% Group # 5

% Names: Eyad Nazir, Eric Le, Jonathan Aguilar, Oscar Martinez, Jacob Cruz

% Sub-Topic: Space Technology and Missions

% House-Keeping commands

clc;clear;close all

% load in data

[spaceX1,spaceX2RAW,spaceX3] = xlsread("spacex\_launch\_data.xlsx"); % only spaceX2RAW will be used

[orbitdata1, orbitdata2, orbitdata3] = xlsread('orbitdataset.xlsx'); % all the satellite are currently active in this dataset

% [launchdata1, launchdata2, launchdata3] = xlsread('LaunchSFR.xlsx');

[SpaceVehicles1,SpaceVehicles2,SpaceVehicles3] = xlsread("SpaceVehicles.xlsx"); % only SpaceVehicles3 will be used

[database1,database2,database3] = xlsread("database.xlsx"); % only database2 will be used

%% filtering (to remove unnecessary data)

TempSpaceX = spaceX2RAW;

TempSpaceX(:,1:8) = [];

TempSpaceX(:,2) = [];

TempSpaceX(1,:) = [];

Tempdatabase = database2;

Tempdatabase(:,1:9) = [];

Tempdatabase(:,6:end) = [];

Tempdatabase(:,2:3) = [];

Tempdatabase(1,:) = [];

success\_failure\_info = [TempSpaceX;Tempdatabase]; % combing the two filtered dataset for the Successful\_Landings and Failed\_Landings functions

%% Limiting down the dataset using user criteria and then performing the needed operations and then outputting it to the command window

REPEAT = 1; % Starting a repeat variable to repeat two functions if the user wants to repeat

while REPEAT == 1 % Assuming Yes = 1 and NO = 2, so if the repeat variable is 1 it means that it will repeat again

% Selecting a daterange from a menu for the user criteria

daterange = menu('Select a decade for the rocket: ', '1950-1960','1961-1970','1971-1980','1981-1990','1991-2000','2001-2010','2011-2020');

daterange\_count = 1; % count variable that counts how many times the user exited out of the menu

% A data validation to loop if the user exits out of the menu until they exit out 3 times it terminates the program

while daterange == 0 && daterange\_count < 3

daterange = menu('Select a decade for the rocket: ', '1950-1960','1961-1970','1971-1980','1981-1990','1991-2000','2001-2010','2011-2020');

daterange\_count = daterange\_count + 1;

end

if daterange == 0

error('Too many attempts. Terminating program.')

end

% Selecting a condition (either retired or active) from a menu for the user criteria

condition = menu('Enter the condition for the rocket: ', 'Active','Retired');

condition\_count = 1;

% A data validation to loop if the user exits out of the menu until they exit out 3 times it terminates the program

while condition == 0 && condition\_count < 3

condition = menu('Enter the condition for the rocket: ', 'Active','Retired');

condition\_count = condition\_count + 1;

end

if condition == 0

error('Too many attempts. Terminating program.')

end

% dates = ['1950-1960','1961-1970','1971-1980','1981-1990','1991-2000','2001-2010','2011-2020'];

% cond = ['Active','Retired'];

% Jacob's function

[rocket\_matrix] = rocket\_filter(condition,daterange);

[R,C] = size(rocket\_matrix);

% If the critera selected does not match ay rockets this will produce a warning and prompt the user to select again

while R == 0 || C == 0

warning("The selected critea does not match any rockets. Select again.")

daterange = menu('Select a decade for the rocket: ', '1950-1960','1961-1970','1971-1980','1981-1990','1991-2000','2001-2010','2011-2020');

condition = menu('Enter the condition for the rocket: ', 'Active','Retired');

rocket\_matrix = rocket\_filter(daterange, condition);

[R,C] = size(rocket\_matrix);

end

% output rocket information (Jonathan's function)

%[country\_name, rocketnumber, rocket\_successful, percentage\_flights] = maxRocketData(rocket\_matrix);

%fprintf('%s has the most %s rockets with an amount of %0.0f. in those years: %s ', country\_name,cond(condition), rocketnumber, dates(daterange))

%fprintf('The %s rocket has %0.0f% successful flights. ', rocket\_successful, percentage\_flights)

REPEAT = menu("Would you like to repeat the program?","Yes","No"); % produce a menu to ask if the user would like to repeat

end

%% Eric's Function

orbitname = ["GEO","LEO","MEO","Elliptical"];

selectedorbit = menu('Enter a desired orbit: ','GEO','LEO','MEO','Elliptical'); % Menu to select a orbit

selectedorbit\_count = 1;

% A data validation to loop if the user exits out of the menu until they exit out 3 times it terminates the program

while selectedorbit == 0 && selectedorbit\_count < 3

selectedorbit = menu('Enter a desired orbit: ','GEO','LEO','MEO','Elliptical');

selectedorbit\_count = selectedorbit\_count + 1;

end

if selectedorbit == 0

error('Too many attempts. Terminating program.')

end

[recommendedornot, orbitcount, totalorbit] = orbit(selectedorbit, orbitdata2);

if recommendedornot == 1

selectedname = orbitname(selectedorbit);

recommendedornot = '';

fprintf('The %s orbit should %sbe used because it only appears %0.0f times on the dataset.\n', selectedname, recommendedornot, orbitcount)%will be blank

else

selectedname = orbitname(selectedorbit);

recommendedornot = 'not';

fprintf('The %s orbit should %s be used because it appears %0.0f times on the dataset.\n', selectedname, recommendedornot, orbitcount)

end

%% Eyad's Function (identify the most successful company and their most common reason of failure)

[MOSTsuccessful,num\_successful] = Successful\_Landings(success\_failure\_info);

[LEASTsuccessful, num\_failure, failure\_reason] = Failed\_landings(success\_failure\_info);

fprintf('The company with the most successful rockets is %s with %0.0f successful or controlled rocket landing outcome', char(MOSTsuccessful),num\_successful)

fprintf('\nAlthough %s is the most successful company, their most common reason for failure is due to %s. \n', char(LEASTsuccessful), char(failure\_reason))

%% Graphing

countryname = categorical({'GEO','LEO','MEO','Elliptical'}); %have to set it up as a categorical array for the bar graph

bar(countryname,totalorbit,'r');

grid on

xlabel("Orbit Names")

ylabel("Number of satellites in orbit")

title("Active satellites in specific orbits")