Guess a Number Game: Control Flow using the \textit{if} Statement and \textbf{While} Loop

Instructor: \textbf{Joel Castellanos}

\textit{e-mail}: joel@unm.edu

\textit{Web}: http://cs.unm.edu/~joel/

\textit{Office}: Farris Engineering Center 319
Assignments

Due Tuesday, Aug 25 (for class):
- Read *Invent Your Own Computer Games with Python, 2nd Edition*
  - Chapter 4: GUESS THE NUMBER

Due Wednesday, Sept 2 (midnight):
- Lab 2: Modified Guess The Number Game
Book Version of Guess a Number Game (Lab 2 will add some mods)

- In this game, the computer generates a hidden random number from 1 to 20.
- The player is asked to guess the number in 6 guesses or less.
- After each guess, the computer will tell you if your guess is too high or too low.
- If you guess the number within six tries, you win.
- This uses random numbers, loops, if statements and input from the user in a fairly short program.
Assignment Operator

Comparison Operator

1) \texttt{bank = 110}
2) \texttt{bank = bank - 50}
3) \texttt{item = "lobster"}

4)

5) \texttt{print(bank)}
6) \texttt{print(bank == 110)}
7) \texttt{print(bank == 60)}
8) \texttt{print(bank == (110 - 50))}
9) \texttt{print(bank == item)}
10) \texttt{print(item == "tasty" )}
11) \texttt{print(item == "lobster" )}

- bank = 110
- bank = 60
- item = "lobster"
- item = "tasty"
- item = "lobster"
## Comparison Operators

<table>
<thead>
<tr>
<th>Operator</th>
<th>Question Asked</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>==</code></td>
<td>is left equal to right?</td>
</tr>
<tr>
<td><code>&gt;</code></td>
<td>is left greater than right?</td>
</tr>
<tr>
<td><code>&gt;=</code></td>
<td>is left greater than or equal to right?</td>
</tr>
<tr>
<td><code>&lt;</code></td>
<td>is left less than right?</td>
</tr>
<tr>
<td><code>&lt;=</code></td>
<td>is left less than or equal to right?</td>
</tr>
<tr>
<td><code>!=</code></td>
<td>is left not equal to right?</td>
</tr>
</tbody>
</table>

1) `print(10 == 20)` ➞ False
2) `print(10 > 20)` ➞ False
3) `print(10 < 20)` ➞ True
4) `print(10 != 20)` ➞ True
Quiz: What is the Output?

1) damage = 66
2) health = 60
3) health = health - damage
4) print(health == 0)
5) print(health >= 0)
6) print(health < 0)

a) False
   False
   True
b) True
   False
   False
c) False
   True
   False
d) True
   True
   False
e) False
   False
   False
The **if** Statement

1) `name = input('What is your Name?')`
2) `if (name == 'Joel'):`
3) `print('Great Name')`
4) `print('You are the best!')`
5) 
6) `print('Good Bye')`

Indented lines are form a **code block**.

The lines in this code block only execute if the logical expression in line 2 is **True**.
if Statement: What is Wrong?

1) name = input('What is your Name?')
2) if (name == 'Joel')
3)    print('Great Name')
4)    print('You are the best!')
5)
6) print('Good Bye')

```
line 2
    if (name == "Joel")
        ^
SyntaxError: invalid syntax
```
if Statement: What is Wrong?

1) name = input('What is your Name?')
2) if (name == 'Joel'):
   3) print('Great Name')
   4) print('You are the best!')
5)
6) print('Good Bye')

line 4
    print("You are the best!")
^  
IndentationError: unexpected indent
if Statement: What is Wrong?

1) grade = input('Enter Grade: ')
2)
3) if (grade >= 72):
4)    print('Huzzah!')
5)    print('You Passed!')
6)    print('Have a cookie.')
7)
8) print('Good bye')

Enter Grade: 78
line 3, in <module>
    if (grade >= 72):
TypeError: unorderable types: str() >= int()
The *if* Statement

1) grade = input('Enter Grade:')
2) grade = int(grade)
3)
4) if (grade >= 72):
5)    print('Huzzah!')
6)    print('You Passed!')
7)    print('Have a cookie.')
8)
9) print('Good bye')
if, elif and else

```python
def foo(grade):
    if (grade < 70):
        print('See you fall 2012')
    elif (grade <= 79):
        print('You get a C')
    elif (grade <= 89):
        print('You get a B')
    else:
        print('You get an A')

>>> foo(71)
You get a C
>>> foo(92)
You get an A
```

One and only one of the print statements executes.
if, **elif** and **else** Blocks

One and only one of these blocks executes.

```python
def foo(grade):
    if (grade < 70):
        print('See you fall 2012')
    elif (grade <= 79):
        print('You get a C')
    elif (grade <= 89):
        print('You get a B')
    else:
        print('You get an A')

print('Done')
```

```python
>>> foo(88)
You get a B
Done
```
if, `elif` and `else` Blocks

One and only one of these blocks executes.

```python
def foo(grade):
    if (grade <= 89):
        print('You get an A')
    elif (grade <= 79):
        print('You get a C')
    elif (grade <= 89):
        print('You get a B')
    else:
        print('do it again')

print('Done')
```

```python
>>> foo(66)
You get a A
Done
```
Quiz: **if, elif and else** Blocks

```python
def moo(x,y):
    if (x+y < 10): print('less than 10.')
    elif (x+y < 20): print('less than 20.')
    elif (x+y < 30): print('less than 30.')
    else: print('30 or more. ')
```

What is the output of the command:

```
>>> moo(7,8)
```

(a) less than 20.
(b) less than 10. less than 20.
(c) less than 20. less than 30.
The **if** and **else**

1) `grade = input('Enter Grade:')`
2) `grade = int(grade)`
3) 
4) `if (grade >= 72):
   print('Huzzah!')`
5) `print('You Passed!')`
6) `else:
   print('It sucks to be you.')`
7) 
8) `print('Good Bye')`
What is Wrong?

1) grade = input('Enter Grade:')
2) grade = int(grade)
3)
4) if (grade >= 90):
5)     print('Huzzah! You get an A. ')
6) if (grade >= 80):
7)     print('Great Job. You get a B. ')
8) if (grade >= 70):
9)     print('Nice Work. You get a C. ')
10) else:
11)     print('It sucks to be you. ')
12) print('Good Bye')
The **if, elif and else**

1) grade = input('Enter Grade:')  
2) grade = int(grade)  
3)  
4) if (grade >= 90):  
5)      print('Huzzah! You get an A.')  
6) elif (grade >= 80):  
7)      print('Great Job. You get a B.')  
8) elif (grade >= 70):  
9)      print('Nice Work. You get a C.')  
10) else:  
11)     print('It sucks to be you.')  
12) print('Good Bye')
What is Wrong?

1) grade = input('Enter Grade:')
2) grade = int(grade)
3)
4) if (grade >= 70):
5)     print('Nice Work. You get a C.')
6) elif (grade >= 80):
7)     print('Great Job. You get a B.')
8) elif (grade >= 90):
9)     print('Huzzah! You get an A.')
10) else:
11)     print('It sucks to be you.')
12) print('Good Bye')
Quiz: What is the output?

a = 77
b = 88

\[
\begin{align*}
\text{if} & \ (a > b): \\
& \quad \text{print('Hooch')} \\
\text{if} & \ (a < b): \\
& \quad \text{print('Quirrell')} \\
\text{if} & \ ((a + b) > 100): \\
& \quad \text{print('Sprout')} \\
\text{if} & \ ((a - b) < 0): \\
& \quad \text{print('Pomfrey')}
\end{align*}
\]

a) Hooch
b) Quirrell
   Sprout
   Pomfrey
c) Quirrell
   Pomfrey
d) Hooch
   Quirrell
   Sprout
e) Sprout
   Pomfrey
Quiz: What is the output?

```
a = 77
b = 88

if (a > 0):
    print('Lupin')
elif (b > 0):
    print('Lovegood')
elif ((a + b) > 0):
    print('Umbridge')
elif ((a * 2) > 0):
    print('Chang')
```

a) Lupin
b) Lovegood
c) Umbridge
d) Chang
e) Lupin Lovegood Umbridge Chang
Quiz: What is the output?

```python
age = 15
if (age < 18):
    print('bob')
else:
    print('paul')

if (age > 10):
    print('bill')
else:
    print('scott')
```

Options:
- a) bob
- b) paul bill
- c) paul scott
- d) bob bill
- e) bob paul bill scott
Quiz: What is the output?

```python
age = 15
if (age < 18):
    print('bob')
else:
    print('paul')

if (age > 10):
    print('bill')
else:
    print('scott')
```

Note: Indenting is different than last quiz!
The **while** Loop

```python
while (guessesTaken < 6):
    print('Take a guess.')
    guess = input()
    guess = int(guess)
    guessesTaken = guessesTaken + 1
    if (guess < number):
        print('Your guess is too low.')
    if (guess > number):
        print('Your guess is too high.')
    if (guess == number):
        break
```

If False... 
...go to the first statement past the loop block.

If True... 
...go to the first statement inside the loop block.
A Simple **while** Loop

```python
n = 4
while (n<10):
    print(n)
    n = n + 1
print('Done')
```

4
5
6
7
8
9
Done
Quiz: What is the Output?

\[ n = 2 \]
\[ \text{while } (n<11): \]
\[ \quad n = n + 2 \]
\[ \text{print}(n) \]

\[ n=4 \quad n=6 \quad n=8 \quad n=10 \quad n=12 \]

a) 2  
b) 8  
c) 10  
d) 12  
e) 2 4 6 8 10
Quiz: What is the Output?

\[ n = 2 \]
\[ \text{while } (n < 11): \]
\[ \text{print}(n) \]
\[ n = n + 2 \]

\[ n=4 \quad n=6 \quad n=8 \quad n=10 \quad n=12 \]

a) 2  

b) 8  

c) 10  

d) 12  

e) 2 4 6 8 10
Quiz: What is the Output?

```
n = 1
while (n<10):
    n = n + 2
print(n)
```

- a) 1 2 4 6 8
- b) 1 2 4 6 8 10
- c) 3 5 7 9 11
- d) 9
- e) 3 5 7 9
Quiz: What is the Output?

```
a = 1
b = 1

while (b<20):
    c = a + b
    a = b
    b = c
    print(a)
```

<p>| | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a)</td>
<td>b)</td>
<td>c)</td>
<td>d)</td>
<td>e)</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>2</td>
<td>3</td>
<td>5</td>
<td>13</td>
<td></td>
</tr>
</tbody>
</table>
Quiz: What is the Output?

```
a = 3
b = 4
while (b<18):
    c = a + b
    a = b
    b = c
print(a)
```

<p>| | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>a)</td>
<td>11</td>
<td></td>
</tr>
<tr>
<td>b)</td>
<td>13</td>
<td></td>
</tr>
<tr>
<td>c)</td>
<td>15</td>
<td></td>
</tr>
<tr>
<td>d)</td>
<td>17</td>
<td></td>
</tr>
<tr>
<td>e)</td>
<td>18</td>
<td></td>
</tr>
</tbody>
</table>
Nested **while** Loops

```python
a = 0
while (a<4):
    print(">>>>>>>>>>>a=",a)
    b = 10
    while (b<40):
        print("   a=",a," , b=",b)
        b = b + 10
    a = a + 1
print("Done")
```

```plaintext
a = 0 , b = 10
a = 0 , b = 20
a = 0 , b = 30
a = 1 , b = 10
a = 1 , b = 20
a = 1 , b = 30
a = 2 , b = 10
a = 2 , b = 20
a = 2 , b = 30
a = 3 , b = 10
a = 3 , b = 20
a = 3 , b = 30
```

Done
Using the Debugger in PyCharm

1) Set a Breakpoint.

2) Start the Debugger.

3) Add variables to Watch List.

4) Walk through the program execution with "Step Over".