Assignment – 3(part-A) Python

End Date: **15 July 9PM**

- This assignment is an individual submission.
- This assignment consists of two parts – A and B. This is A.
- Total 100 Marks with duration of 1 week.
- All submissions should be done as per instructions.
- All other conditions are open to your interpretations.

Given a file ***rules.json*** with rules written in the following format:

```json
[
  {
    "fields": {
      "profession": "Healthcare",
      "travel": "No",
      "symptomatic": "No"
    },
    "results": "High Risk"
  },
  {
    "fields": {
      "profession": "Healthcare",
      "travel": "No",
      "symptomatic": "Yes"
    },
    "results": "High Risk"
  },
  {
    "fields": {
      "profession": "Healthcare",
      "travel": "Yes",
      "symptomatic": "Yes"
    },
    "results": "High Risk"
  },
  {
    "fields": {
      "profession": "IT",
      "travel": "No",
      "symptomatic": "No"
    },
    "results": "Low Risk"
  },
  {
    "fields": {
      "profession": "IT",
      "travel": "Yes",
      "symptomatic": "No"
    },
    "results": "Moderate Risk"
  },
  {
```

```
    "fields": {
      "profession": "IT",
      "travel": "Yes",
      "symptomatic": "Yes"
    },
    "results": "High Risk"
  },
  {
    "fields": {
      "profession": "Police",
      "travel": "No",
      "symptomatic": "No"
    },
    "results": "Moderate Risk"
  },
  {
    "fields": {
      "profession": "Police",
      "travel": "Yes",
      "symptomatic": "No"
    },
    "results": "High Risk"
  },
  {
    "fields": {
      "profession": "Police",
      "travel": "Yes",
      "symptomatic": "Yes"
    },
    "results": "High Risk"
  }
]
```

And a file *data.csv* that contains multiple records in the following format (Email, Profession, Travel, Symptomatic, Chronic):

Email	Profession	Travel	Symptomatic	Chronic
a@b.com	Healthcare	No	No	No
x@y.com	IT	Yes	No	No

Based on the rules and the incoming data from data file, create a new file *results.csv* in the following schema:

Email	Profession	Travel	Symptomatic	Chronic	Results
a@b.com	Healthcare	No	No	No	High Risk
x@y.com	IT	Yes	No	No	Moderate Risk

Result is derived based on the values of the fields mentioned in the rules.

Instructions

1. *rules.json* is a static JSON file. That means for the submitted code, file content will not change. Rule list is exhaustive. Incoming data in *data.csv* will always have combination found in Rule JSON File

2. File format for Data file and New file with Result tag will be Comma Separated.

3. Ensure you lookup the rules file, match the incoming record data with right tag (as per the rules) and then insert the data into *results.csv* .
   For example
   data.csv has three columns (*Profession, Travel, Symptomatic*). Rules Tag is derived from these three keys from *rules.json* file. Once you get the tag, insert the Data record along with new Result column in *results.csv* file.

4. HINTS:
   a. You can create separate logic for deriving rules from JSON file and do look up on the rules when inserting data.
   b. You are free to generate any intermediary files.
   c. You are free to use any python libraries. (supplied with standard python installation)

Grading Criteria:

   a. Able to **read** from RULES JSON file and **write** appropriate tag into Results file. (**20**)
   b. Submit a Flow chart for code logic (**10**)
   c. Accuracy of Result tag is 100%. Evaluators will test the code on a different Data file to evaluate the accuracy of Result's column. Students are required to create their own data file, however schema of data file should be unchanged (**25**)
   d. Code is properly commented. Comments are inputted **#.** (**5**)
   e. Rule logic to derive Tags from RULES JSON (**25**).
      If you use only If and else to derive logic – **12.5 marks**
      OR If you use Dictionary along with if and else – **25 marks**
      Else 0
   f. Moodle submission format (**5**):

      <roll-number>.zip

      ------- lookup.py

      ------- rules.py *(Code to derive Rules from JSON File)*

      ------- results.csv *(Code to lookup on the rules and insert into Result's file)*

      ------- flow_chart (png/jpeg/jpg)

Note:

It is recommended that the code is flexible to accommodate change in specs in the future assignments. This is however not a criterion for grading.