Name- AARYAN BAIRAGI roll_no-47004

CODE-

```
import java.util.*;

public class FMeasureEMeasure
 {

   public static void main(String[] args) {
      // Sample input: Retrieved documents (Answer Set A)
      Set<String> retrievedDocs = new HashSet<>(Arrays.asList("D1", "D2", "D4", "D8"));

      // Relevant documents for query q1 (Rq1)
      Set<String> relevantDocs = new HashSet<>(Arrays.asList("D2", "D3", "D5", "D6"));

      // Calculate precision and recall
      double precision = calculatePrecision(retrievedDocs, relevantDocs);
      double recall = calculateRecall(retrievedDocs, relevantDocs);

      // Calculate harmonic mean (F1-score)
      double f1Score = calculateF1Score(precision, recall);

      // Calculate E-measure with alpha = 0.5 (balanced)
      double eMeasure = calculateEMeasure(precision, recall, 0.5);

      // Print results
      System.out.printf("Precision: %.2f\n", precision);
      System.out.printf("Recall: %.2f\n", recall);
      System.out.printf("F1-Score (Harmonic Mean): %.2f\n", f1Score);
      System.out.printf("E-Measure (alpha=0.5): %.2f\n", eMeasure);
   }
```

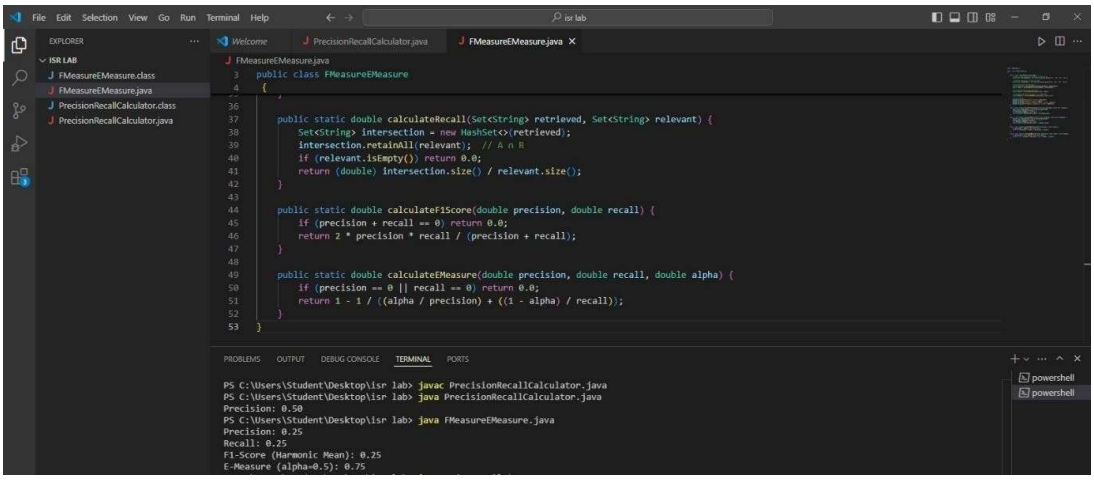
```
    public static double calculatePrecision(Set<String> retrieved, Set<String> relevant)
        { Set<String> intersection = new HashSet<>(retrieved);
        intersection.retainAll(relevant);  // A ∩ R
        if (retrieved.isEmpty()) return 0.0;
        return (double) intersection.size() / retrieved.size();
    }

    public static double calculateRecall(Set<String> retrieved, Set<String> relevant)
        { Set<String> intersection = new HashSet<>(retrieved);
        intersection.retainAll(relevant);  // A ∩ R
        if (relevant.isEmpty()) return 0.0;
        return (double) intersection.size() / relevant.size();
    }

    public static double calculateF1Score(double precision, double recall)
        { if (precision + recall == 0) return 0.0;
        return 2 * precision * recall / (precision + recall);
    }

    public static double calculateEMeasure(double precision, double recall, double alpha)
        { if (precision == 0 || recall == 0) return 0.0;
        return 1 - 1 / ((alpha / precision) + ((1 - alpha) / recall));
    }
}
```

OUTPUT:-


File Edit Selection View Go Run Terminal Help
isr lab
EXPLORER
ISR LAB
FMeasureEMeasure.class
FMeasureEMeasure.java
PrecisionRecallCalculator.class
PrecisionRecallCalculator.java
Welcome
PrecisionRecallCalculator.java
FMeasureEMeasure.java
FMeasureEMeasure.java
3 public class FMeasureEMeasure
4 {
36
37 public static double calculateRecall(Set<String> retrieved, Set<String> relevant) {
38 Set<String> intersection = new HashSet<>(retrieved);
39 intersection.retainAll(relevant); // A ∩ R
40 if (relevant.isEmpty()) return 0.0;
41 return (double) intersection.size() / relevant.size();
42 }
43
44 public static double calculateF1Score(double precision, double recall) {
45 if (precision + recall == 0) return 0.0;
46 return 2 * precision * recall / (precision + recall);
47 }
48
49 public static double calculateEMeasure(double precision, double recall, double alpha) {
50 if (precision == 0 || recall == 0) return 0.0;
51 return 1 - 1 / ((alpha / precision) + ((1 - alpha) / recall));
52 }
53 }
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\Student\Desktop\isr lab> javac PrecisionRecallCalculator.java
PS C:\Users\Student\Desktop\isr lab> java PrecisionRecallCalculator.java
Precision: 0.50
PS C:\Users\Student\Desktop\isr lab> java FMeasureEMeasure.java
Precision: 0.25
Recall: 0.25
F1-Score (Harmonic Mean): 0.25
E-Measure (alpha=0.5): 0.75
powershell
powershell