From in class, Oct 30, 2018

% script to that demonstrates structures

% and function calls

% charles stanier

% process control

% oct 2018

% using the Seider Tate heat transfer correlation

% as an example

close all

clear variables

Pe = 15; % Peclet number

L = 10; % meters

D = 0.1; % meters

vis\_cor.docorrection = 1; % set to one to do the viscosity correction

% set to zero to skip the viscosity correction

vis\_cor.bulkvisc = 100;

vis\_cor.bulkvisc\_units = 'cm (s g)-1';

vis\_cor.wallvisc = 95;

vis\_cor.wallvisc\_units = 'cm (s g)-1';

Nu = SeiderTate( Pe, L, D, vis\_cor )

% graph of Pe from 1 to 10000

Pe\_vec = logspace( log10(1), log10(10000), 100);

for i=1:length(Pe\_vec)

Nu\_vec(i) = SeiderTate( Pe\_vec(i), L, D, vis\_cor );

end

loglog(Pe\_vec,Nu\_vec,'k\*-')

xlabel('Peclet Number');

ylabel('Nusselt Number');

% previous example demonstrated a structure with phones

% phone(1).owner = 'Charles Stanier';

% phone(1).model = 'iphone 8';

% phone(1).mass = 67;

% phone(1).mass\_units = 'grams';

%

% phone(2).owner = 'Caleigh Stanier';

% phone(2).model = 'iphone 6';

% phone(2).mass = 65;

% phone(2).mass\_units = 'gram';

%

% TM = phone\_calc( phone )

function Nu = SeiderTate( Pe, L, D, vs )

% this takes in Peclet number

% length , diameter

% viscosity structure with fields

% .docorrection (1 to enable, 0 to disable)

% .bulkvisc (viscosity along centerline)

% .bulkvisc\_units (string with units)

% .wallvisc (viscosity at wall)

% .wallvisc\_units (string with units)

% Charles Stanier

% charles-stanier@uiowa.edu

% oct 2018

% source: Welty, Wicks, Wilson Rorrer

% Fundamentals of Momentum mass and heat transfer

% edition 5

Nu\_no\_correct = 1.86\*(Pe\*D/L)^(1/3);

% note D and L assumed to be in same units

% but no error checking for this

if vs.docorrection == 1

% do viscosity correction,

% viscosity at wall vs. bulk effects result

mult=(vs.bulkvisc/vs.wallvisc)^0.14;

% error check on units

comp\_flag = strcmp( vs.bulkvisc\_units,vs.wallvisc\_units);

if comp\_flag == 1

% OK

else

% not ok

disp('viscosity units do not match');

Nu=NaN;

return;

end

Nu = Nu\_no\_correct \* mult;

else

Nu = Nu\_no\_correct;

end