www.covenantuniversity.edu.ng

Covenant University

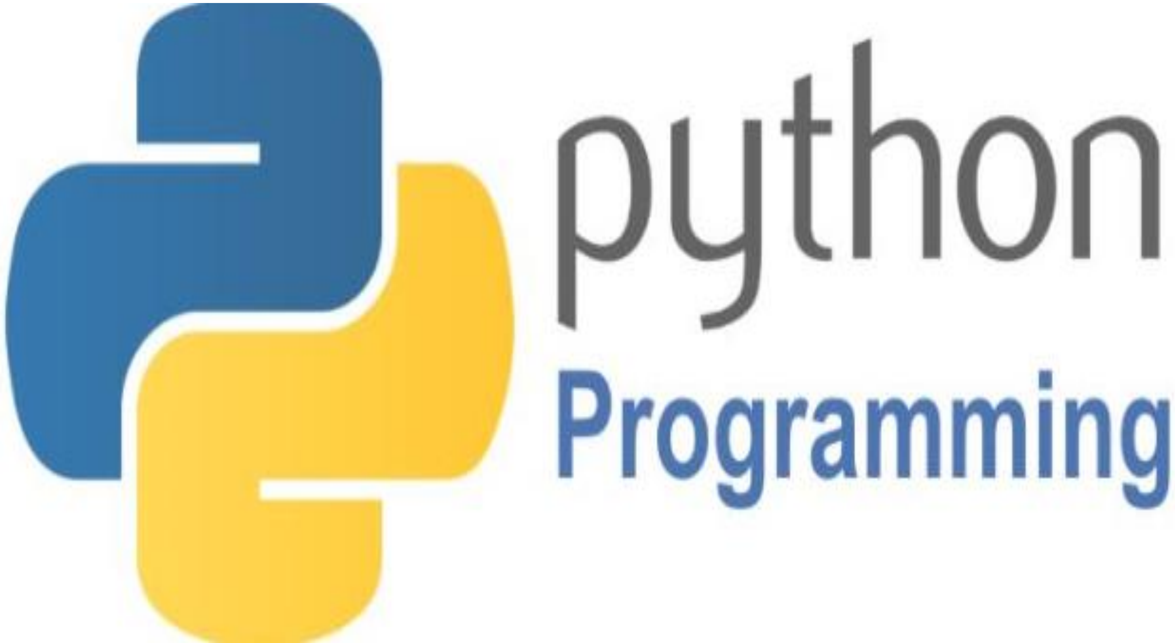
Raising a new Generation of Leaders

PET328

COMPUTER APPLICATIONS IN PETROLEUM ENGINEERING

PET328: COMPUTER APPLICATIONS IN PETROLEUM ENGINEERING

(With Python Programming)



Olatunde O. Mosobalaje (PhD)

Department of Petroleum Engineering,
Covenant University, Ota
Nigeria

OUTLINE

- Preambles
  - The Appetizer
  - The Toolbox
  - The Embedded Course
  - Introduction to Computer Programming
- Getting Started with Python
  - Basic Python Objects
  - Conditional Execution
  - Repeated Execution
  - Functions
- Python Data Structures
  - Strings
  - Lists
  - Tuples
  - Dictionaries
- Application Projects
  - Oil Reservoir Volumetrics
  - Material Balance Analysis
  - PVT Properties

Conditional Statements

- Conditional statements are written to make it possible for a program to check for some condition(s) and decide to either:
  - perform a statement(s) or skip the statement(s)
    - or
  - choose between alternative (branches of) statements.
- So, the concept of condition is central to this kind of statements.
- These conditions are crafted using the concept of Boolean expressions.

Conditional Statements

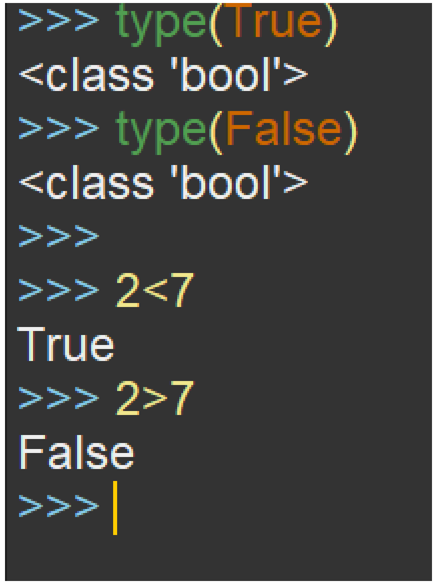
Boolean Expressions

- A Boolean is a type of value; it can only be either True or False

- Just like the integer type can take values 1, 2, 3 e.t.c; the Boolean type can take one of just two values: True or False.

- For this reason, ‘True’ and ‘False’ are Python keywords reserved for Boolean values; a variable must not be named using these words.

- Now, a Boolean expression is essentially a comparison expression that evaluates to either True or False.



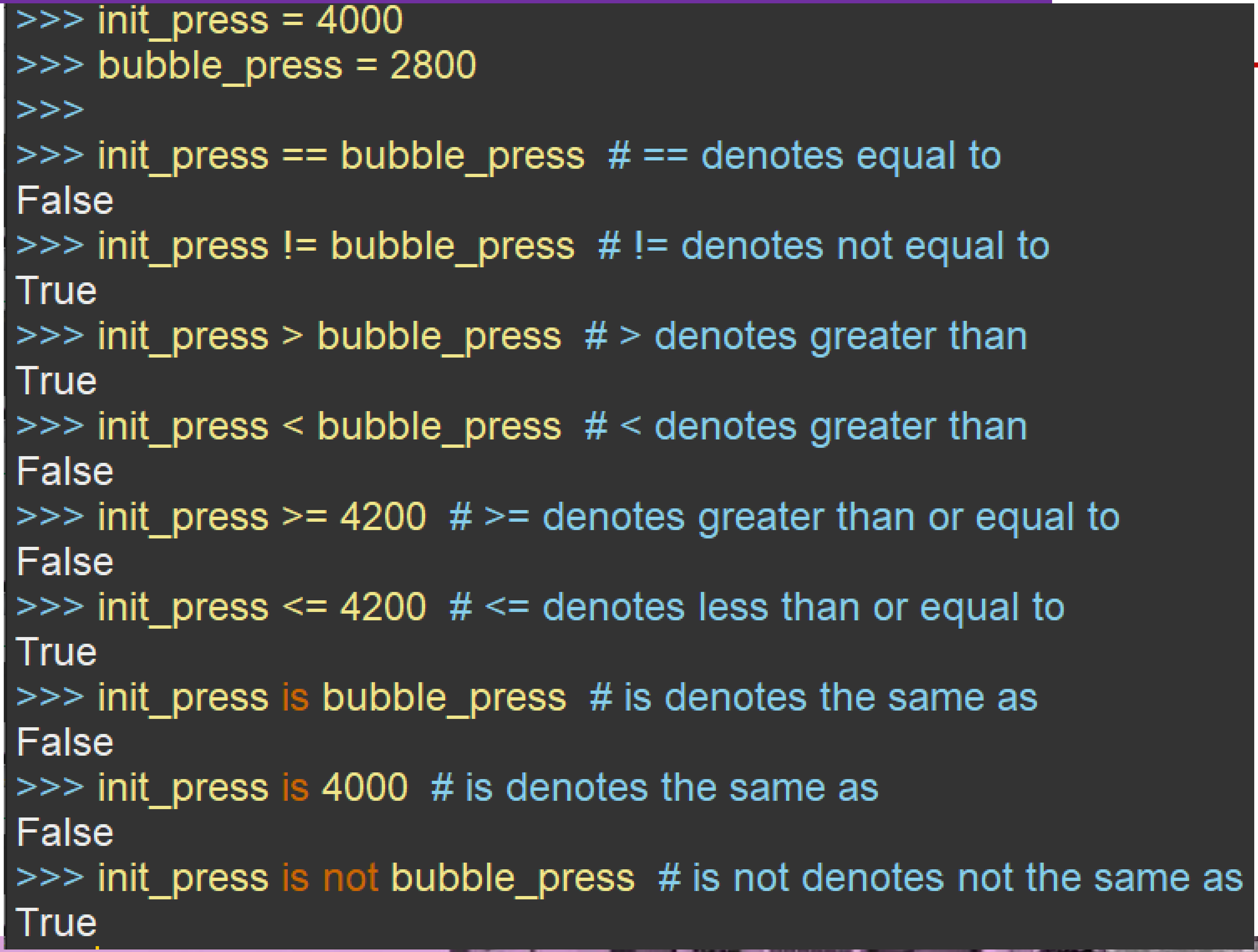
```
>>> type(True)
<class 'bool'>
>>> type(False)
<class 'bool'>
>>>
>>> 2<7
True
>>> 2>7
False
>>> 
```

Conditional Statements

Boolean Expressions

- Boolean expressions are constructed using comparison operators listed here.

- Take note that = is an assignment operator while == is a comparison operator.



```
>>> init_press = 4000
>>> bubble_press = 2800
>>>
>>> init_press == bubble_press  # == denotes equal to
False
>>> init_press != bubble_press  # != denotes not equal to
True
>>> init_press > bubble_press  # > denotes greater than
True
>>> init_press < bubble_press  # < denotes greater than
False
>>> init_press >= 4200  # >= denotes greater than or equal to
False
>>> init_press <= 4200  # <= denotes less than or equal to
True
>>> init_press is bubble_press  # is denotes the same as
False
>>> init_press is 4000  # is denotes the same as
False
>>> init_press is not bubble_press  # is not denotes not the same as
True
```

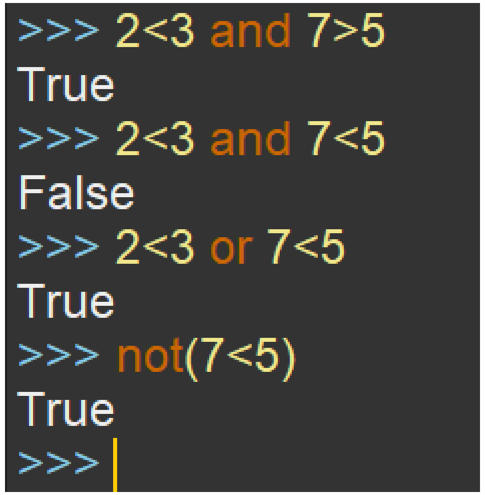
Conditional Statements

Logical Operators

- Sometimes, multiple conditions needed to be checked in a conditional statement.

- Logical operators are used to combine Boolean expressions
  - *and* returns True if all conditions are true, otherwise, False is returned.
  - *or* returns True if at least one of the conditions is true, otherwise, False is returned.

- Logical operators are also used to negate Boolean expressions
  - *not* returns True for a false condition and vice-versa

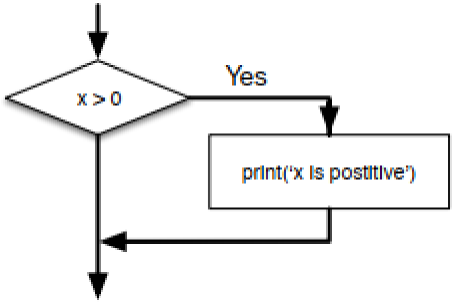


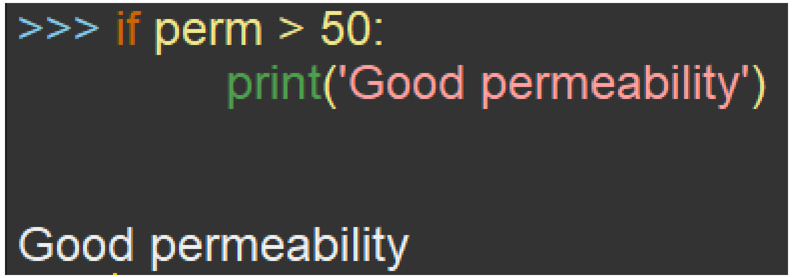
```
>>> 2<3 and 7>5
True
>>> 2<3 and 7<5
False
>>> 2<3 or 7<5
True
>>> not(7<5)
True
>>>
```

Conditional Statements

if… Statement (Conditional Execution)

- if statements evaluates the given condition; performs the given statement(s) if condition is true and skips the given statement(s) if condition is false.
- The condition(s) is written after the *if* keyword and ended with a colon i.e. (:)
- The statements to be performed or skipped are written as an indented block in subsequent line(s).
- Remove the indentation in lines after the if block.





```
>>> if perm > 50:
        print('Good permeability')


Good permeability
```

Conditional Statements

if… Statement (Conditional Execution)

Petroleum engineering application

- Computing pseudo-critical gas properties using Sutton's correlation
  - Sutton developed a correlation for estimating for estimating P_{pc} and T_{pc} as functions of gas gravity.
  - Here is the first step in Sutton's procedure:
    - If the gas mixture contains <12 mol% of CO_2, < 3% of Nitrogen and no H_2S, then the parameter γ_h takes the same value as the given gas gravity; no need for correction.
    - However, if gas mixture contains >12 mol% of CO_2 OR >3% of Nitrogen OR any H_2S, then the parameter γ_h is determined thus:

$$\gamma_h = \frac{\gamma_w - 1.1767 y_{H_2S} - 1.5196 y_{CO_2} - 0.9672 y_{N_2} - 0.622 y_{H_2O}}{1 - y_{H_2S} - y_{CO_2} - y_{N_2} - y_{H_2O}}.$$

Conditional Statements

if… Statement (Conditional Execution)

Petroleum engineering application

- Computing pseudo-critical gas properties using Sutton's correlation
  - The first step in Sutton's can be executed with an if … statement.
  - Observe that the procedure implies that if any of the impurities in the gas exceeds the stated threshold value, then, the given gas gravity (γ_w) need to be corrected for the effects of the impurities, using the given equation.
  - However, the correction task should be neglected if none of the impurities exceeds its threshold value.

Conditional Statements

if… Statement (Conditional Execution)

Petroleum engineering application

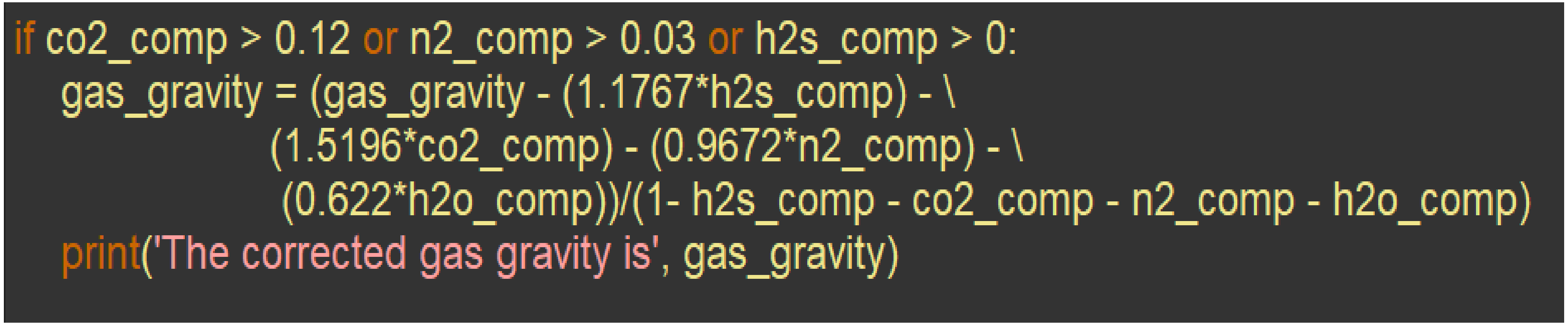
- Computing pseudo-critical gas properties using Sutton's correlation
    - To execute this procedure, we simply construct a Boolean condition to test if any threshold is violated.
    - If the condition is evaluated as True, then a block of statement to perform the gas gravity correction is executed.
    - If the condition is evaluated to be False, there is no need for the correction, hence, the block of statement is skipped.

Conditional Statements

if… Statement (Conditional Execution)

Petroleum engineering application

- Computing pseudo-critical gas properties using Sutton’s correlation



```
if co2_comp > 0.12 or n2_comp > 0.03 or h2s_comp > 0:
    gas_gravity = (gas_gravity - (1.1767*h2s_comp) - \
                  (1.5196*co2_comp) - (0.9672*n2_comp) - \
                   (0.622*h2o_comp))/(1- h2s_comp - co2_comp - n2_comp - h2o_comp)
    print('The corrected gas gravity is', gas_gravity)
```

The full script for this computation is available here

Conditional Statements

if… Statement (Conditional Execution)

Assignment 2

Upgrade the *demo_gas_grav_corr.py* script (hosted on *TTOWG/ PET328_2021_Class* GitHub repository) to perform the entire Sutton's procedure. Save the upgraded script as *sutton_correlation.py*, commit and push it to your GitHub repository. Submit the URL to your copy of *PET328_2021_Class* repository. Furthermore, send a pull request to the original *TTOWG/ PET328_2021_Class* repository.

-
- The complete Sutton's algorithm is available here.

Conditional Statements

if… Statement (Conditional Execution)

Assignment 2

You may test run your script with the following data:

- **Inputs**
  - y_{CO2} = 0.0164
  - y_{N2} = 0.0236
  - y_{H2S} = 0.1841
  - Gas gravity = 0.6992

Outputs

- Corrected gas gravity = 0.5604
- Ppch = 682.3 psia.
- Tpch = 341.8 deg Rankine
- Ppc = 799.0 psia.
- Tpc = 403.3 deg Rankine

GETTING STARTED WITH PYTHON

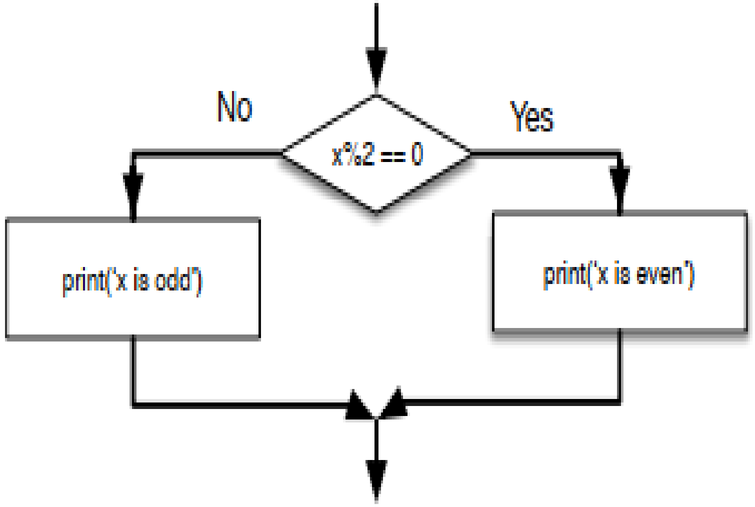
Conditional Statements

if…then…else Statement (Alternative Execution)

- The if…then…else structure is deployed when there are two alternative tasks and a condition that determines which of the two alternatives should be executed.

- Essentially, there will be a Boolean condition, and two blocks (branches) of statements.

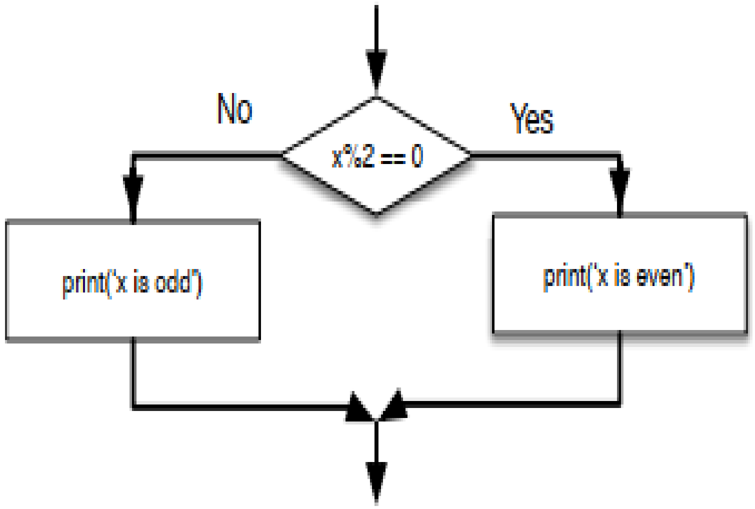
- The first branch (after the condition) is to be executed if the condition evaluates to True while the second branch (after the keyword 'else') is executed if the condition evaluates to False.

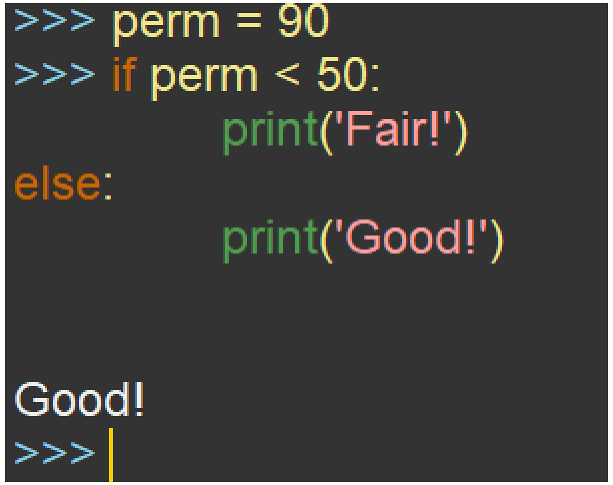


Conditional Statements

if…then…else Statement (Alternative Execution)

- The condition(s) is written after the *if* keyword and ended with a colon i.e. (:)
- Then, the statement(s) to be performed if condition is True (i.e., Branch True) are written as an indented block in subsequent line(s).
- Thereafter, the keyword 'else' is written on the next line just after the Branch True. The 'else' keyword should be indented to the same level as the 'if' keyword.
- Finally, the statement(s) to be performed if condition is False (i.e., Branch False) are written as an indented block in subsequent line(s).





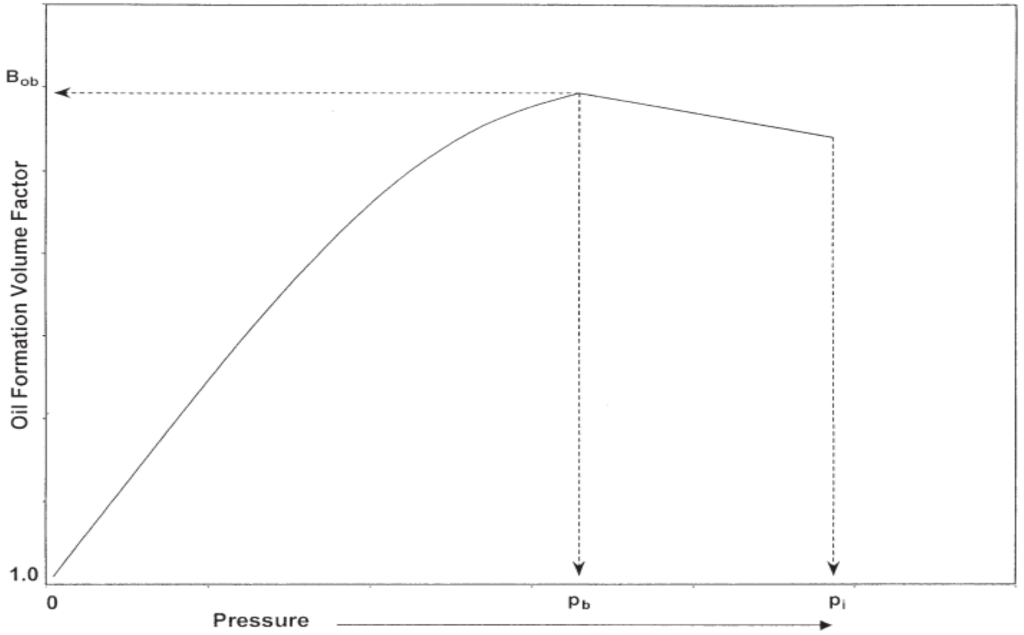
```
>>> perm = 90
>>> if perm < 50:
            print('Fair!')
else:
            print('Good!')



Good!
>>> 
```

Conditional Statements

if…then…else Statement (Alternative Execution)

Petroleum engineering application

- Computing oil formation volume factor, Bo.
  - The variation of oil formation volume factor, Bo, with pressure is divided into two pressure regimes, as shown.



Conditional Statements

if…then…else Statement (Alternative Execution)

Petroleum engineering application

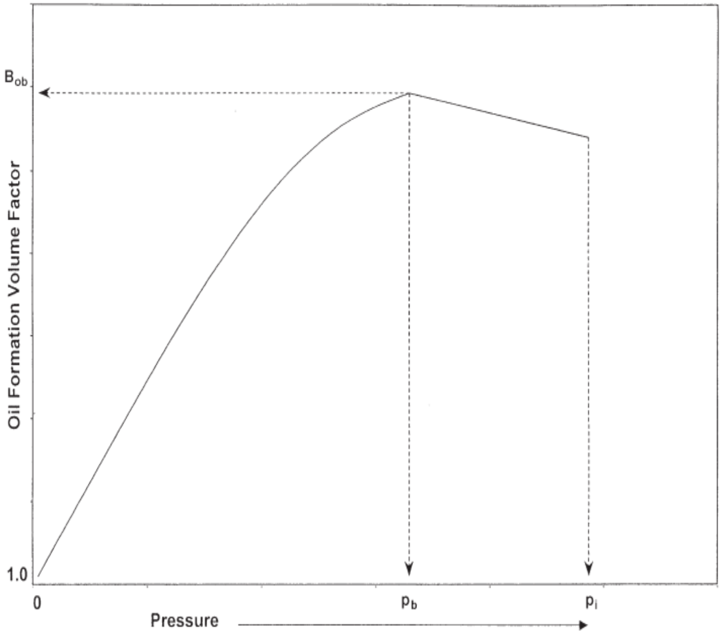
- Computing oil formation volume factor, Bo.

For pressures below or equal to bubble point, Standing's correlation for calculating Bo is herein presented:

$$B_o = 0.9759 + 0.00012F^{1.2} ----- -2.35$$

$$Where\ F = R_s\left(\frac{\gamma_g}{\gamma_o}\right)^{0.5} + 1.25\mathrm{T_F} -- -2.36$$

Note: $\mathrm{T_F}$ is temperature in degree Fahrenheit.



Conditional Statements
if…then…else Statement (Alternative Execution)

Petroleum engineering application

- Computing oil formation volume factor, Bo.

For pressure above bubble point, the analytical equation for computing Bo is given as:

$$B_o = B_{\mathrm{ob}} e^{[c_o(P_b - P)]} ---- -2.37$$

Bob is the Bo at bubble point and can be computed using Equations 2.35 and 2.36

Conditional Statements

if…then…else Statement (Alternative Execution)

Petroleum engineering application

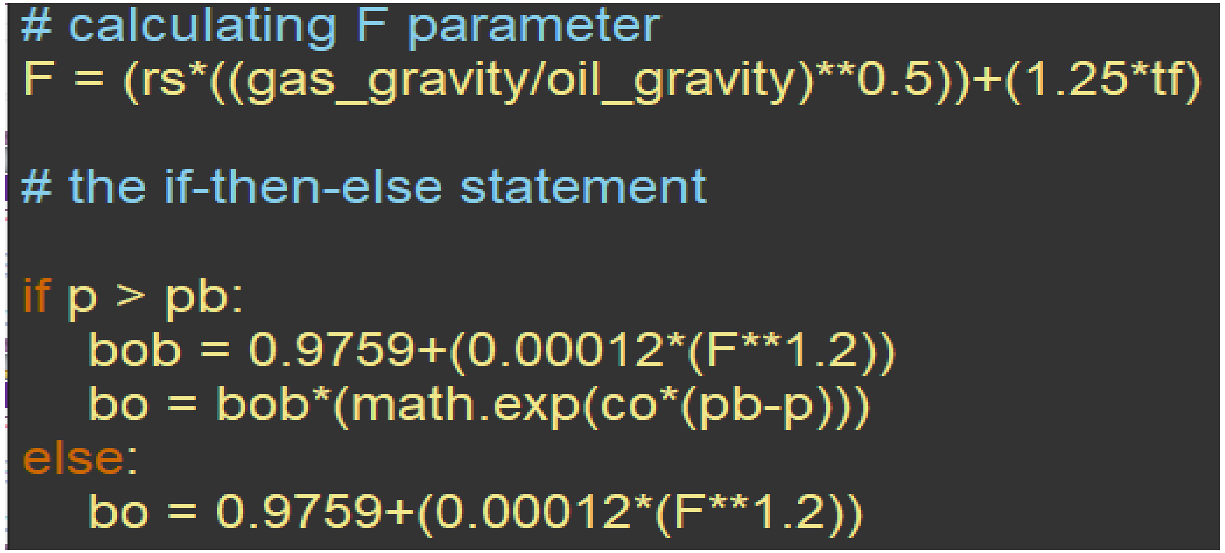
- Computing oil formation volume factor, Bo.
    - To execute this procedure, we simply construct a Boolean condition to test if the current reservoir pressure, p, is greater than the bubble-point pressure of the reservoir.
    - If the condition is evaluated as True, then a block of statement to implement Equation 2.37 is executed.
    - Else, if the condition is evaluated to be False, then a block of statement to implement Equation 2.35 is executed.
    - Note that Equation 2.36 need to be implemented for either of the alternatives, hence, the line to execute it is written before the if…then…else statement.

Conditional Statements

if…then…else Statement (Alternative Execution)

Petroleum engineering application

- Computing oil formation volume factor, Bo.



```
# calculating F parameter
F = (rs*((gas_gravity/oil_gravity)**0.5))+(1.25*tf)

# the if-then-else statement

if p > pb:
    bob = 0.9759+(0.00012*(F**1.2))
    bo = bob*(math.exp(co*(pb-p)))
else:
    bo = 0.9759+(0.00012*(F**1.2))
```

The full script for this computation is available here.

Conditional Statements

Reading Assignment

- Read on Chained Conditionals (if…elif… statements) and Nested Conditionals, in the recommended textbook for this course (pages 34 – 36)

Recommended Textbook: *Python for Everybody: Exploring Data using Python 3*, by Charles Severance.

```
>>>#TTOWG!
>>>print(‘…to the only wise God’)
```