



MCD-LEVEL1^{Q&As}

MuleSoft Certified Developer - Level 1 (Mule 4)

Pass Mulesoft MCD-LEVEL1 Exam with 100% Guarantee

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

<https://www.passapply.com/mcd-level1.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

What DataWeave expression transforms the example XML input to the CSV output?



```
<?xml version="1.0" encoding="UTF-8"?>
<sale>
  <item itemId="592" saleId="1000">
    <shipping>international</shipping>
    <desc>T-shirt Navy</desc>
    <size>L</size>
    <quantity>1</quantity>
    <price>20</price>
  </item>
  <item itemId="972" saleId="1000">
    <shipping>domestic</shipping>
    <desc>Cargo Shorts</desc>
    <size>XL</size>
    <quantity>2</quantity>
    <price>30</price>
  </item>
</sale>
```

```
Output Payload • Preview
1 %dw 2.0
2 output application/csv
3 ---
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
index,sale,itemName,itemPrice,item
0,1000,T-shirt Navy,20,592
1,1000,Cargo Shorts,60,972
```

- A. `payload.sale.*item map ((value,index) -> {
 index: index,
 sale: value.@saleId,
 itemName: value.desc,
 itemPrice: (value.price) * (value.quantity),
 item: value.@itemId
})`
- B. `payload.sale.*item map ((value,index) -> {
 index: index,
 sale: value.saleId,
 itemName: value.desc,
 itemPrice: (value.price) * (value.quantity),
 item: value.itemId
})`
- C. `payload.sale.item map ((value,index) -> {
 index: index,
 sale: value.@saleId,
 itemName: value.desc,
 itemPrice: (value.price) * (value.quantity),
 item: value.@itemId
})`
- D. `payload.sale.item map ((value,index) -> {
 index: index,
 sale: value.saleId,
 itemName: value.desc,
 itemPrice: (value.price) * (value.quantity),
 item: value.itemId
})`



- A. Option A
- B. Option B
- C. Option C
- D. Option D

Correct Answer: A

Correct answer is as below. Attributes in the incoming xml payload are always accessed using @. Similarly *item is required as we have multiple items in the request %dw 2.0 output application/csv

```
payload.sale.*item map ((value, index) -> {  
  index: index,  
  sale: value.@saleId,  
  itemName: value.desc,  
  itemPrice: (value.quantity) * (value.price),  
  item: value.@itemId  
})
```

QUESTION 2

Refer to the exhibits.



```
<flow name="logPayload" >
  <scheduler doc:name="Scheduler" >
    <scheduling-strategy >
      <fixed-frequency />
    </scheduling-strategy>
  </scheduler>
  <set-payload value="#[{'year': '2020'}]" doc:name="{ 'year': '2020'}" />
  <logger level="INFO" doc:name="Logger" message="????" />
</flow>
```

The Set Payload transformer's value is set to `{'year': '2020'}`.

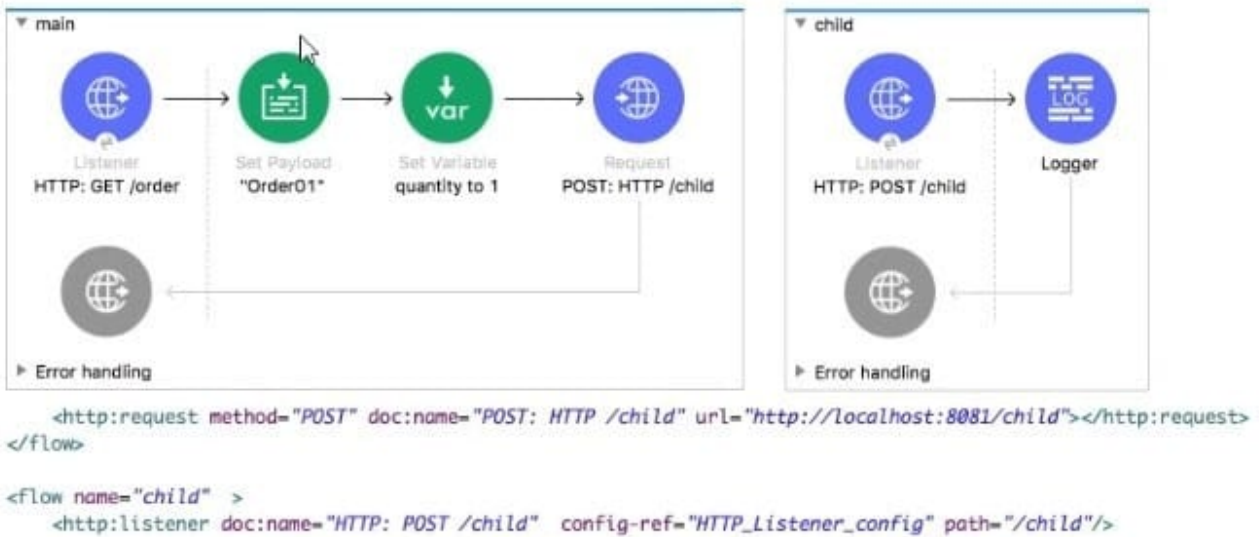
What message value should be added to the Logger component to output the message `The year is 2020`, without hardcoding 2020?

- A. `#[The year is $(payload.year)]`
- B. `The year is #[payload.year]`
- C. `#[The year is " + payload.year]`
- D. `#[The year is ++ payload.year]`

Correct Answer: D

QUESTION 3

Refer to the exhibits.



The main flow contains an HTTP Request. The HTTP Listeners and HTTP Request use default configurations.

What values are accessible in the child flow after a web client submits a request to <http://localhost:8081/order>? col or = red?

- A. payload
- B. payload quantity var
- C. payload color query param
- D. payload quantity var color query param

Correct Answer: A

QUESTION 4

A company has an API to manage purchase orders, with each record identified by a unique purchase order ID. The API was built with RAML according to MuleSoft best practices. What URI should a web client use to request order P05555?

- A. /orders/{P05555}
- B. /orders/order=P05555
- C. /orders?order=P05555
- D. /orders/P05555

Correct Answer: D

QUESTION 5

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

As can be seen in RAML, POST expects input in application/json format which eliminates two of the options as two options are in xml format. Now out of the two remaining options, one has id field in request which is only mentioned for get

response and not for POST request. Hence id field is not expected in POST request.

Hence correct answer is

```
{  
  "name": "GoerdiLa Forge",  
  "address": "1 Westland CA",  
  "customer_since": "2014-01-04",  
  "balance": "4829.29",  
  "bank_agend_id": "12556"  
}
```

[Latest MCD-LEVEL1 Dumps](#)

[MCD-LEVEL1 Practice Test](#)

[MCD-LEVEL1 Study Guide](#)