**You need to implement Linear regression given the randomly generated data in python notebook.**

**1) def hyp(theta, X):** Implement hypothesis for equation of line.

$$h(X, \theta) = X.\theta$$

**2) def cost_function(theta,X,Y):** Implement cost function ( Mean Squared Error ) for linear regression training

$$cost = \frac{1}{2m} \sum_{i=0}^{m} (h(X^i, \theta) - Y^i)^2$$

**3)  def derivative_cost_function(theta,X,Y):** Implement Derivative of cost function (error function ) to find rate of change of error in regression

$$\frac{\delta}{\delta\theta_j} = \frac{1}{m} \sum_{i=0}^{m} (h(X^i, \theta) - Y^i) * X_j$$

**4) def GradientDescent(X,Y,cost_function,derivative_cost_function,maxniter):** Implement gradient descent algorithm to train linear regression Model given the following algorithm

```
for i in range(0, numiter):
        # hyp=hypothesis=(theta,X)
        # loss= hyp.T-Y
        # Cost = sum(loss**2)/2.0*nexamples
        # print cost
        # gradiants= loss.T . X.T / nexamples



        theta = theta - alpha * gradient
    return theta
```