```matlab
% Nutrition Planner with Age and Gender Options
clc;
clear;
% Food Database (10 items)
foods = {'Apple', 'Banana', 'Chicken', 'Rice', 'Milk',
'Eggs', 'Broccoli', 'Bread', 'Yogurt', 'Fish'};
cost =     [25, 6,   220, 33,   17,  8,  50,   3,  28,  120];
protein =  [1,  1,   75,  15,   8,   6,  5,    2,  10,  45];
calories = [95, 105, 184, 650,  127, 72, 630,  75, 60,  210];
carbs =    [25, 27,  0,   140,  13,  0,  63,   14, 4,   0];
fat =      [0,  0,   2,   2,    5,   5,  30,   1,  3,   2];
fiber =    [4,  3,   0,   0,    0,   0,  24,   1,  0,   0];
% Nutrition Requirements Data
% Different Age groups: 19-30, 31-50, 51+
Data = struct();
% Male requirements
Data.M = struct();
Data.M.Young  = [56, 2200, 130, 49, 34];    % 19-30
Data.M.Adult = [56, 2200, 130, 49, 31];     % 31-50
Data.M.Senior = [56, 2000, 130, 44, 28];    % 51+
% Female requirements
Data.F = struct();
Data.F.Young  = [46, 1600, 130, 36, 28];    % 19-30
Data.F.Adult = [46, 1600, 130, 36, 25];     % 31-50
Data.F.Senior = [46, 1600, 130, 36, 22];    % 51+
% Manual User Inputs
% 'M  ' or 'F' for gender
while (1)
   gender = upper(input('\nEnter gender (M/F): ', 's'));
   if (gender=='M' || gender=='F')
       break;
   else, fprintf('Please enter M or F\n');
   end
end
age = 0;       % User's age in years
while age<18 || age>100
  age= input('Enter age(19-100): ');
end
```

```
budget=0;       %User's budget in INR
Budget = input('Enter Budget=');
while (1)
  preference = upper(input('Enter your
preference(V=veg/N=non-veg): ', 's'));
   if (preference=='V' || preference=='N')
       break;
   else, fprintf('Please enter V or N\n');
   end
end
while (1)
  LIntolerance = upper(input('Are you a Lactose
intolerant(Y=yes/N=no): ', 's'));
   if (LIntolarance=='Y' || LIntolarance=='N')
       break;
   else, fprintf('Please enter Y or N\n');
   end
end

% Determine age group
if age >= 19 && age <= 30
   ageGroup = 'Young';
elseif age >= 31 && age <= 50
   ageGroup = 'Adult';
else
   ageGroup = 'Senior';
end
% Get appropriate user information
MinNutrients = Data.(gender).(ageGroup);
numItems = length(foods);
f = cost;
% Nutrition matrix (5 nutrients × 10 foods)
nutrition = [protein; calories; carbs; fat; fiber];
nonvegIDX=[3,6,10];
LactoseIDX=[5,9];
% Serving constraints for all foods
intcon = 1:numItems;            %making sure all serving are
in integer
```

```
lb = zeros(numItems, 1);        %making sure no negative
serving allowed
ub = 4 * ones(numItems, 1);     % Maximum 5 servings per food
item
ub(3)=3;   % max 3 servings allowed , per serving = 250g
ub(4)=3;   % max 3 servings allowed , per serving = 500g
ub(5)=8;   % max 8 servings allowed , per serving = 250ml
ub(9)=3;   % max 3 servings allowed , per serving = 100g
ub(10)=3;  % max 3 servings allowed , per serving = 200g
if(preference=='V')
   ub(nonvegIDX)=0;
end
if(LIntolerance=='Y')
   ub(LactoseIDX)=0;
end
% Constraint Matrix (only minimum nutrients and budget)
A = [
   -nutrition;    % 5 rows: nutrient floors (≥ min)
   -cost          % 1 row: budget floor (≥ Budget)
];
b = [
   -MinNutrients(:); % 5 elements
   -Budget        % 1 element
];
% Solving the problem using integer linear programming ---->
A*x=b
optimal = optimoptions('intlinprog','Display','off');
[x, totalCost, exitflag,output] = intlinprog(f, intcon, A, b,
[], [], lb, ub, optimal);
Z=totalCost;
if Z>Budget     %when budget is lower than the required
nutrient contents
   fprintf('\nSince the budget is very low, The lowest
optimal budget is:\n');
end
% User Information
fprintf('\nNutrition Planner for %s, Age %d (%s)', gender,
age, ageGroup);
```

```
fprintf('\n----------------------------------------');
% Results
if exitflag > 0
   fprintf('\nOptimal Meal Plan (Cost: Rs.%.2f)\n',
totalCost);%displaying the best optimal meal plan with its
cost.
   fprintf('---------------------------------\n');
   for i = 1:numItems      %running a loop
       if x(i) > 0  % Show foods with servings > 0
           fprintf('%-12s: %.0f servings\n', foods{i}, x(i));
       end
   end

achieved = nutrition * x;   %calculating the nutrition
obtained
   % Nutrition output
   fprintf('\nNutrition Summary:\n');
   fprintf('%-12s %8s %8s %8s\n', 'Nutrient', 'Actual',
'Min');
   fprintf('\n--------------------------------------------');
   nutrients = {'Protein(g)', 'Calories(cal)', 'Carbs(g)',
'Fat(g)', 'Fiber(g)'};

   for i = 1:5
       fprintf('\n%-12s %8.1f %8.1f %12s\n', ...
           nutrients{i}, achieved(i), MinNutrients(i));
%displaying the nutrient type,its present value in meal,
minimun required nutrition
   end
end
fprintf('\n--------------------------------------------\n');
% Code for Bar Graph
figure;
bar(nutrients,[achieved, MinNutrients(:)])
title('Achieved vs Minimum Required Nutrients');
ylabel('Amount');
xlabel('Nutrients')
```

```
legend('Nutrient constituent in meal','Minimum nutrients in
meal')
grid on;
```