1. Consider the following mathematical function

$$f(x, \mu_0, \mu_1, \sigma_0, \sigma_1, \theta_0) = \sum_{k=0}^{1} \Phi\Big(\frac{x - \mu_k}{\sigma_k}\Big)\theta_k,$$

where $\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^{x} e^{-u^2/2} du$ is the cumulative distribution function of a standard normal distribution, and $\theta_1 = 1 - \theta_0$.

The functions $\Phi(x)$ can be computed using the `NormalDist` class from the `statistics` library. To load this into Python, use the command:
`from statistics import NormalDist`.

The function $\Phi(x)$ can then be computed using `NormalDist().cdf(x)`. For example, $\Phi(1)$ can be computed with the Python command: `NormalDist().cdf(1)`.

Write a Python function called `NormalMixDF` that computes $f(x, \mu_0, \mu_1, \sigma_0, \sigma_1, \theta_0)$. The function should have the following function definition

```
def NormalMixDF(x, mu, sigma=[0,0], theta0=0.5, log_p=False):
```

- `x` is a numeric variable
- `mu` is a list containing two numeric variables
- `sigma` is a list with two numeric elements `sigma[0]` and `sigma[1]`.
  If either `sigma[0] <= 0.001` or `sigma[1] <= 0.001`, the function should set both `sigma[0] = 0.01 + abs(mu[0]) + abs(mu[1])` and `sigma[1] = 0.01 + abs(mu[0]) + abs(mu[1])`.
- `theta0` is a numeric variable such that `0 <= theta0 <= 1`.
- `log_p` is a Boolean variable. If `log_p = False`, the function should return $f(x, \mu_0, \mu_1, \sigma_0, \sigma_1, \theta_0)$. If `log_p = True`, the function should return: $\ln\big(f(x, \mu_0, \mu_1, \sigma_0, \sigma_1, \theta_0)\big)$. (Note that you can compute the natural log ln by importing the function `log` function from the `math` library).

For example, the function call `NormalMixDF(x=0, mu=[-1.0, 1.0], sigma=[0.5, 1.0] )` should return the value 0.56795.

Check that your function works properly by running the following Python code:

```
print( NormalMixDF(x=2, mu=[0, 1.5]) )
print( NormalMixDF(x=2, mu=[0, 1.5], sigma=[1.0, 1.0]) )
print( NormalMixDF(x=2, mu=[0, 1.5], sigma=[1.0, 1.0], theta0=0.8) )
print( NormalMixDF(x=2, mu=[0, 1.5], theta0=0.8, log_p=True) )

print( NormalMixDF(x=1, mu=[-1.3, 2.5], sigma=[1.0, 1.0]) )
print( NormalMixDF(x=1, mu=[-1.3, 2.5]) )
```