Main.java

```
import java.util.ArrayList;

public class main {
    public static void main(String[] args) {
        ArrayList<Pair<Character, Double>> a = new ArrayList<>();
        a.add(new Pair<>('0', 0.0));
        a.add(new Pair<>('a', 0.0));
        a.add(new Pair<>('5', 0.0));
        a.add(new Pair<>('x', 0.0));
        CreateFiles files = new CreateFiles(a);

        Shannon s = new Shannon();
    }
}
```

Pair.java

```
public class Pair<T, V> {
    private T first;
    private V second;


    Pair(T a, V b){
        this.first = a;
        this.second = b;
    }

    public void setFirst(T first) {
        this.first = first;
    }

    public void setSecond(V second) {
        this.second = second;
    }

    public T getFirst() {
        return first;
    }

    public V getSecond() {
        return second;
    }
}
```

CreateFiles.java

```
import java.io.FileWriter;
import java.io.IOException;
import java.util.ArrayList;
import java.util.Comparator;
```

```
public class CreateFiles {
    private ArrayList<Pair<Character, Double>> alphabet;
    private final ArrayList<Double> P1 = new ArrayList<>();
    private final ArrayList<Double> P2 = new ArrayList<>();



    public CreateFiles(ArrayList<Pair<Character, Double>> A){
        try {
            FileWriter f1 = new FileWriter("F1.txt", false);
            FileWriter f2 = new FileWriter("F2.txt", false);
            this.alphabet = A;

            System.out.println("F1:");
            this.GenerateF1();
            this.Print();
            this.Write(f1);
            f1.close();
            this.alphabet.remove(this.alphabet.size() - 1);

            System.out.println("Theory");
            this.plogp(1, this.P1);
            ArrayList<Double> H1 = this.PaPb(this.P1, this.P1);
            this.plogp(2, H1);
            ArrayList<Double> H2 = this.PaPb(H1, this.P1);
            this.plogp(3, H2);

            System.out.println("F2:");
            this.GenerateF2();
            this.Print();
            this.Write(f2);

            System.out.println("Theory");
            this.plogp(1, this.P2);
            H1 = this.PaPb(this.P2, this.P2);
            this.plogp(2, H1);
            H2 = this.PaPb(H1, this.P2);
            this.plogp(3, H2);

        } catch (IOException e) {
            e.printStackTrace();
        }

    }

    private void Write(FileWriter File) {
        for(int i = 0; i < 11000; ++i){
            double rand = Math.random();
            for(int j = 1; j < this.alphabet.size(); ++j){
                if(rand >= this.alphabet.get(j - 1).getSecond() && rand <
this.alphabet.get(j).getSecond()){
                    try {
                        File.write(this.alphabet.get(j - 1).getFirst());
                    } catch (IOException e) {
                        e.printStackTrace();
                    }
                }
            }
        }
        try {
            File.flush();
        } catch (IOException e) {
            e.printStackTrace();
        }
```

```
    }

    private void Print() {
    }

    void GenerateF1() {
        double p = 0.0;
        for(int i = 0; i < this.alphabet.size(); ++i){
            this.alphabet.get(i).setSecond(p);
            this.P1.add(((double) (1) / this.alphabet.size()));
            p += (double) (1) / this.alphabet.size();
        }
        this.alphabet.add(new Pair<>(' ', 1d));
    }


    void GenerateF2() {
        ArrayList<Integer> num = new ArrayList<>();
        int sum = 0, tmp;
        for(int i = 0; i < this.alphabet.size(); ++i){
            tmp = (int) (Math.random() * 100);
            num.add(tmp);
            sum += tmp;
        }
        double p = 0.0;
        for(int i = 0; i < this.alphabet.size(); ++i){
            this.alphabet.get(i).setSecond(p);
            this.P2.add((double) (num.get(i)) / sum);
            p += P2.get(i);

        }
        this.alphabet.add(new Pair<>(' ', 1d));
        this.alphabet.sort(Comparator.comparing(Pair::getSecond));
    }
    void plogp(int length, ArrayList<Double> P) {
        double H = 0;
        for (Double a : P) {
            H += (-1) * (a * Math.log10(a) / Math.log10(2));

        }

        H /= length;

        System.out.println("H" + length + " = " + H);
    }
    ArrayList<Double> PaPb(ArrayList<Double> P, ArrayList<Double> P1){
        ArrayList<Double> H = new ArrayList<>();

        for(Double i : P)
            for(Double j : P1)
                H.add(i * j);
        return H;
    }

}
```

Shannon.java

```
import java.io.FileReader;
import java.io.IOException;
import java.util.ArrayList;
import java.util.Scanner;
```

```
public class Shannon {
    private FileReader F1, F2;
    ArrayList<Pair<String, Double>> P1 = new ArrayList<>();

    Shannon(){
        try {
            this.F1 = new FileReader("F1.txt");
            this.F2 = new FileReader("F2.txt");

            System.out.println("From first file:");
            this.H(1);
            System.out.println("From second file:");
            this.H(2);

        }catch (IOException ex){
            ex.printStackTrace();
        }
    }

    private void H(int flag) throws IOException {
        String buff = "";
        if(flag == 1){
            Scanner scan = new Scanner(this.F1);
            buff = scan.nextLine();
            F1.close();
        }else if(flag == 2){
            Scanner scan = new Scanner(this.F2);
            buff = scan.nextLine();
            F2.close();
        }
        int j, k;
        for(int d = 0; d < 2; ++d){
            StringBuilder s = new StringBuilder();
            for(k = 0; k < d; ++k){
                s.append(buff.charAt(k));
            }
            for(int i = k; i < buff.length(); ++i){
                s.append(buff.charAt(i));
                for(j = 0; j < this.P1.size(); j++){
                    if(this.P1.get(j).getFirst().equals(s.toString())){
                        Double tmp = this.P1.get(j).getSecond();
                        this.P1.get(j).setSecond(++tmp);
                        break;
                    }
                }
                if(j == this.P1.size()){
                    this.P1.add(new Pair<>(s.toString(), 1d));
                }
                s.delete(0, 1);
            }

            Double sum = 0d;
            for(var i : P1){
                sum += i.getSecond();
            }
            for (Pair<String, Double> stringDoublePair : P1) {
                stringDoublePair.setSecond(stringDoublePair.getSecond() /
sum);

            this.plogp(d + 1);
            this.P1.clear();
        }
    }
```

```
    private void plogp(int length) {
        double H = 0d;

        for (var a : this.P1)
            H += (-1) * (a.getSecond() * Math.log10(a.getSecond()) /
Math.log10(2));

        H /= length;

        System.out.println("H " + length + " = " + H);
    }
}
```