



MCD-LEVEL1-DELTA^{Q&As}

MuleSoft Certified Developer - Level 1 (Mule 4) DELTA

Pass Mulesoft MCD-LEVEL1-DELTA Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.passapply.com/mcd-level1-delta.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Mulesoft
Official Exam Center

- ⚙️ **Instant Download** After Purchase
- ⚙️ **100% Money Back** Guarantee
- ⚙️ **365 Days** Free Update
- ⚙️ **800,000+** Satisfied Customers



**QUESTION 1**

Refer to the exhibit.

```
#%RAML 1.0
title: ACME Telecom API
version: 1.0

/plans:
  get:
    responses:
      200:
        body:
          application/json:
            example: |
              [
                {
                  "plan_type": "Super Saver 500",
                  "plan_details": "all-inclusive",
                  "monthly_discount": 0.10
                },
                {
                  "plan_type": "Business Plus 1000",
                  "plan_details": "business package",
                  "monthly_discount": 0.20
                }
              ]
```

The API needs to be updated using the company-wide standard for the Plan data type. The Object data type has already been published in Anypoint Exchange with the global reference .

ACME/DataTypes/PlanDataType.raml.

What is a valid RAML specification that reuses the Plan data type?



A.

```
##RAML 1.0
title: ACME Telecom API
version: 1.0

dataTypes:
  Plan: !include ACME/DataTypes/PlanDataType.raml

/plans:
  get:
    responses:
      200:
        body:
          application/json:
            type: Plan[]
            example: !include ACME/Examples/PlanExamples.raml
```

B.

```
##RAML 1.0
title: ACME Telecom API
version: 1.0

types:
  Plan: !reference ACME/DataTypes/PlanDataType.raml

/plans:
  get:
    responses:
      200:
        body:
          application/json:
            type: Plan[]
            example: !reference ACME/Examples/PlanExamples.raml
```

C.

```
##RAML 1.0
title: ACME Telecom API
version: 1.0

dataTypes:
  Plan: !reference ACME/DataTypes/PlanDataType.raml

/plans:
  get:
    responses:
      200:
        body:
          application/json:
            type: Plan[]
            example: !reference ACME/Examples/PlanExamples.raml
```

D.

```
##RAML 1.0
title: ACME Telecom API
version: 1.0

types:
  Plan: !include ACME/DataTypes/PlanDataType.raml

/plans:
  get:
    responses:
      200:
        body:
          application/json:
            type: Plan[]
            example: !include ACME/Examples/PlanExamples.raml
```



A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: D

QUESTION 2

Refer to the exhibit.

The screenshot displays a web application interface with two main panels. The left panel, titled 'order.xml', shows an XML document with two items. The first item has an order ID of 592, international shipping, is a T-shirt in Navy color, size L, with a quantity of 1 and a price of 20. The second item has an order ID of 972, domestic shipping, is Cargo Shorts, size XL, with a quantity of 2 and a price of 30. The right panel, titled 'Output Payload', shows the resulting JSON output. It is a JSON array with two objects. The first object corresponds to the first item in the XML, and the second object corresponds to the second item. The JSON output is as follows:

```
[
  {
    "index": 0,
    "orderId": "592",
    "itemName": "T-shirt Navy",
    "lineItemPrice": 20
  },
  {
    "index": 1,
    "orderId": "972",
    "itemName": "Cargo Shorts",
    "lineItemPrice": 60
  }
]
```

What Database expression transforms the input to the output?



- A.

```
payload.order.*item map ( (value,index) -> {  
    index: index,  
    orderId: value.orderId,  
    itemName: value.item,  
    lineItemPrice: (value.price as :number) * (value.quantity as :number)  
})
```
- B.

```
payload.order.*item map ( (value,index) -> {  
    index: index,  
    orderId: value.@orderId,  
    itemName: value.item,  
    lineItemPrice: (value.price as Number) * (value.quantity as Number)  
})
```
- C.

```
payload.order.*item map ( (value,index) -> {  
    index: index,  
    orderId: value.@orderId,  
    itemName: value.item,  
    lineItemPrice: (value.price as :number) * (value.quantity as :number)  
})
```
- D.

```
payload.order.*item map( (value,index) -> {  
    index: index,  
    orderId: value.orderId,  
    itemName: value.item,  
    lineItemPrice: (value.price as Number) * (value.quantity as Number)  
})
```

A. Option A

B. Option B

C. Option C

D. Option D

Correct Answer: B

QUESTION 3

A web client sends a request to `http://localhost:8081?dept=sales`. What is the correct DataWeave expression to access the value of `dept`?

A. `attributes.queryParams.dept`

B. `attributes.dept`

C. `message.queryParams.dept`



D. vars.dept

Correct Answer: A

QUESTION 4

A Set Variable component saves the current payload to a variable.

What is the DataWeave parent expression to access the variable?

A. #[value]

B. #[vars]

C. #[var]

D. #[values]

Correct Answer: B

QUESTION 5

To avoid hard-coding values, a flow uses some property placeholders and the corresponding values are stored in a configuration file.

Where does the configuration file's location need to be specified in the Mule application?

A. The pom.xml file

B. A global element

C. The mule-art if act .json file

D. a flow attribute

Correct Answer: B

[MCD-LEVEL1-DELTA PDF Dumps](#)

[MCD-LEVEL1-DELTA Exam Questions](#)

[MCD-LEVEL1-DELTA Braindumps](#)