Exercise

Complete the output of the following program.

```python
def display_intro():
    message = "Game of Nim by Adriana Ferraro"
    length = len(message)
    symbols = "*" * length
    print(symbols)
    print(message)
    print(symbols)

message = "bye bye!"
display_intro()
print(message)
```

Output:

```
****************************
Game of Nim by Adriana Ferraro
****************************
```

CompSci 101 - Principles of Programming
Exercise

Complete the get_discount() function which returns the discount amount (a float rounded to 2 decimal places). The function is passed two parameters, the amount and the discount rate (an integer %).

```python
def get_discount(amount, rate):
    discount_message = "Discount: $" + str(get_discount(234, 5))
    print(discount_message)
    discount_message = "Discount: $" + str(get_discount(125, 15))
    print(discount_message)
```

Discount: $11.7
Discount: $18.75
Exercise

Complete the get_discount_message() function which returns a string made up of the rate of discount, the string "% Discount: $", and the discount amount. The function has two parameters, the discount amount and the rate of discount (a number).

```python
def get_discount_message(discount, rate):

discount_message = get_discount_message(11.7, 5)
print(discount_message)

discount_message = get_discount_message(98.55, 15)
print(discount_message)
```

5% Discount: $11.7
15% Discount: $98.55
Exercise

Complete the print_docket() function which prints the sales docket information (the format should be as shown in the example output shown). The function is passed two arguments, the price and the discount rate (an int %). Your function code **MUST** make a call to both functions: get_discount() and get_discount_message().

```python
def get_discount(amount, rate):
    #code from slide 23

def get_discount_message(discount, rate):
    #code from slide 24

def print_docket(price, rate):

print_docket(234, 5)
print()  
print_docket(657, 15)
```

Original price $234
5% Discount: $11.7
Price $222.3

Original price $657
15% Discount: $98.55
Price $558.45
Exercise

The following program prompts the user for a number of items to be packaged. Each box can hold 10 items. Any left over items require an extra box. The first 6 boxes cost $8 each and any boxes above the first 6, cost $5 each. The program executes as shown in the example outputs below. **Design the functions needed** to write this program and write the main code for this program.

```
Enter number of items: 20
Items: 20
Boxes needed: 2
Cost: $16

Enter number of items: 36
Items: 36
Boxes needed: 4
Cost: $32

Enter number of items: 102
Items: 102
Boxes needed: 11
Cost: $73

Enter number of items: 65
Items: 65
Boxes needed: 7
Cost: $53
```
Exercise

From the previous slide.

```python
#write the main code below
items_per_box = 10

Enter number of items: 102
Items: 102
Boxes needed: 11
Cost: $73
```