```
fprintf('Plotting data..\n')

%Loading data
data = load('ex31.txt');

%Create Matrix of Training Examples
X = data(:,1); % Indexing in Octave Starts from one
y = data(:,2);
m = length(y);

%Plotting data
plot(X, y, 'rx', 'MarkerSize', 10)
xlabel('Population of City in 10,000');
ylabel('Profit in $10,000s');
axis([4,24,-5,25]);


%Let's check correlation between the data
correln = corr(X,y);
fprintf('Correlation between the data is:- %.3f\n',correln);
fprintf('Program paused. Press enter to continue.\n');
pause;

%Calculating Gradient Descent
fprintf('\nRunning Gradient Descent...\n');
theta = zeros(2, 1); % initialize fitting parameters

%Setting Gradient descent settings
iterations = 15000;
alpha = 0.01;

%Cost Function
function J = cost(X, y, theta, m)
    i = 1:m;
    J = (1/(2*m)) * sum( ( (theta(1) + theta(2).*X(i,1)) - y(i)).^2 );
end

%Gradient Descent function
```

```
function theta = gradientDescent(X, y, theta, alpha, iterations, m)

    for iter = 1:iterations

        k = 1:m;
        t0 = theta(1) - (alpha/m) * sum( (theta(1) + theta(2).*X(k,1)) - y(k) );
        t1 = theta(2) - (alpha/m) * sum(((theta(1) + theta(2).*X(k,1)) - y(k) ).*X(k,1));

        theta(1) = t0;
        theta(2) = t1;
    end
end

%Compute and display initial cost
initial_cost = cost(X, y, theta, m);
fprintf('\nInitial cost before running Gradient Descent is %f\n',initial_cost);

%Gradient Descent
theta = gradientDescent(X, y, theta, alpha, iterations, m);
fprintf('\nTheta found by Gradient Descent:- %f %f\n',theta(1), theta(2));
fprintf('\nCost after completing Gradient Descent:- %f\n', cost(X, y, theta, m));
fprintf('Program paused. Press enter to continue.\n');
pause;

%Plotting the Regression line
hold on; % keep previous plot visible
plot(X(:,1), theta(1)+X*theta(2), '-');
legend('Training data','Regression Line');
hold off; % Don't plot any more plots on this figure

% Predict values for population sizes of 35,000 and 70,000
predict1 = theta(1) + 3.5*theta(2);
fprintf('\nFor population = 35,000, we predict a profit of %f\n',predict1*10000);
predict2 = theta(1) + 7*theta(2);
fprintf('For population = 70,000, we predict a profit of %f\n',predict2*10000);


```