

## TASK

=====

Time to complete : 3 days

### JSON Validator

Validate a Json against a given schema file

Create a class called JsonValidator .The class should be able to validate:

1. required fields. Example: id, name field to be declared as they are mandatory
2. atleast one of many fields to be present. Example: one of home phone or cell phone or work phone fields
3. either one field or another field Either birth date or govt id number
4. mutually exclusive fields (if one is present, the other should not be present)
5. field value to be one of a set of values. Example: field day can have only one of SU,MO,TU,WE,TH,FR,SA (enum)

Define the following methods

1. validate\_schema(json\_file, schema\_file)  
method should return True or False if validation succeeds or fails

=====

Coding Standard: Please follow PEP8 coding standards.

Doc strings for functions should be mentioned as below:

=====

Example:

```
def send_metric(
    self,
    metric_name: str,
    datapoint: Union[float, int],
    tags: Optional[List[str]] = None,
    type_: Optional[str] = None,
    interval: Optional[int] = None,
) -> Dict[str, Any]:
    """
    Sends a single datapoint metric to DataDog
    :param metric_name: The name of the metric
    :type metric_name: str
```

```
:param datapoint: A single integer or float related to the metric
:type datapoint: int or float
:param tags: A list of tags associated with the metric
:type tags: list
:param type_: Type of your metric: gauge, rate, or count
:type type_: str
:param interval: If the type of the metric is rate or count, define the corresponding interval
:type interval: int
"""
```