

```

function [statenames ,myregion, latitudes, longitudes, fuelchosen, numsfuel] = Filter
(rawdata)
%FILTER
%
%Syntax
%[statenames ,myregion, latitudes, longitudes, fuelchosen, numsfuel] = Filter(rawtable)
%
% asks the user which region they want to
%look at, which fuel type they want, and will return values:
%[statenames(cell) ,myregion (char), latitudes(double), longitudes(double), fuelchosen
(cell), numsfuel(double)]

%Brenan Luck
%little updates By prof.Liron (**)
%Section 7
%Exam 3

%instead of deleting the first three rows, make copy of 3rd to bottom first
mytable = rawdata(3:end,:);
regions = unique(mytable(:,6)); %choose unique regions

%ask the user which region they want
regionchosen = listdlg('PromptString','Select a Region:',...
    'SelectionMode','single',...
    'ListString',regions);
myregion = regions{regionchosen};%**

%find what rows in the table equal the region chosen
rows = strcmp(myregion,mytable(:,6));
rows = find(rows == 1); %NOT NECESSARY %**

%find state names associated with rows
statenames = mytable(rows,1);

%ask what fuel type
fuels = rawdata(2,7:end);

fuelselected = listdlg('PromptString','Select a Region:','ListString',fuels);
%also chooses the column

%shows the name of the fuel selected
fuelchosen = fuels(fuelselected);

%find column of fuel
column = fuelselected+6; %INSTEAD of deleting columns 1-6

%find fuel value for region
numsfuel = cell2mat(mytable(rows, column));

%find the latitudes of desired states
latitudes = cell2mat(mytable(rows,4));

%find the longitudes of desired states
longitudes = cell2mat(mytable(rows,5));

%{
testing
[~,~,rawTable]=Filter('2009_RenewableEnergy_v02.xlsx');
[statenames ,myregion, latitudes, longitudes, fuelchosen, numsfuel] = Filter(rawTable)
%}

```