

```
In [11]: def chickens(count):
        return f"Number of chickens: {count}" if count < 10 else "Number of chickens: many"

In [12]: def string_both_ends_3(s):
        return s if len(s) < 3 else s[:3]+s[len(s)-3:]

In [13]: def first_char_replace(s):
        return s[0]+s[1:].replace(s[0].lower(), '@').replace(s[0].upper(), '@')

In [14]: def string_jumble(a, b):
        return b[:2]+a[2:]+ ' '+a[:2]+b[2:]

In [15]: def match_first_last(words):
        words = [s[0]+s[-1] for s in words if len(s)>=2]
        sum = [0]
        while len(words) > 0:
            sum.append(words.count(words[0]))
            words.remove(words[0])
        return max(sum)

In [16]: def group_strings(words):
        return sorted(words)

In [17]: def sort_last(tuples):
        return sorted(tuples, key=lambda x:x[1])

In [18]: def main():
        print('Number of chickens')
        print(chickens(4))
        print(chickens(9))
        print(chickens(10))
        print(chickens(99))

        print('\n3 characters from both ends')
        print(string_both_ends_3('spring'))
        print(string_both_ends_3('Intelligence'))
        print(string_both_ends_3('a'))
        print(string_both_ends_3('xyz'))

        print('\nReplace occurrences of first character')
        print(first_char_replace('babble'))
        print(first_char_replace('aardvark'))
        print(first_char_replace('google'))
        print(first_char_replace('Ooogle'))

        print('\nString Jumble')
        print(string_jumble('mix', 'pod'))
        print(string_jumble('dog', 'dinner'))
        print(string_jumble('gnash', 'sport'))
        print(string_jumble('pezzy', 'firm'))

        print('\nMatching first and last characters')
        print(match_first_last(['aba', 'xyz', 'aa', 'a', 'bbb']))
```

```

print(match_first_last(['', 'x', 'ay', 'ayx', 'ax']))
print(match_first_last(['aaa', 'be', 'abc', 'aello']))

print ('\nGroup string in a list')
print(group_strings(['bbb', 'ccc', 'axx', 'xzz', 'aaa']))
print(group_strings(['ccc', 'abb', 'aaa', 'xcc', 'aaa']))
print(group_strings(['mix', 'xyz', 'apple', 'xanadu', 'aardvark']))

print ('\nsort_last')
print(sort_last([(1, 3), (3, 2), (2, 1)]))
print(sort_last([(2, 3), (1, 2), (3, 1)]))
print(sort_last([(1, 7), (1, 3), (3, 4, 5), (2, 2)]))

```

In [19]:

```

...
Python files .py are modules. Modules can define variables, functions, and classes.
When a Python interpreter reads a Python file, it first sets a few special variables
Then it executes the code from the file.

One of those variables is called __name__.

When the interpreter runs a module, the __name__ variable will be set as __main__
if the module that is being run is the main program.

If the code is importing the module from another module, then the __name__
variable will be set to that module's name.
...
# Standard boilerplate to call the main() function.
if __name__ == '__main__':
    main()

```

Number of chickens
 Number of chickens: 4
 Number of chickens: 9
 Number of chickens: many
 Number of chickens: many

3 characters from both ends
 spring
 Intnce
 a
 xyzxyz

Replace occurrences of first character
 ba@@le
 a@rdv@rk
 goo@le
 O@@gle

String Jumble
 pox mid
 dig donner
 spash gnort
 fizzy perm

Matching first and last characters
 2
 2
 1

Group string in a list
 ['aaa', 'axx', 'bbb', 'ccc', 'xzz']
 ['aaa', 'aaa', 'abb', 'ccc', 'xcc']
 ['aardvark', 'apple', 'mix', 'xanadu', 'xyz']

sort_last
 [(2, 1), (3, 2), (1, 3)]

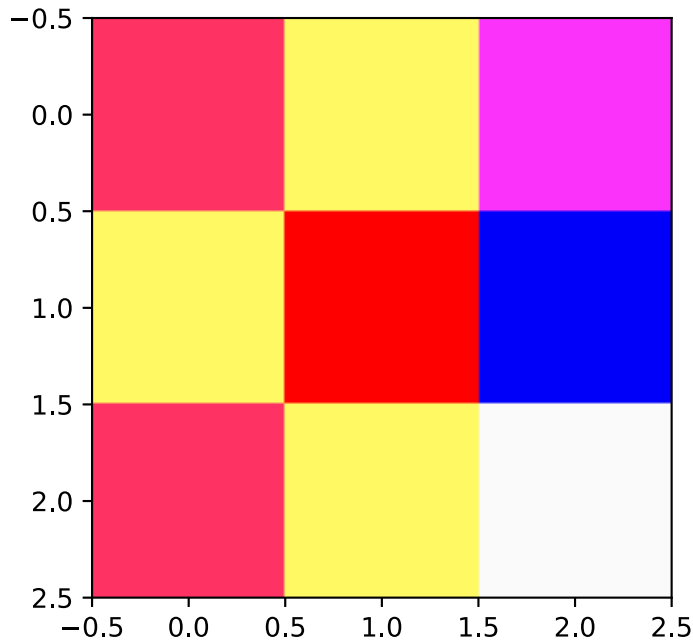
```
[(3, 1), (1, 2), (2, 3)]
[(2, 2), (1, 3), (3, 4, 5), (1, 7)]
```

In [20]:

```
#6.8
import numpy as np
import matplotlib.pyplot as plt

def IIIDarrToImage(IIID):
    plt.imshow(IIID)

IIIDarrToImage(np.array([[255,50,100],[255,250,100],[250,50,250]],[255,250,100],[2
```



In [21]:

```
# 6.9
import pandas as pd
df = pd.read_csv('train.csv')

print("Age between 18 and 30")
print(len(df[(df['Age'] < 30) & (df['Age'] > 18)]))
```

Age between 18 and 30
245

In [22]:

```
print("Females survivors age between 18 and 30")
print(len(df[(df['Sex'] == 'female')
              & (df['Age'] < 30)
              & (df['Age'] > 18)
              & (df['Survived'] == 1)]))
```

Females survivors age between 18 and 30
59