**f# T-REX – Full Stack Developer Exercise**

**Solar irradiance** is the measure of average solar energy hitting the surface of the

earth for some location over some period of time, usually over a month.

You are asked to build a **server-based utility** for fetching monthly solar- irradiance data for some set of coordinates in the US, using the [NREL Solar Resource Data API](http://developer.nrel.gov/docs/solar/solar-resource-v1/).

<https://developer.nrel.gov/docs/solar/solar-resource-v1/>

The test objective is to scan **Solar irradiance** measures from predefined states in the US and enable them to perform analytics and notifications across the data.

The utility will run on the server as a CLI interface to be written with a code language that you choose (**pre define**).

**Step 1**

**Configurable site management**:

The utility gives the user the option to define in configuration file:

The sites we wish to track:

Each site contains the **attributes** : ‘site name’ , ‘longitude’ , ‘latitude’

**Step 2**

**Fetch data**

You should fetch the 3 metric : avg\_dni , avg\_ghi , avg\_lat\_tilt from [NREL Solar Resource Data API](http://developer.nrel.gov/docs/solar/solar-resource-v1/).

And print them to the screen for each site that define under the configuration

Example for the API response:

// 20190704101023

//

https://developer.nrel.gov/api/solar/solar\_resource/v1.json?api\_key=DEMO\_KEY&lat=40&lon=-105

{

"version": "1.0.0",

"warnings": [

],

"errors": [

],

"metadata": {

"sources": [

"Perez-SUNY/NREL, 2012"

]

},

"inputs": {

"lat": "40",

"lon": "-105"

},

"outputs": {

"avg\_dni": {

"annual": 6.06,

"monthly": {

"jan": 5.0,

"feb": 5.34,

"mar": 5.94,

"apr": 6.11,

"may": 6.36,

"jun": 7.43,

"jul": 7.48,

"aug": 6.65,

"sep": 6.81,

"oct": 5.82,

"nov": 5.11,

"dec": 4.67

}

},

"avg\_ghi": {

"annual": 4.81,

"monthly": {

"jan": 2.5,

"feb": 3.43,

"mar": 4.69,

"apr": 5.69,

"may": 6.6,

"jun": 7.25,

"jul": 7.14,

"aug": 6.24,

"sep": 5.35,

"oct": 3.85,

"nov": 2.75,

"dec": 2.19

}

},

"avg\_lat\_tilt": {

"annual": 5.82,

"monthly": {

"jan": 4.79,

"feb": 5.4,

"mar": 6.07,

"apr": 6.11,

"may": 6.25,

"jun": 6.47,

"jul": 6.58,

"aug": 6.44,

"sep": 6.53,

"oct": 5.71,

"nov": 4.99,

"dec": 4.47

}

}

}

}

**Step 3**

**Analytics**

|  |  |  |  |
| --- | --- | --- | --- |
| **#** | **Type** | **Metric** | **args** |
| 1 | Max in months | avg\_dni | months=[Jan,Feb] |
| 2 | Min annually | avg\_ghi | None |

Your task is to design a solution that will support both analytics (any many more to come in the future)

you can choose only one of the analytics to implement

\*Although there are only 2 example here you should build it in a way it can scale to support **many kinds of analytics**

**Step 4**

**Notification**

Your task is to implement a notification system that will alert if a condition match your definition:

Notification example:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **#** | **Type** | **On** | **Metric** | **Months** | **Action** | **args** |
| 1 | sum\_greater\_then\_notfication | Colorado | avg\_dni | [jan,oct] | cli\_print | greater\_then=10 |
| 2 | sum\_less\_then\_notfication | all sites | avg\_ghi | [jan,oct,nov] | send\_slack | less\_then=12 |

1.In this first example the user will get a **CLI alert** if the accumulate avg\_dni in Colorado on jan,oct will be greater than 10.

2.on this second example the user will get a **SLACK alert** if the accumulate avg\_ghi in all defined sites on an,oct,no will be less than 12

Your task is to design and implement a solution for this kind of notifications.

\*Although there are only 2 example here you should build it in a way it can scale to support many **actions type , and many kinds of notifications**

\*for this specific implementation you do not need to implement the slack alert

Good luck