**Case Study**

**General Instructions:**

* Order data is provided a .js file. You have to import in your program.
* Node Library: request-promise, lodash, dotenv, express
* Exercise 1 should be completed, show the output and once approved you should move on to the next exercise

**About Order data**

* Order has header and Line Items {line details)
* Order Item has two important fields like MaterialGrp, TrackingNumber
* 2 Material groups in the order lines –
  + 78000090, - has 2 item - 2 tracking number - One has value , other has empty string,
  + 52131600 – has 4 items – 2 of them having same tracking number, other 2 have different tracking number

**Exercise 1:**

**Output is json: array of objects**

1. Use Get method to get

* order and all order details
  + Input : None
* Use URL parameter to get specific order lines (pass MaterialGrp / TrackingNumber / LineNumber)
  + Input : /getOrder/{MaterialGrp}
  + Input: /getOrder/{TrackingNumber}
* Use QueryString to get specific order lines (pass MaterialGrp / TrackingNumber / LineNumber)
  + Input : /getOrder?MaterailGrp=52131600&
  + Input : /getOrder?TrackingNumber=

1. Use POST method to get all the above details
   1. Input : /getOrder
      1. Body { "MaterailGrp": "52131600" }

**Exercise 2:**

Once Exercise 1 is over

* Create another with the copy and run is different port
* From app1 – You have to call App2 getOrder method to display data
* Do it for Get and POST routes/methods

**Exercise 3:**

Input : None

Data Processing rules: Get only Unique Order Items unique by MaterialGrp / TrackingNumber

Output json : Line Items as array of Objects

**Exercise 4:**

Method : Get/POST

Data Processing rules: Want to group by MaterialGrp and TrackingNumber

* 2 Material groups in the order lines –
* 78000090, - 2 tracking number - One has value , other has empty string
* 52131600 – has 4 items – 2 of them having same tracking number, other 2 have different tracking number

Input: none

Output json : Should be json object – Flatten the arrays and prepare the Object

{

"OrderNumber": "3179656",

"MaterialGrp\_1": "78000090",

"MaterialDesc\_1": "Oversize Freight",

"Grp1\_Noof\_Trackings": 2,

"Grp1\_LineNumber\_1": "000060",

" Grp1\_TrackingNumber\_1": "",

" Grp1\_LineNumber\_2": "000070",

" Grp1\_TrackingNumber\_2": "394099345960",

"MaterialGrp\_2": "52131600",

"MaterialDesc\_2": "MTL-Blinds / Shades",

"Grp2\_Noof\_Trackings": 3,

"Grp2\_LineNumber\_1": "000010",

" Grp2\_TrackingNumber\_1": "394099345961",

" Grp2\_LineNumber\_2": "000020",

" Grp2\_TrackingNumber\_2": "394099345962",

" Grp2\_LineNumber\_3": "000030",

" Grp2\_TrackingNumber\_3": "394099345963"

}

**Exercise 5:**

Once you have learnt MongoDb

Insert create 2 collections – order\_headers and order\_details

Use CURD operations – (Insert / update / delete / read) to get data and repeat the exercise 1 to 3