Practical no. – 2

Aim: write a program to create a new plot by rotating the given numbers by a degree 90, 180, 270 degrees

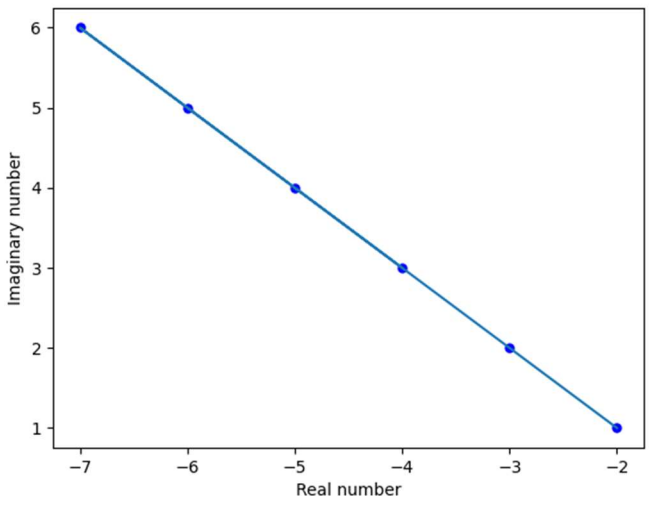
1. Rotation by 90 degrees

Code:

```
import numpy as np
import matplotlib.pyplot as plt

s = np.array([1+2j,2+3j,4+5j,5+6j,6+7j,3+4j])
l=np.array([z*1j for z in s])
print(l)

x = l.real
y = l.imag
plt.scatter(x,y,label="Comple number",color="b",s=25,marker="o")

plt.xlabel("Real number")
plt.ylabel("Imaginary number")
plt.plot(x,y)
plt.show()
```

Output:

```
[-2.+1.j -3.+2.j -5.+4.j -6.+5.j -7.+6.j -4.+3.j]
```



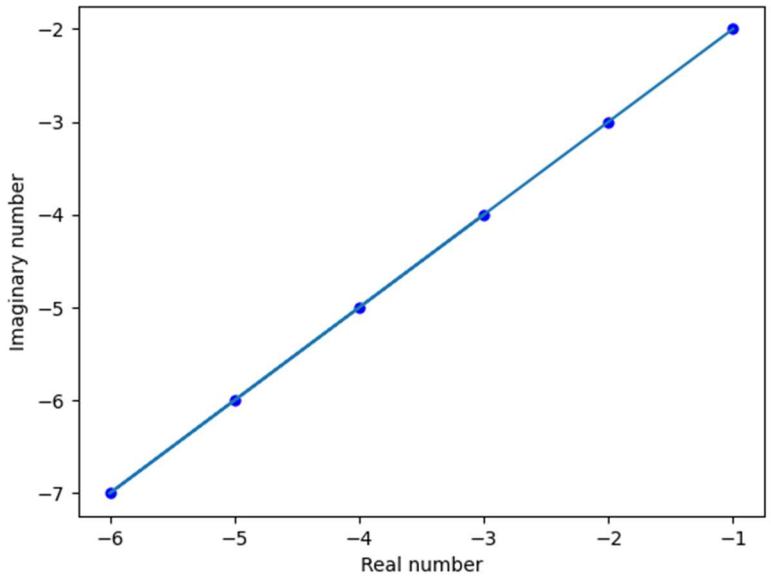
2. Rotation by 180 degrees

Code:

```
import numpy as np
import matplotlib.pyplot as plt

s = np.array([1+2j,2+3j,4+5j,5+6j,6+7j,3+4j])

l = np.array([z*-1 for z in s])
print(l)
x=l.real
y=l.imag

plt.scatter(x,y,label="Comple number",s=25,color="b",marker="o")
plt.xlabel("Real number")
plt.ylabel("Imaginary number")
plt.plot(x,y)
plt.show()
```

Output:

```
[-1.-2.j -2.-3.j -4.-5.j -5.-6.j -6.-7.j -3.-4.j]
```



3. Rotation by 270 degrees

Code:

```
import numpy as np
import matplotlib.pyplot as plt

s = np.array([1+2j,2+3j,4+5j,5+6j,6+7j,3+4j])

# main logic "z*1j-1 for z in s"
# where s is the array of complex number
# iterates through every element in s and multiply every element
by 1j-1
# z store the singal value i.e. 1+2j so on for each iteration
# as it is in the "[ ]" creates an list of new complex nums and
using array method convert that into an array form
l = np.array([z*1j-1 for z in s])
print(l)

x=l.real
y=l.imag

plt.scatter(x,y,label="Complex
number",color="b",s=25,marker='o')

plt.xlabel("Real number")
plt.ylabel("Imaginary number")

plt.plot(x,y)
plt.show()
```

Output:

```
[-2.+1.j -3.+2.j -5.+4.j -6.+5.j -7.+6.j -4.+3.j]
```