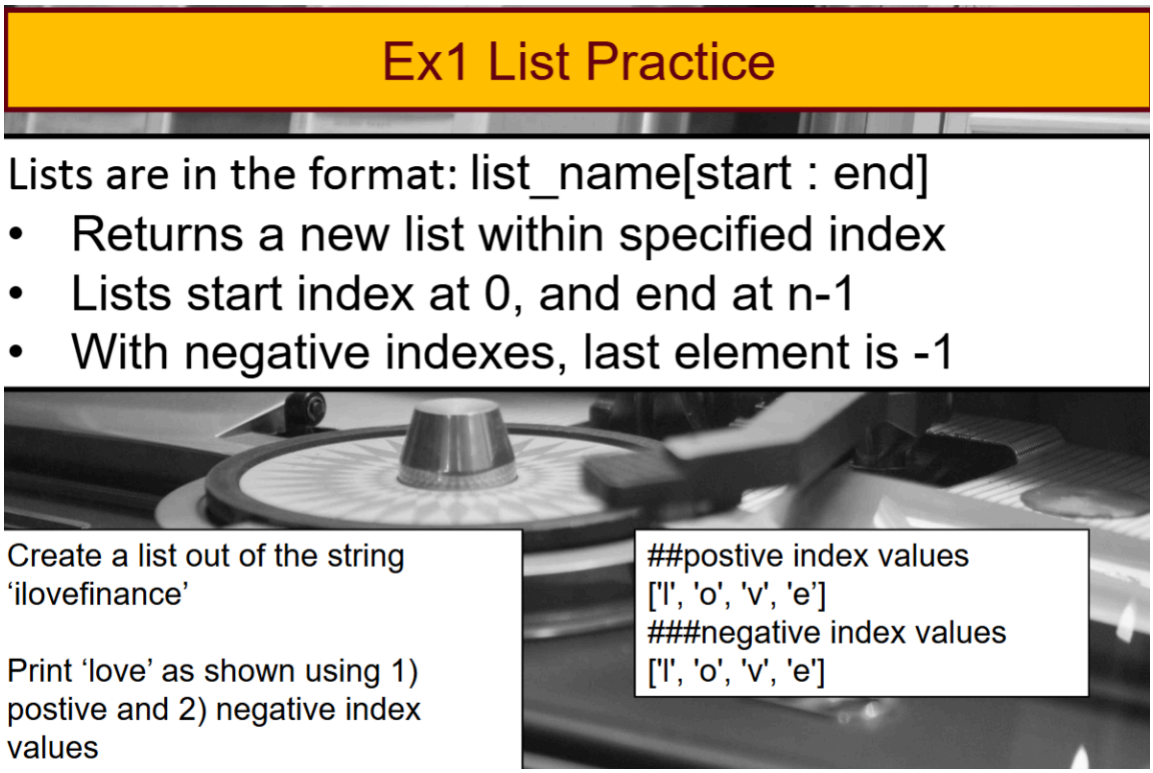
Class 8 - July 31th Notes

EX -1

Slide 12



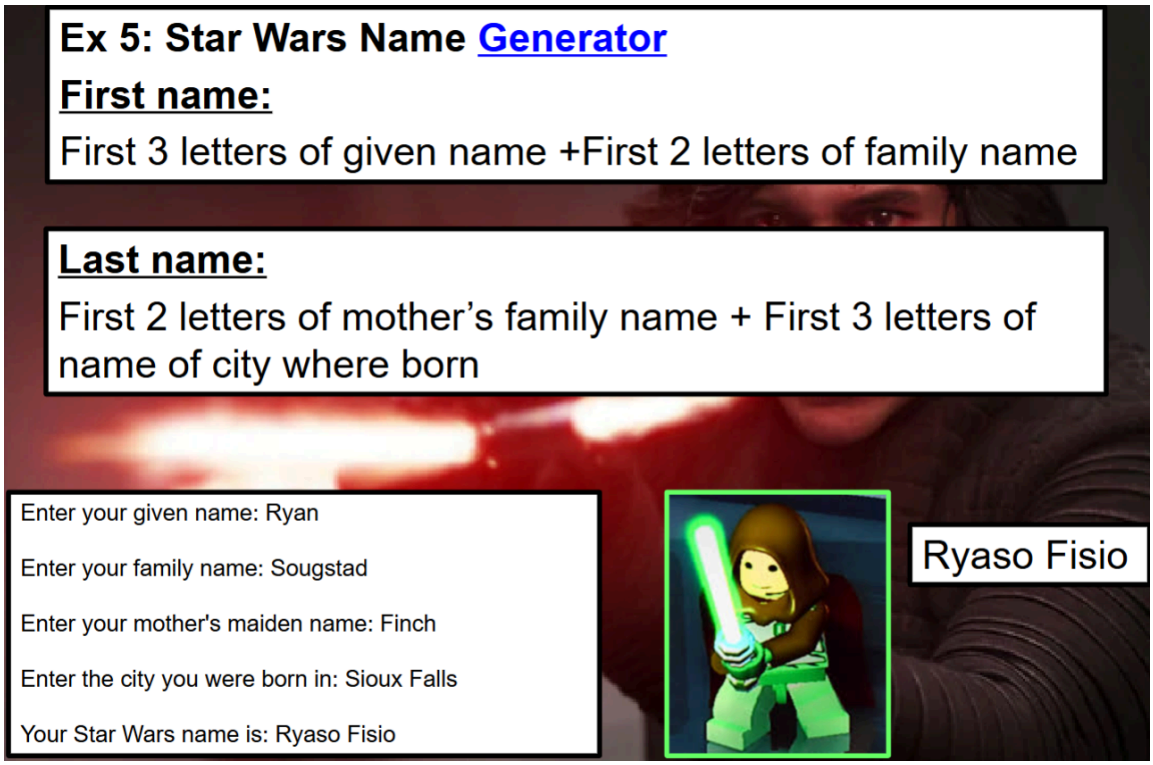
In [19]:
```
#Ex 1 Create a list out of the string 'ilovefinance'
#Slice and print ['l', 'o', 'v', 'e'] using 1) postive and 2) negative index values

l1 = list('ilovefinance')
print(l1[1:5])
print(l1[-11:-7])
```

```
['l', 'o', 'v', 'e']
['l', 'o', 'v', 'e']
```

Ex4 already completed

Ex - 5

Slide 29



In [58]:
```
fname = input("Enter your given name: ").lower()
lname = input("Enter your family name: ").lower()
mname = input("Enter your mother's maiden name: ").lower()
city = input("Enter the city you were born in: ").lower()

#Rohah Medel
star_war_fname = fname[0:3]+lname[0:2]
star_war_fname=star_war_fname.capitalize()
star_war_lname = mname[0:2]+city[0:3]
star_war_lname=star_war_lname.capitalize()
starwarName = star_war_fname + " " + star_war_lname
print(starwarName)
```

```
Rohah Medel
```

Practise

In [14]:
```
list1 = [33,44,66,33,21]
list2 = list1
###list2 now references list1 in memory
###if list1 is changed, the values list2 refers to also change (list 2 is not copie

list1[4] = 101

print(list1)
print(list2)
```

```
[33, 44, 66, 33, 101]
[33, 44, 66, 33, 101]
```

In [12]:
```
list1 = [33,44,66,33,21]
###Here we copy list1 and assign those values to list2
list2 = []+list1
list1[4] = 101

print(list1)
print(list2)
```

```
[33, 44, 66, 33, 101]
[33, 44, 66, 33, 21]
```

In [16]:
```
###List slicing dem0
numbers = [3,4,6,7,8,8,11]
print(numbers[2:5])
print(numbers[:4])
print(numbers[1:5:2])
print(numbers[-5:-3])
print(numbers[-7])
```

```
[6, 7, 8]
[3, 4, 6, 7]
[4, 7]
[6, 7]
3
```

In [32]:
```
###Lst Methods Demo
dj=['AXP','AAPL','BA','CAT','CVX','CSCO','KO','DIS',DWDP','XOM','GE','GS','HD','IBM
dj.append('MMM')
print(dj)
dj.sort()
print(dj)
x = dj.index('BA')
print(x)
dj.insert(3,"MMM")
print(dj)
dj.remove('CVX')
print(dj)
dj.reverse()
print(dj)
dj[0] = 'msft'
print(dj)
```

```
['AXP', 'AAPL', 'BA', 'CAT', 'CVX', 'CSCO', 'KO', 'DIS',DWDP', 'XOM', 'GE', 'GS', 'H
D', 'IBM', 'MMM']
['AAPL', 'AXP', 'BA', 'CAT', 'CSCO', 'CVX', 'DIS',DWDP', 'GE', 'GS', 'HD', 'IBM', 'K
O', 'MMM', 'XOM']
2
['AAPL', 'AXP', 'BA', 'MMM', 'CAT', 'CSCO', 'CVX', 'DIS',DWDP', 'GE', 'GS', 'HD', 'I
BM', 'KO', 'MMM', 'XOM']
['AAPL', 'AXP', 'BA', 'MMM', 'CAT', 'CSCO', 'DIS',DWDP', 'GE', 'GS', 'HD', 'IBM', 'K
O', 'MMM', 'XOM']
['XOM', 'MMM', 'KO', 'IBM', 'HD', 'GS', 'GE', 'DIS',DWDP', 'CSCO', 'CAT', 'MMM', 'B
A', 'AXP', 'AAPL']
['msft', 'MMM', 'KO', 'IBM', 'HD', 'GS', 'GE', 'DIS',DWDP', 'CSCO', 'CAT', 'MMM', 'B
A', 'AXP', 'AAPL']
```

In [38]:
```
#Demo Exception Handling
stocks = ["IBM","GOOG","FB","AAPL","TSLA"]
print("The stocks in the portfolio are",stocks)
loser = input('Which stock should I remove? ')
stocks.remove(loser)
print("Now the stocks in the portfolio are",stocks)
###execute and enter a string that does not match one in list to see value error
```

```
The stocks in the portfolio are ['IBM', 'GOOG', 'FB', 'AAPL', 'TSLA']
Now the stocks in the portfolio are ['IBM', 'GOOG', 'FB', 'TSLA']
```

In [40]:
```
#Demo Exception Handling
stocks = ["IBM","GOOG","FB","AAPL","TSLA"]
print("The stocks in the portfolio are",stocks)
loser = input('Which stock should I remove? ')
try:
    stocks.remove(loser)
    print("Now the stocks in the portfolio are",stocks)

except ValueError:
    print('That stock is not in the portfolio!')
```

```
The stocks in the portfolio are ['IBM', 'GOOG', 'FB', 'AAPL', 'TSLA']
That stock is not in the portfolio!
That stock is not in the portfolio!
```

In [41]:
```
def main():
    NUM_EMPLOYEES = int(input("Enter number of employees: "))
    hours = [0] * NUM_EMPLOYEES

###in the loop notice we print the input statement separate from input function
###inout function doesn't allow formatting
    for index in range(NUM_EMPLOYEES):
        print('Enter the hours worked by employee ',
              index + 1, ': ', sep='', end='')
        hours[index] = float(input())

    pay_rate = float(input('Enter the hourly pay rate: '))
    for index in range(NUM_EMPLOYEES):
        gross_pay = hours[index] * pay_rate
        print('Gross pay for employee ', index + 1, ': $',
              format(gross_pay, ',.2f'), sep='')

main()
```

```
Enter the hours worked by employee 1: Enter the hours worked by employee 2: Gross pa
y for employee 1: $210.00
Gross pay for employee 2: $225.00
```

In [ ]:
```

```