1. The energy and water council in London Ontario oversees 6 city districts. Each district consumes an average amount of water per day (in tons per day). Due to increasing scarcity of water, the city wants each district to reduce the amount of water used to 80% of the previous year.

Average amount of water used by Londoners on September 2nd 2021, by district (in tons per day):

- Oakridge: 20
- Masonville: 30
- Lambeth: 100
- Sunningdale: 62
- Old East Village: 94
- White Hills: 120

Put all information into a numpy array, and calculate London's water consumption goal for September 2nd 2022. Then plot 2021, and projected 2022 data in a scatterplot with different colours and a labelled x and y axis. Remember to import all of the packages you wish to use to solve this problem. (Hint: create an additional X array with indexes correlating to all 6 districts. You do not need to label them by their district name).

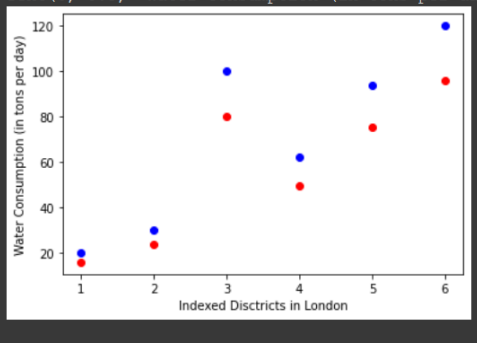
```
import numpy as np
import matplotlib.pyplot as plt

water_usage_2021 = np.array([20, 30, 100, 62, 94, 120])
water_usage_goal_2022 = water_usage_2021 * 0.8

x_labels = np.array([1, 2, 3, 4, 5, 6])

plt.scatter(x_labels, water_usage_2021, c = "blue")
plt.scatter(x_labels, water_usage_goal_2022, c = "red")
plt.xlabel("Indexed Disctricts in London")
plt.ylabel("Water Consumption (in tons per day)")
```

```
Text(0, 0.5, 'Water Consumption (in tons per day)')
```



2. Use two nested for loops to print out this design of asterisks.

* * * * *

* * * *

* * *

* *

*

Note: in python you can use print(“* ”, end = “”) to print a single star without a line break and you can use print(“”) to print a line skip.

```
for i in range(5):
  for j in range (i, 5):
    print("* ", end= "")
  print("")
```

```
* * * * *
* * * *
* * *
* *
*
```

3.

(a) Solve $u'(t) + \frac{1}{t}u(t) = e^t$ subject to $u(1) = 3$.

Then display u(t) over the interval between 0 and 5 and a y-limit between 0 and 100. Remember to import sympy.

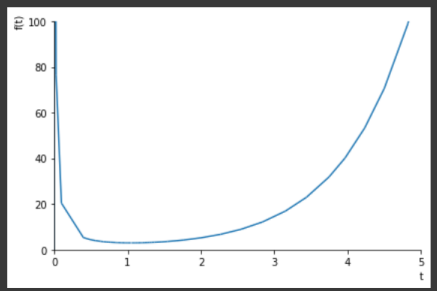
```
from sympy import *
from sympy.solvers.ode import dsolve
from sympy.plotting import plot
import numpy as np

t = Symbol("t")
u = Function("u")(t)

de = Eq(u.diff(t) + 1 / t * u, np.e ** t)

solution = dsolve(de, u, ics={u.subs(t, 1):3})

plot(solution.rhs, xlim = [0, 5], ylim = [0, 100])
```




```
<sympy.plotting.plot.Plot at 0x7f1d5441a2d0>
```

4. Take the numpy array:

Celsius = np.array([10, 100, 0, -30]) and convert it to F

Remember the equation of conversion is (°C × 9/5) + 32 = °F

Then print this array.

```
import numpy as np

celcius = np.array([10, 100, 0, -30])
farenheit = celcius * 9 / 5 + 32

print(farenheit)
```

```
[ 50. 212.  32. -22.]
```