Summary

Problem Statement

With the enormous number of languages and file types used for writing logical source or for data purposes, it is very important for a product like BlueOptima to effectively identify and categorize a file into its type. And this has to be done solely based on Extension and Name of the file itself.

This work sample requires you to identify different sources that could be used to identify details of a file type like following (but not limited to)

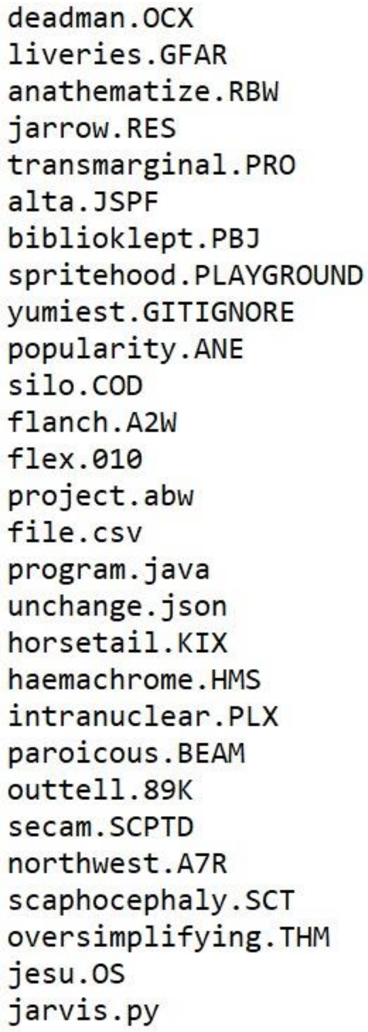
1. Short Description (explaining the usage of the file type)
2. Category (i.e. Logical Source, Configuration, Data, etc.)
3. Language Family (Java, Python, Perl, etc.)
4. Programming Paradigm (Procedural, OOP, Dynamic, etc.)
5. Associated applications

Execution Flow

- Extracted (Web scrapping) data from **File-Extensions.org** using jsoup and stored in database using **dataBaseInsetion.java** and implemented in **FileExtension.java**.
- Extract MIME of extensions using **Apache Tika**.
- Extracted (Web Scrapping) data alphabetically from **FileInfo.com** using jsoup and stored in database using **dataBaseInsetion.java** and implemented in **FileInfo.java**.
- Created **input.txt** file for passing input,
- Created **output.txt** to store the output.
- Implementation of code starts from **FileExtensionInfo.java**.
- Reading the file input.txt line by line and getting extensions.
- We have extracted extension of file in class **GetExtension** using its function findExtension.
- Now we use the class **getInfoFromDatabase.java** to get all data related to that extension.
- At last we print all the information in proper format.
- After code is successfully implemented the message “Success u can check result in output.txt” is shown.

Input

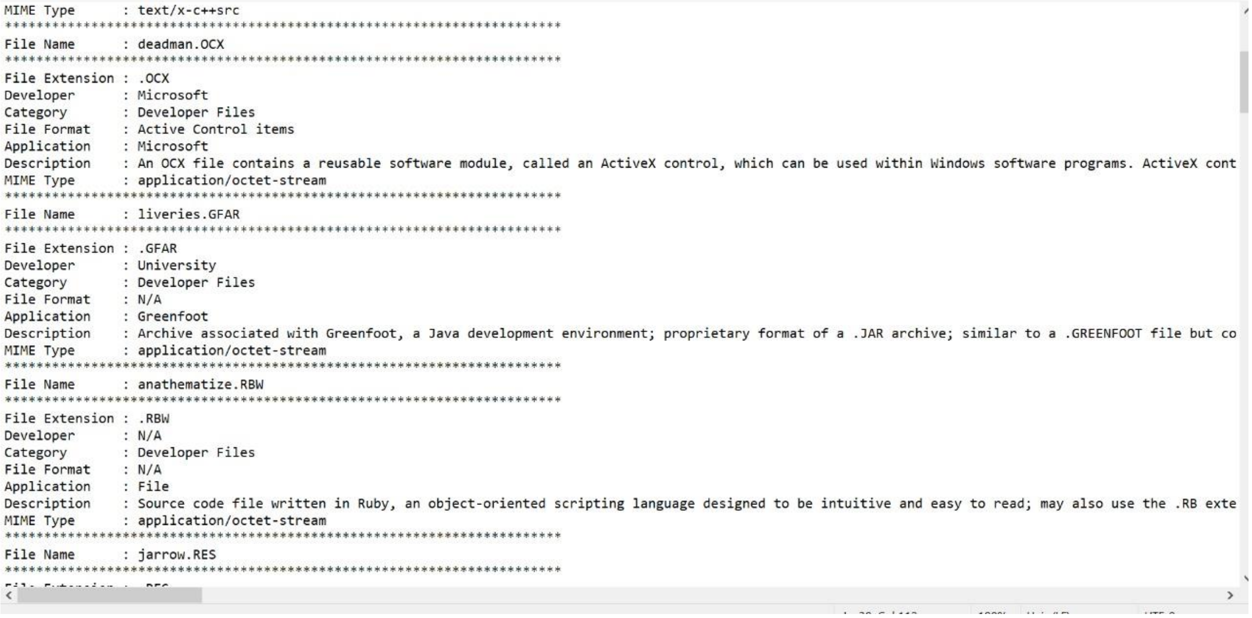
Input is taken from input.txt. It contains various file names with extensions. It can be found in input directory of D-fileTypeIdentifiction.



```
deadman.OCX
liveries.GFAR
anathematize.RBW
jarrow.RES
transmarginal.PRO
alta.JSPF
biblioklept.PBJ
spritehood.PLAYGROUND
yumiest.GITIGNORE
popularity.ANE
silo.COD
flanch.A2W
flex.010
project.abw
file.csv
program.java
unchange.json
horsetail.KIX
haemachrome.HMS
intranuclear.PLX
paroicous.BEAM
outtell.89K
secam.SCPTD
northwest.A7R
scaphocephaly.SCT
oversimplifying.THM
jesu.OS
jarvis.py
```

Output

Output is taken from output.txt. It contains data related to extension of files taken from input.txt. It can be found in output directory of D-fileTypeIdentifiction.



```
MIME Type      : text/x-c++src
***********************************************************************
File Name      : deadman.OCX
***********************************************************************
File Extension : .OCX
Developer      : Microsoft
Category       : Developer Files
File Format    : Active Control items
Application    : Microsoft
Description    : An OCX file contains a reusable software module, called an ActiveX control, which can be used within Windows software programs. ActiveX cont
MIME Type      : application/octet-stream
***********************************************************************
File Name      : liveries.GFAR
***********************************************************************
File Extension : .GFAR
Developer      : University
Category       : Developer Files
File Format    : N/A
Application    : Greenfoot
Description    : Archive associated with Greenfoot, a Java development environment; proprietary format of a .JAR archive; similar to a .GREENFOOT file but co
MIME Type      : application/octet-stream
***********************************************************************
File Name      : anathematize.RBW
***********************************************************************
File Extension : .RBW
Developer      : N/A
Category       : Developer Files
File Format    : N/A
Application    : File
Description    : Source code file written in Ruby, an object-oriented scripting language designed to be intuitive and easy to read; may also use the .RB exte
MIME Type      : application/octet-stream
***********************************************************************
File Name      : jarrow.RES
***********************************************************************
```

Steps to run the program

- In /input/ create your own input file or use the provided one.
- Execute the main program: *src/extensionInfo/FileExtensionInfo.java*.
- Enter the root and password of your database to establish connection.
- Check the output in output.txt.

Developers

- Shauvik Pujari
- Khushi