

MuleSoft Developer Code Challenge

Requirements

- Place your MuleSoft project that includes configuration XML and RAML files in a GitHub repo that we can clone and run
 - Use MuleSoft 4.x
 - Implement the following challenges
1. Design a flow, exposed as an API, that accepts the below Input JSON and provides a response (see Output below) with HTTP Response Code **201**.

Input JSON:

```
{ "employee": { "id": "A007", "first_name": "Jan", "last_name": "Doe" } }
```

Output:

Jan Doe employee profile is created in HRMS System.

2. Using a Dataweave function, create a separate flow that triggers the flow from #1 above, capture the response message, and publish a response message for this new flow.
3. Create a reusable Java function that converts a given string into camelCase and call that function inside Dataweave.
4. Based on the Input below -

Input #1

```
{ "employee": [ { "id": "A007", "first_name": "Jan", "last_name": "Doe" }, { "id": "A008", "first_name": "Ron", "last_name": "David" }, { "id": "A009", "first_name": "Jane", "last_name": "Kane" }, { "id": "A010", "first_name": "Marsh", "last_name": "Megan" }, { "id": "A011", "first_name": "Kelly", "last_name": "Mitch" }, { "id": "A012", "first_name": "Chris", "last_name": "Vaste" } ] }
```

Input # 2

```
{ "scoreBoard": [ { "id": "A007", "subjects": [ "Maths", "English", "Arts", "Drawings" ], "score": [ 80, 75, 89, 100 ] }, { "id": "A008", "subjects": [ "Maths", "English", "Arts", "Drawings" ], "score": [ 70, 85, 69, 90 ] }, { "id": "A021", "subjects": [ "Maths", "English", "Arts", "Drawings" ], "score": [ 90, 87, 77, 95 ] } ] }
```

- 4.1 Create a flow that joins Input #1 & Input #2 and produce a response in the below JSON format based on matching ID's in each input.

Response Output Format:

```
{"certificate":[{"id":"XXX","firstName":"XXX","LastName":"YYY","MarksBySubjectWise":{"Maths":80,"English":90,"Arts":89,"Drawings":100}},{"id":"XXX","firstName":"XXX","LastName":"YYY","MarksBySubjectWise":{"Maths":60,"English":80,"Arts":99,"Drawings":50}}]}
```

- 4.2 Create a flow that joins Inputs #1 & #2 and produce a response that includes all records from Input #1 that may not include corresponding records from Input #2.
5. Create a Mule flow that accepts a number (for ex: 10 as a URI Parameter) and produce the following JSON array as an output response.

(List of "N" natural numbers).

**** Do not use any custom Java or Dataweave functions.**

Output :: [1,2,3,4,5,6,7,8,9,10]

How long will this take - that totally depends on you. We think an estimate for something that matches the requirements above will be about 45 - 60 minutes.

If you have questions - please feel free to reach out to us through the recruiter and ask! We're a collaborative team and would be happy to discuss the challenge and clarify anything for you.