**Development of the algorithm**

We make use of the requests library in python to get the data from the API 30 times where the data received each time corresponds to a set of 20 coin flips. This data is then stored in a list(this makes it easy to iterate through the draws as we proceed with our algorithm)

I have considered the initial biases for heads for both the coins(A&B) to be 0.4 and 0.6 respectively.

**Expectation-Maximization Algorithm**: It is a method for finding the maximum likelihood estimates of the parameters in our system.It has two main steps

* The Expectation step:In this step we calculate the posterior probability of each data point.The posterior probability is calculated using the likelihood function.The likelihood function returns how likely a data function will belong to a particular class.The result of this step will give the expected values of the no of heads and no of tails of coin A and B respectively.
* The Maximization step:In this the values of theta which maximize the number of heads and tails is produced

The final values of theta for coin A is 0.325082200586326 and the final value of theta for

coin B is 0.6577727012615032

**Alternatives to my implementation:**

An alternative implementation to the E-M algorithm is by making use of the [sklearn.mixture](https://scikit-learn.org/stable/modules/classes.html#module-sklearn.mixture).GaussianMixture function. I chose to go with my approach over this one because in my approach one can easily understand the implementation of the algorithm while if we use the functionthe working is abstracted .