Prelab 4

We encourage you to work together on the Pre Lab. The Pre Lab is not graded but will help you prepare for your lab session. In the Pre Lab, you may find questions to answer. We do not require you to provide us the answers, but we do recommend you to try to answer these questions. If you have any questions on the material in the pre lab, first check the book and recitation slides, if you do not find your answer please email your recitation TA or the course instructors.

Drawing a face

The following code draws a face on the screen. **Run the code in your idle (3.4) to see the output**. Download the file [here](#) and save it as face.py

```python
# face.py - Simple face
#
from graphics import *

def Main():
    #Create a window 500x500 pixels
    win = GraphWin('Shapes', 500, 500)
    
    # Make the window scaled
    # bottom leftmost corner is (0,0)
    # top rightmost corner is (10,10)
    win.setCoords(0.0, 0.0, 10.0, 10.0)
    
    #Draw a circle centered at 5,5
    center = Point(5, 5)
    circ = Circle(center, 4)
    circ.setFill('yellow')
    circ.draw(win)

    # Draw left eye
    eyel = Circle(Point(3,6), 1)
    eyel.setFill("red")
    eyel.draw(win)

    # Draw right eye
    eyer = Circle(Point(7,6), 1)
    eyer.setFill("red")
    eyer.draw(win)

    # Draw mouth
    rect = Rectangle(Point(4, 2), Point(6, 3))
    rect.setFill("blue")
```

rect.draw(win)

# Draw hat
hat = Polygon(Point(1, 8), Point(3, 9.5), Point(7, 9.5), Point(9, 8))
hat.setFill("orange")
hat.draw(win)

#Draw nose
nose1 = Line(Point(5, 5.5), Point(6, 4))
nose1.setWidth(5)
nose1.draw(win)
nose1.setFill("grey")
nose2 = Line(Point(5, 4), Point(6, 4))
nose2.setWidth(5)
nose2.draw(win)
nose2.setFill("grey")

# Draw message
message = Text(Point(5, 0.5), "Click anywhere to quit")
message.draw(win)

# Wait until we click mouse in the window
win.getMouse()

win.close()

Main()

Boolean Operations

Any object in Python can be tested for truth value, for use in an if or while condition or as operand of the Boolean operations. Boolean operations include and, or, not.

**x or y**: At least one of x or y is true, **x or y** is true.

Truth table:

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>x or y</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>

**x and y**: Both x or y are true, **x and y** is true.

Truth table:

<table>
<thead>
<tr>
<th>x</th>
<th>y</th>
<th>x and y</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>T</td>
<td>T</td>
</tr>
<tr>
<td>T</td>
<td>F</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>F</td>
<td>F</td>
</tr>
</tbody>
</table>
**not x**: If \( x \) is true, \( \text{not } x \) is false; otherwise is true.

Truth table:

<table>
<thead>
<tr>
<th>( x )</th>
<th>( \text{not } x )</th>
</tr>
</thead>
<tbody>
<tr>
<td>T</td>
<td>F</td>
</tr>
<tr>
<td>F</td>
<td>T</td>
</tr>
</tbody>
</table>

What are the input of the following examples?

```python
>>> print(True and True)
>>> print(False and True)
>>> print(1==1 and 2==1)
>>> print("test" == "test")
>>> print(1==1 or 2!=1)
>>> print("test" != "testing")
>>> print(not(True and False))
>>> print(3==3 and not("testing" == "testing" or "Python" == "Fun"))
```

**Decision Structure**

The **if-else** statement is used to choose exactly one out of two statements to be executed. For instance:

```python
if(condition):
    statement-1
else:
    statement-2
```

If the condition is true, statement-1 is executed and statement-2 is skipped. Otherwise, statement-1 is skipped and statement-2 is executed. statement-1, as well as statement-2 can either be single statement or compound statements.

To choose one statement to be executed from a group of statements we can use nested **if-else** statement. For example:

```python
if(condition-1):
    statement-1
elif(condition-2):
    statement-2
```

Computer Science Courses - http://courses.cs.purdue.edu/
elif(condition-n):
  statement-n
else:
  statement

In this example, condition-1, condition-2, ..., condition-n are expressions that evaluate to true or false.
statement-1, statement-2, ..., statements-n can either be single statement or compound statements.

The following code gives you an example how to implement logic operations in python.

def main():
  #ask for user input
  ch = input("Please input a single character: ")
  #check the input length
  if(not(len(ch)==1)):
    #input string length does not equal to 1
    print("Input is not a single character!")
  else:
    #input is a single character
    if((ch >= 'a') and (ch <= 'z')):
      print("This is a lower-case letter")
    elif((ch>='a') and (ch <='z')):
      print("This is a upper-case letter")
    elif((ch>='0') and (ch<='9')):
      print("This is a digit character")
    elif((ch=='+' or (ch=='-') or (ch=='*') or (ch=='/'))):
      print("This is an operator")
    else:
      print("Sorry! Input cannot be recognized")

main()