Algorithm:
Our problem was how we can classify water by looking at the four features. We have four feature, around five hundreds of dataset and two classes. After conducting some research we choose KNN and Naïve Bayes classifier algorithm. As we are trying to predict tomorrow's water quality so we don't have the feature data for the next morning. Here we applied Linear Regression to solve this problem.

Naive Bayes: We used Naïve Bayes classification because of its speed and accuracy. It works on some mathematical theory, so there is no complex training method needed. It's a member of probabilistic classifier.

Our problem was classify the water from the four features. As we know it works on probabilistic theory, so we had to summarize data for faster Gaussian Probability calculation. We created two different 2D array to store data. First array to save the only safe water training data and the second one to save unsafe water data. Then we calculated the mean and Variance using these formula.

$$Mean, \qquad \mu = \frac{1}{N}\sum_{k=1}^{n} X_k$$

$$Variance = \ (\delta)^2 = \frac{1}{N-1}\sum_{k=1}^{n} (X_k - \mu)^2$$

Then we find the posterior for both of the safe and unsafe class.

$$Posterior(safe) = \frac{P(safe) * P(temperature) * P(ph) * P(turbidity) * P(EC)}{evidence}$$

$$where, P(x) = probabilty\ for\ x.$$
$$all\ the\ fetuare\ probabilty\ only\ for\ safe\ water$$

$$Posterior(unsafe) = \frac{P(unsafe) * P(temperature) * P(ph) * P(turbidity) * P(EC)}{evidence}$$

$$where, P(x) = probabilty\ for\ x.$$
$$all\ the\ fetuare\ probabilty\ only\ for\ unsafe\ water$$

$$P(safe) = P(unsafe) = 0.5$$

$$P(feature) = \frac{1}{\sqrt{2\pi\delta^2}} \exp\left(\frac{-(x-\mu)^2}{2\delta^2}\right)$$

$$evidence = P(safe) * P(temperature) * P(ph) * P(turbidity) * P(EC) + P(unsafe)$$
$$* P(temperature2) * P(ph2) * P(turbidity2) * P(EC2)$$

$$where, ph\ indicates\ the\ temperature\ for\ safe\ water\ and\ ph2\ for\ unsafe\ water.$$

After these calculations we get $Posterior(safe)$ and $Posterior(unsafe)$. If $Posterior(safe)$ is greater than $Posterior(unsafe)$ then we classify it as Safe water, otherwise it is Unsafe.

KNN: K-Nearest Neighbor, broadly known as KNN. It is a classification algorithm that classifies by checking the Euclidian distance from test to trains. This one is very straight forward algorithm.

First of all, we split the dataset into two different parts with the ratio of 70:30. 70% data for training and rest of the 30% data for testing.

For each of test instance, we find the Euclidian distance by all the features the store the distance in a vector. After finding distances from all the training instances, we sort the distances and consider closest k neighbors. Then depending on the majority it decides which class this test data belong to.

This algorithm uses very simple calculation, so it's an easy task for a normal processor and it is very fast.

$$Euclidian\ Distance\ d(a,b) = \sqrt{(a_1 - b_1)^2 + (a_2 - b_2)^2 + \cdots + (a_n - b_n)^2}$$

Linear Regression: Though our problem was a classification algorithm, but it is also possible to break it into parts which is individually a regression problem and we can then combine those result and classify it. We have the safe range for every feature.

We used linear regression to find individual value for PH, Turbidity etc. Then check if it is into the safe range or not.

Linear regression takes the training dataset and sends it to the learning algorithm and learning algorithm returns an equation.

X	Y
1	31
2	29
3	29

Table: sample table for temperature.

X is the input index that indicates the day of the year. And Y indicates the temperature on X'th day. After passing the training dataset we get an equation that is similar to

$$Y = A + BX$$

Then we can find the Y for any X. After finding the expected temperature, PH, Turbidity and EC, we check if it is the safe range or not. Then we declares it is safe or unsafe.