Parsing lines in java: line.split(delimiter)

```
import java.io.*;

public static void main(String [] args) throws Exception
{
        //loop through the file and parse tokens
        BufferedReader bufRead = new BufferedReader
                                                (new FileReader(args[0]));
        String line;    // String that holds current file line

        while ((line=bufRead.readLine()) != null)
        {
                String [] parsed=line.split(",");
                double [] dataRecord=new double[parsed.length];
                for(int i=0;i<parsed.length;i++)
                        dataRecord[i]=Double.parseDouble(parsed[i]);

        }
        bufReader.close();
}
```

Parsing lines in java: StringTokenizer (line, delimiter)

```
import java.io.*;
import java.util.*;

public static void main(String [] args) throws Exception
{
        //loop through the file and parse tokens
        BufferedReader bufRead = new BufferedReader(new
        FileReader(args[0]));
        String line;    // String that holds current file line

        while ((line=bufRead.readLine()) != null)
        {
                StringTokenizer st=new StringTokenizer(line,",");
                while(st.hasMoreTokens())
                        System.out.println(st.nextToken());
        }
        bufRead.close();
}
```

How to convert tokens into WEKA Instances: header

```
import weka.core.*;

Attribute num1 = new Attribute("age");
Attribute num2 = new Attribute("income");

FastVector labels = new FastVector();
labels.addElement("no");  //0
labels.addElement("yes"); //1
Attribute cls = new Attribute("class", labels);

FastVector attributes = new FastVector();
attributes.addElement(num1);
attributes.addElement(num2);
attributes.addElement(cls);

Instances dataset = new Instances("Gold dataset", attributes, 0);
```

How to convert tokens into WEKA Instances: single data record

```
while ((line=bufRead.readLine()) != null)
{
        String [] parsed=line.split(",");
        double [] dataRecord=new double[parsed.length];
        int i=0;
        for(;i<parsed.length-1;i++)
                dataRecord[i]=Double.parseDouble(parsed[i]);

        //last value is a class
        if(parsed[i].trim().equalsIgnoreCase("no"))
                dataRecord[i]=0;
        else
                dataRecord[i]=1;
}
dataset.add(new Instance(1,dataRecord));
```

How to open native .arff input

```
import weka.core.Instances;
import java.io.*;

public class LoadArff
{
        public static void main(String [] args) throws Exception
        {
                //1. ****************************
                //load arff training set

                BufferedReader reader = new BufferedReader(
                                                        new FileReader(args[0]));
                Instances data = new Instances(reader);
                reader.close();
                // setting class attribute
                data.setClassIndex(data.numAttributes() - 1);
        }
}
```