Filetype Identification,

Version: 1.0 R1

Released: Monday, October 16, 2017

Team Size: 3 Developers

Duration: 3-5 Days (full time)

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

Deliverable 1:

Identify relevant data sources from where a filetype information (as described above) can be extracted based on filename or file extension. List at least 5 relevant sources and explain the rationale on why it should be used.

Deliverable 2:

Write a program to fetch the required information from at least 3 (ideally) of the identified sources. This program should take an input of filenames or extensions, as by your logic and generate the output in a readable format for the passed list.

The input and output formats should be designed accordingly and should also take into consideration usability in mind.

Requirements:

1. Solution should be able to handle large volumes for filetypes. I.e. the Solution could be required to parse and get information for 10 to 2000 filetypes.
2. The data collected should be merged in ways so that it provides the user with the most meaningful results which are easy to understand to perform analysis on.
3. The solution needs to be reasonably fast and multi-threading be used.

Evaluation:

Your submission will be evaluated on both deliverables. While deliverable 1 will be used to understand your research and analysis skills, deliverable 2 will be used to understand your coding abilities. Once the submission is made, we will arrange a follow up discussion to understand from you the solution itself and ask any question that come out of the submission you make.

Important: To make evaluation accurate and less time consuming please take note of following.

1. **Code formatting:** Please indent the code (using tabs or space) and use camel case when defining variables and functions.

2. **Documentation:** Please document the code where necessary and document it just enough. Excessive documentation is worse than no documentation.
3. **Overall summary:** Please explain on how to build and execute the code. How and what inputs to be passed and where output can be seen. A general execution flow will also be good for the evaluator to understand the solution that is being provided. As mentioned above, this needs to be just enough to make evaluation effective and avoid excessive documentation.