)
ax.set_yticks(np.array([-1, -0.5, 0, 0.5, 1]))
ax.set_yticklabels(['-1', '-0.5', '0', '0.5', '1'], fontsize=10)
ax.set_xlim([-1, 1])
ax.set_ylim([-1, 1])
ax.spines['top'].set_visible(False)
ax.spines['right'].set_visible(False)
ax.spines['left'].set_position(('data',0))
ax.spines['bottom'].set_position(('data',0))
ax.grid(False)
ax.set_aspect(1)

plt.plot(x,y)
plt.title('A square of side 1')
plt.show()
```

A square of side 1

