Coding Test

Test 1: .Net Core/Spring Boot:

Create a .Net Core REST API, which accepts the below array as JSON request and perform the following tasks

1. Create directory “Users”, If not exist at specified path in a config file
    a. Create Sub Directory “IN”.
2. Store the Request as JSON file in “IN” directory” created above

JSON Request:


```
[
    {
        "ID": 64,
        "UserID": 7,
        "EmployeeID": "CLGAXO",
        "SiteName": "MULGRAVE",
        "BusinessUnitName": "Telstra Logistics - Melbourne",
        "AccountName": "IBM AUSTRALIA LTD",
        "GroupName": "Transport",
        "CategoryName": "Activity - Productive",
        "TypeName": "Transport - Freight Sorting",
        "Date": "2018-02-14",
        "Duration": "00:30",
        "IsProcessed": false

    },
    {
        "ID": 66,
        "UserID": 7,
        "EmployeeID": "CLGAXO",
        "SiteName": "MULGRAVE",
        "BusinessUnitName": "Telstra Logistics - Melbourne",
        "AccountName": "IBM AUSTRALIA LTD",
        "GroupName": "Picking",
        "CategoryName": "Activity - Productive",
        "TypeName": "Picking - Bulk",
        "Date": "2018-02-15",
        "Duration": "00:30",
        "IsProcessed": false
    }
]
```

Coding Test

Test 2: JavaScript:

Create a new object from the “activity[]” below with the Employee ID as the property key (Group by Employee ID) with an array of activities for each employee. Output should be a 2D array. Avoid using for loops to iterate the array when building this new array object.

For this you can submit your response via. https://stackblitz.com/ and send us the url

```
var activity=[
    {
        "ID": 64,
        "UserID": 7,
        "EmployeeID": "CLGAXO",
        "SiteName": "MULGRAVE",
        "BusinessUnitName": "Telstra Logistics - Melbourne",
        "AccountName": "IBM AUSTRALIA LTD",
        "GroupName": "Transport",
        "CategoryName": "Activity - Productive",
        "TypeName": "Transport - Freight Sorting",
        "Date": "2018-02-14",
        "Duration": "00:30",
        "IsProcessed": false

    },
    {
        "ID": 66,
        "UserID": 7,
        "EmployeeID": "CLGAXO",
        "SiteName": "MULGRAVE",
        "BusinessUnitName": "Telstra Logistics - Melbourne",
        "AccountName": "IBM AUSTRALIA LTD",
        "GroupName": "Picking",
        "CategoryName": "Activity - Productive",
        "TypeName": "Picking - Bulk",
        "Date": "2018-02-15",
        "Duration": "00:30",
        "IsProcessed": false
    },
    {
        "ID": 67,
        "UserID": 7,
        "EmployeeID": "CLGAXO",
        "SiteName": "MULGRAVE",
        "BusinessUnitName": "Telstra Logistics - Melbourne",
        "AccountName": "IBM AUSTRALIA LTD",
        "GroupName": "CPE RASS",
        "CategoryName": "Activity - Productive",
        "TypeName": "CPE RASS",
```

```
        "Date": "2018-02-15",
        "Duration": "00:15",
        "IsProcessed": false
    },
    {
        "ID": 71,
        "UserID": 7,
        "EmployeeID": "CLGAXO",
        "SiteName": "MULGRAVE",
        "BusinessUnitName": "Telstra Logistics - Melbourne",
        "AccountName": "IBM AUSTRALIA LTD",
        "GroupName": "Inventory",
        "CategoryName": "Activity - Unproductive",
        "TypeName": "Inventory Relocation's",
        "Date": "2018-02-15",
        "Duration": "01:30",
        "IsProcessed": false
    },
    {
        "ID": 72,
        "UserID": 5,
        "EmployeeID": "HENDERSA",
        "SiteName": "MULGRAVE",
        "BusinessUnitName": "Telstra Logistics - Melbourne",
        "AccountName": "IBM AUSTRALIA LTD",
        "GroupName": "CPE",
        "CategoryName": "Activity - Productive",
        "TypeName": "CPE",
        "Date": "2018-02-15",
        "Duration": "00:30",
        "IsProcessed": false
    },
    {
        "ID": 90,
        "UserID": 5,
        "EmployeeID": "HENDERSA",
        "SiteName": "MULGRAVE",
        "BusinessUnitName": "Telstra Logistics - Melbourne",
        "AccountName": "IBM AUSTRALIA LTD",
        "GroupName": "CPE RASS",
        "CategoryName": "Activity - Productive",
        "TypeName": "CPE RASS",
        "Date": "2018-03-14",
        "Duration": null,
        "IsProcessed": false
    }
];
```

Coding Test

Test 3 SQL:

Provide the full set of SQLs and output for the following:

1) Create a new table called “model”, and insert the following records, with an auto-incrementing key called “ID”

Make	Model
Toyota	Corolla
Toyota	Camry
Nissan	Duke
Nissan	Duke
Mazda	Mazda 3
Mazda	CX5
Toyota	Camry
Ford	Raptor

2) write SQL statements to remove any records with duplicate make and model combinations from the table while keeping the record with the highest ID.