Assignment 9 – Working with Numbers

Numbers

- Integers
- Floating-point numbers

Math Module

import math

1. Code the following:
   Screen Capture #1 (2 points)

```
#! /usr/bin/env python3

import math as m

num1 = int(input("Input a number: "))
num2 = int(input("Input another number: "))

print()

# The power function
print(str(num1) + " to the power of " + str(num2)
      + " is " + str(m.pow(num1, num2)))

# The square root function
print("The square root of " + str(num1)
      + " is " + str(m.sqrt(num1)))

# The ceiling function
print("The nearest integer (rounded up) to " + str(num1/num2)
      + " is " + str(m.ceil(num1/num2)))

# The floor function
print("The nearest integer (rounded down) to " + str(num1/num2)
      + " is " + str(m.floor(num1/num2)))

# Calculate the area of a circle using the pi function
print ("The area of the circle with radius " + str(num1)
       + " is: " + str(m.pi * num1**2))

```

2. Run the app:
   Screen Capture #2 (2 points)

```
C:\Users\Saddleback\AppData\Local\Programs\Python\Python36\python
Input a number: 25
Input another number: 3

25 to the power of 3 is 15625.0
The square root of 25 is 5.0
The nearest integer (rounded up) to 8.333333333333334 is 9
The nearest integer (rounded down) to 8.333333333333334 is 8
The area of the circle with radius 25 is: 1963.4954084936207

Process finished with exit code 0
```

Formatting Numbers

float

3. Code the following:

   Screen Capture #3 (2 point)

```
#! /usr/bin/env python3

number = 12345.6789

# Format number with 2 decimal places
print("{:.2f}".format(number))

# Format number with 4 decimal places
print("{:.4f}".format(number))
```

```
Run:  SC3
C:\Users\Kelly\PycharmProjects\students\fir
12345.68
12345.6789

Process finished with exit code 0
```

integer

4. Code the following:

   Screen Capture #4 (2 point)

```
#! /usr/bin/env python3

number = 12345

# Format number without comma
print("{:d}".format(number))

# Format number with comma
print("{:,d}".format(number))
```

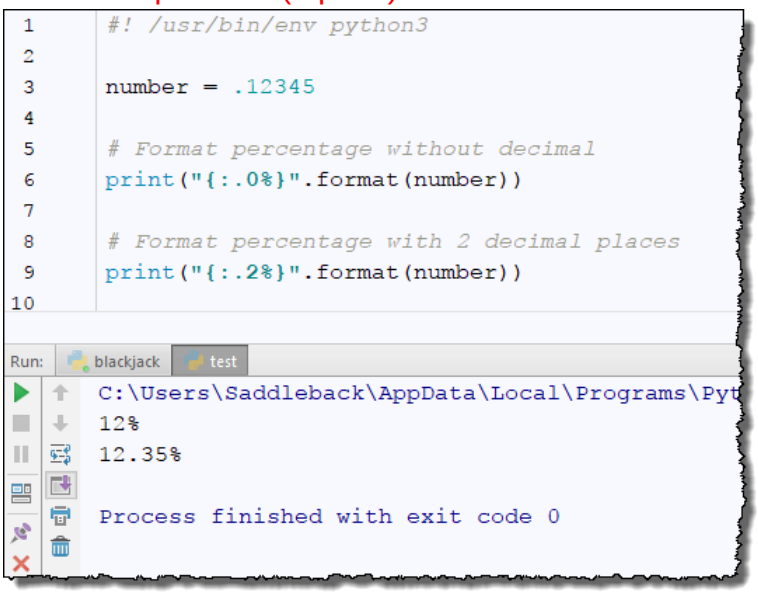
```
Run:  blackjack  test
C:\Users\Saddleback\AppData\Local\Pro
12345
12,345

Process finished with exit code 0
```

percentage

5. Code the following:

Screen Capture #5 (2 point)



```
#! /usr/bin/env python3

number = .12345

# Format percentage without decimal
print("{:.0%}".format(number))

# Format percentage with 2 decimal places
print("{:.2%}".format(number))
```

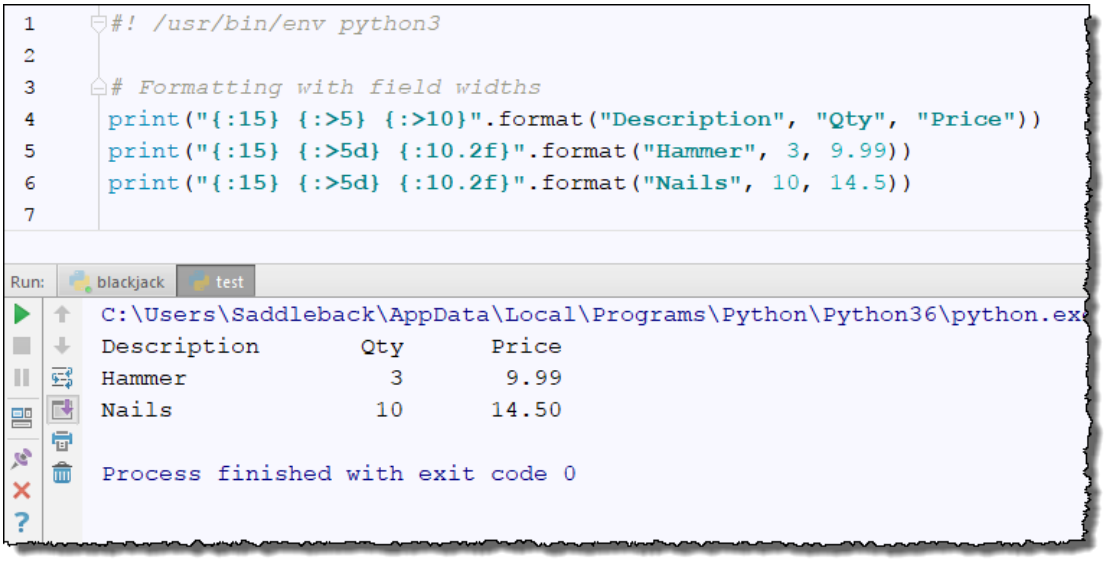
```
Run: blackjack test
C:\Users\Saddleback\AppData\Local\Programs\Pyt
12%
12.35%

Process finished with exit code 0
```

field widths

6. Code the following:

Screen Capture #6 (2 point)



```
#! /usr/bin/env python3

# Formatting with field widths
print("{:15} {:>5} {:>10}".format("Description", "Qty", "Price"))
print("{:15} {:>5d} {:10.2f}".format("Hammer", 3, 9.99))
print("{:15} {:>5d} {:10.2f}".format("Nails", 10, 14.5))
```

```
Run: blackjack test
C:\Users\Saddleback\AppData\Local\Programs\Python\Python36\python.ex
Description       Qty      Price
Hammer              3       9.99
Nails              10      14.50

Process finished with exit code 0
```

locale module

7. Code the following:

   Screen Capture #7 (3 point)

```
#! /usr/bin/env python3

import locale as lc

# Set the locale
result = lc.setlocale(lc.LC_ALL, "")  # Windows
if result[0] == "C":
    lc.setlocale(lc.LC_ALL, "en_US")  # Mac OS X

# Display currency
print(lc.currency(12345.67, grouping=True))

# Display integer
print(lc.format_string("%d", 12345, grouping=True))

# Display float
print(lc.format_string("%.2f", 12345.67, grouping=True))
```

```
Run:  sc06
"C:\Users\Saddleback\Assignment 09 new\Scripts\python.exe" "C
$12,345.67
12,345
12,345.67
```

Decimals

8. Code the following:

Screen Capture #8 (3 points)

```
#! /usr/bin/env python3

from decimal import Decimal

order_total = Decimal("100.05")
discount_percent = Decimal(".1")
discount = order_total * discount_percent

subtotal = order_total - discount
tax_percent = Decimal(".08")
sales_tax = subtotal * tax_percent
invoice_total = subtotal + sales_tax

print("Order Total:", str(order_total))
print("Discount:", str(discount))
print("Subtotal:", str(subtotal))
print("Sales Tax:", str(sales_tax))
print("Invoice Total:", str(invoice_total))

```

```
Run:  blackjack  test
C:\Users\Saddleback\AppData\Local\Programs\Pytho
Order Total: 100.05
Discount: 10.005
Subtotal: 90.045
Sales Tax: 7.20360
Invoice Total: 97.24860

Process finished with exit code 0
```

9. Format the totals using field width to align

Screen Capture #9 (2 points)

```
C:\Users\Saddleback\AppData\Local\Pro
Order Total:      100.05
Discount:          10.00
Subtotal:          90.04
Sales Tax:          7.20
Invoice Total:     97.25

Process finished with exit code 0
```

Extra Credit

To get full points for each extra credit, you must include screen captures of the running output as well as the python (.py) code files.

Extra Credit #1 – Interest Calculator (+1 Extra Credit)

Create a program that calculates the interest on a loan.

```
Interest Calculator

Enter loan amount:   520000
Enter interest rate: 5.375

Loan amount:          $520,000.00
Interest rate:              5.375%
Interest amount:       $27,950.00

Continue? (y/n): y

Enter loan amount:   4944.5
Enter interest rate: 1.3

Loan amount:            $4,944.50
Interest rate:              1.300%
Interest amount:            $64.28

Continue? (y/n): n

Bye!
```

Specifications:

- The formula for calculating the interest amount is:
  loan_amount * (interest_rate / 100)
- Use the Decimal class to make sure that all calculations are accurate. It should round the interest that's calculated to two decimal places, rounding up if the third decimal place is five or greater.
- The interest rate that's displayed can have up to 3 decimal places.
- Assume that the user will enter valid decimal values for the loan amount and interest.

Extra Credit #2 – Aircraft Fuel Calculator (+1 Extra Credit)

Create a program that calculates the amount of time and fuel for a 1980 Cessna 172N to fly a specified distance.

```
Aircraft Fuel Calculator

Distance in nautical miles: 180
Flight time: 1 hour(s) and 30 minute(s)
Required fuel: 16.8 gallons

Continue? (y/n): y

Distance in nautical miles: 121
Flight time: 1 hour(s) and 0 minute(s)
Required fuel: 12.7 gallons

Continue? (y/n): n

Bye!
```

Specifications:

- Assume that a 1980 Cessna 172N can fly 120 nautical miles (knots) per hour.
- Assume that a 1980 Cessna 172N burns 8.4 gallons of gas per hour.
- For safety, add a half hour to the flight time when calculating the amount of required fuel.
- Round the amount of required fuel to 1 decimal place. For safety, always round up, never down.
- Assume that the user will enter valid data.