1.Take a problem in hand, say we have 5000 electric motor to drive robots, out of which 999 are defective. Let us assume that we have already build a Naïve Bayes machine learning model to classify the working and defective electric motors. Wonderfully, our model identified, 1315, motors as defective, out of which 800 are truly defective and 515 are identified as defective which are

not defective. Now populate your confusion matrix correctly.

In [ ]:

```
#This Question Attached Photo
```

2.After populating the confusion matrix, define a function that calculate precision, recall, specificity, negative predicted value, true positive rate, false positive rate, average, accuracy, F1 score and Fβ score (assume β = 0.3, try to change your β from 0 to 1 and report your result). Do not create multiple function file. Submit a consolidate one function file.

In [5]:

```
def calculate_evaluation_metrics(confusion_matrix, beta=0.3):
    TP = confusion_matrix[0, 0]
    FN = confusion_matrix[0, 1]
    FP = confusion_matrix[1, 0]
    TN = confusion_matrix[1, 1]

    precision = TP / (TP + FP)
    recall = TP / (TP + FN)
    specificity = TN / (TN + FP)
    negative_predictive_value = TN / (TN + FN)

    true_positive_rate = TP / (TP + FN)
    false_positive_rate = FP / (FP + TN)

    accuracy = (TP + TN) / (TP + FN + FP + TN)
    f1_score = 2 * (precision * recall) / (precision + recall)
    f_beta_score = (1 + beta**2) * (precision * recall) / ((beta**2 * precision) + rec

    return {"Precision": precision, "Recall": recall, "Specificity": specificity,
            "Negative Predictive Value": negative_predictive_value,
            "True Positive Rate": true_positive_rate, "False Positive Rate": false_pos
            "Accuracy": accuracy, "F1 Score": f1_score, "F{0} Score".format(beta): f_b
```

In [6]:

```
import numpy as np

confusion_matrix = np.array([[800, 199], [515, 3486]])

evaluation_metrics = calculate_evaluation_metrics(confusion_matrix, beta=0.5)
print(evaluation_metrics)
```

```
{'Precision': 0.6083650190114068, 'Recall': 0.8008008008008008, 'Specificity': 0.8712
821794551362, 'Negative Predictive Value': 0.9459972862957937, 'True Positive Rate':
0.8008008008008008, 'False Positive Rate': 0.12871782054486378, 'Accuracy': 0.8572,
'F1 Score': 0.6914433880726015, 'F0.5 Score': 0.6390797251957181}
```

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]:

In [ ]: