Top of Form

Question 1: **Correct**

Refer to the exhibits. How many private flows does APIKIt generate from RAML specification?

[Larger image](javascript:void(0))

* ​

2

* ​

3

* ​

1

* ​

4

**(Correct)**

**Explanation**

APIKIt Creates a separate flow for each HTTP method. Hence 4 private flows would be generated.

MuleSoft Documentation Referrence : <https://docs.mulesoft.com/mule-runtime/4.3/build-application-from-api>

Bottom of Form

Top of Form

Question 2: **Correct**

'Refer to exhibits. What message should be added to Logger component so that logger prints "The city is Pune" (Double quote should not be part of logged message)

[Larger image](javascript:void(0))

* ​

The city is #[payload.City]

**(Correct)**

* ​

#["The city is" ++ payload.City]

* ​

#[The city is ${payload.City}

* ​

The city is + #[payload.City]

**Explanation**

Correct answer is The city is #[payload.City]

Answer can get confused with the option #["The city is" ++ payload.City] But note that this option will not print the space between is and city name. This will print The city isPune

Bottom of Form

Top of Form

Question 3: **Correct**

Refer to the exhibit. What is the output payload in the On Complete phase?

[Larger image](javascript:void(0))

* ​

The records are processed by all batch steps

* ​

Summary statistics with No record data

**(Correct)**

* ​

The original payload : [1,2,3]

* ​

The records are processed by last batch step

**Explanation**

This is a tricky question. On complete phase payload consists of summary of records processed which gives insight on which records failed or passed.

Hence correct answer is **Summary statistics with No record data**

MuleSoft Documentation Reference : <https://docs.mulesoft.com/mule-runtime/4.3/batch-processing-concept#on-complete>

Bottom of Form

Top of Form

Question 4: **Correct**

**A RAML example fragment named StudentExample.raml is placed in the examples folder in an API specification project. What is the correct syntax to reference the fragment?**

* ​

examples: !include StudentExample.raml

* ​

examples: #import examples/StudentExample.raml

* ​

examples: #import StudentExample.raml

* ​

examples: !include examples/StudentExample.raml

**(Correct)**

**Explanation**

Correct answer is examples: !include examples/StudentExample.raml

correct syntax is  examples: !include <full/relative file path>

Bottom of Form

Top of Form

Question 5: **Correct**

**What valid RAML retrieves details on a specific order by its orderId as a URI parameter?**

* ​
  1. /orders:
  2. get:
  3. /{orderId}:
* ​
  1. /orders:
  2. /{orderId}:
  3. get:

**(Correct)**

* ​
  1. /orders:
  2. /get:
  3. /orderId
* ​
  1. /orders:
  2. /orderId:
  3. get:

**Explanation**

Correct answer is below as it follows the correct syntax

/orders:

/{orderId}:

get:

Bottom of Form

Top of Form

Question 6: **Correct**

Refer to the exhibit. A web client submits a request to the HTTP Listener and the HTTP Request throws an error. What payload and status code are returned to the web client?

[Larger image](javascript:void(0))

* ​

Response body: "Error" Default response status code: 200

**(Correct)**

* ​

Response body: "Success-End" Default response status code: 200

* ​

Response body: error.description Default response status code: 200

* ​

Response body: "Success-Begin" Default response status code: 200

**Explanation**

When HTTP Request throws an error, it goes in error handling section. Note here that On Error Continue scope is being used here is error flow. When On Error Continue scope is invoked, all the processors in error block are executed and success response is sent back to the client with payload which is set in error flow. In this case payload is set to "Error" value in error block. Hence it will be sent back to client with http code as 200 as On error continue always sends success error code. Hence option 4 is correct answer.

Bottom of Form

Top of Form

Question 7: **Correct**

Refer to the exhibits. The main flow is configured with three error handlers. A web client submits the request to the HTTP Listener and the HTTP Request throws an HTTP:NOT\_FOUND error. What response message is returned?

[Larger image](javascript:void(0))

* ​

Other error

* ​

HTTP:NOT\_FOUND

* ​

Success - main flow

* ​

APP: API RESOURCE NOT FOUND

**(Correct)**

**Explanation**

In this case ,HTTP:NOT\_FOUND error originating from HTTP:GET acme.com/virgin is mapped to custom error APP:API\_RESOURCE\_NOT\_FOUND. Hence when HTTP:NOT\_FOUND error is encountered, it would be caught by error handler with type as  APP:API\_RESOURCE\_NOT\_FOUND. This error processor sets payload to APP:API\_RESOURCE\_NOT\_FOUND.

Hence correct answer is **APP: API RESOURCE NOT FOUND**

Bottom of Form

Top of Form

Question 8: **Correct**

Refer to the exhibits. What can be added to the flow to persist data across different flow executions?

[Larger image](javascript:void(0))

* ​

Properties of Mule runtime flow Object

* ​

Properties of mule runtime app Object

* ​

Session variables

* ​

Key/value pair in Object store

**(Correct)**

**Explanation**

An object store is a facility for storing objects in or across Mule applications. Mule runtime engine (Mule) uses object stores to persist data for eventual retrieval. Internally, Mule uses object stores in various filters, routers, and other message processors that need to store states between messages.

Object stores are available in all deployment targets. If you deploy your application to CloudHub, you can also use Object Store V2.

Correct answer is **Key/value pair in Object store**

MuleSoft Documentation reference :  <https://docs.mulesoft.com/mule-runtime/4.3/mule-object-stores#use-cases>

Bottom of Form

Top of Form

Question 9: **Correct**

A flow contains an HTTP Listener as the event source. What is the DataWeave expression to log the Content-Type header using a Logger component?

* ​

#["Content-Type: " ++ attributes.headers.'content-type']

**(Correct)**

* ​

#["Content-Type: " + headers.'content-type']

* ​

#["Content-Type: " + attributes.headers.'content-type']

* ​

#["Content-Type: " ++ headers.'content-type']

**Explanation**

Correct answer is **#["Content-Type: " ++ attributes.headers.'content-type']**

Reason in as below

1) Concatenation is always with ++ sign and not with  + sign which makes two of the options incorrect

2) Headers can be accessed with attributes.headers and not with only headers which makes below option incorrect

#["Content-Type: " ++ headers.'content-type']

Bottom of Form

Top of Form

Question 10: **Correct**

Refer to the exhibits. What expression correctly specifies input parameters to pass the city and state values to SQL query?

[Larger image](javascript:void(0))

* ​
  1. #[[
  2. "Pune",
  3. "Maharashtra"
  4. ]]
* ​
  1. #[inputParams: [
  2. "Pune",
  3. "Maharashtra"
  4. ]]
* ​
  1. #[inputParams: {
  2. city: "Pune",
  3. state: "Maharashtra"
  4. }]
* ​
  1. #[{
  2. city: "Pune",
  3. state: "Maharashtra"
  4. }]

**(Correct)**

**Explanation**

Correct answer is

#[{

city: "Pune",

state: "Maharashtra"

}]

MuleSoft Documentation Reference : <https://docs.mulesoft.com/db-connector/1.9/database-connector-select>

Bottom of Form

Top of Form

Question 11: **Correct**

Refer to the exhibits. What is written to the records.csv when the flow executes?

[Larger image](javascript:void(0))

* ​

An error message

* ​

Nothing

* ​

JSON payload

**(Correct)**

* ​

payload converted to csv

**Explanation**

Transform Message Add write\_date is coverting payload in JSON format and same JSON payload is avaialble to file write processor. However, if the payload is a different format (for example, not CSV) , you can place the transformation inside the Write operation to generate content that will be written without producing a side effect on the message in transit. This is not done in this case. By default, the connector writes whatever is in the message payload. Hence JSON payload will be written to file.

Correct answer is **JSON payload**

Bottom of Form

Top of Form

Question 12: **Correct**

Refer to the exhibit. What data is expected by the POST /accounts endpoint?

[Larger image](javascript:void(0))

* ​
  1. <item>
  2. <address>1 Westland CA</address>
  3. <balance>4829.29</balance>
  4. <bank\_agend\_id>12556</bank\_agend\_id>
  5. <customer\_since>2014-01-04</customer\_since>
  6. <name>GoerdiLa Forge</name>
  7. </item>
* ​
  1. {
  2. "id": "777744"
  3. "name": "GoerdiLa Forge",
  4. "address": "1 Westland CA",
  5. "customer\_since": "2014-01-04",
  6. "balance": "1229.29",
  7. }
* ​
  1. <item>
  2. <id>777744</id>
  3. <address>1 Westland CA</address>
  4. <balance>1229.29</balance>
  5. <bank\_agend\_id>12556</bank\_agend\_id>
  6. <customer\_since>2014-01-04</customer\_since>
  7. </item>
* ​
  1. {
  2. "name": "GoerdiLa Forge",
  3. "address": "1 Westland CA",
  4. "customer\_since": "2014-01-04",
  5. "balance": "4829.29",
  6. "bank\_agend\_id": "12556"
  7. }

**(Correct)**

**Explanation**

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"

}

Bottom of Form

Top of Form

Question 13: **Correct**

Refer to the exhibit. The API needs to be updated using the company wide standard for the Plan data type. The Object data type has already been published to Anypoint Exchange with the global reference.  ACME/DataTypes/PlanData.raml  What is valid RAML specification that reuses the Plan Data type?

* ​
  1. #%RAML 1.0
  2. title: ACME Telecom API
  3. dataTypes:
  4. Plan: !import ACME/DataTypes/PlanDataType.raml
  6. /plans:
  7. get:
  8. responses:
  9. 200:
  10. body:
  11. application/json:
  12. type: Plan[]
  13. example: !import ACME/Examples/PlanExamples.raml
* ​
  1. #%RAML 1.0
  2. title: ACME Telecom API
  3. types:
  4. Plan: !referrence ACME/DataTypes/PlanDataType.raml
  6. /plans:
  7. get:
  8. responses:
  9. 200:
  10. body:
  11. application/json:
  12. type: Plan[]
  13. example: !referrence ACME/Examples/PlanExamples.raml
* ​
  1. #%RAML 1.0
  2. title: ACME Telecom API
  3. types:
  4. Plan: !include ACME/DataTypes/PlanDataType.raml
  6. /plans:
  7. get:
  8. responses:
  9. 200:
  10. body:
  11. application/json:
  12. type: Plan[]
  13. example: !include ACME/Examples/PlanExamples.raml

**(Correct)**

* ​
  1. #%RAML 1.0
  2. title: ACME Telecom API
  3. dataTypes:
  4. Plan: !referrence ACME/DataTypes/PlanDataType.raml
  6. /plans:
  7. get:
  8. responses:
  9. 200:
  10. body:
  11. application/json:
  12. type: Plan[]
  13. example: !referrence ACME/Examples/PlanExamples.raml

**Explanation**

RAML keyword to use a reference is !include.

Hence correct answer is

#%RAML 1.0

title: ACME Telecom API

types:

Plan: !include ACME/DataTypes/PlanDataType.raml

/plans:

get:

responses:

200:

body:

application/json:

type: Plan[]

example: !include ACME/Examples/PlanExamples.raml

Bottom of Form

Top of Form

Question 14: **Correct**

Refer to the exhibits. A webclient  submits the request to http://localhost:8081/flights?destination=SFO and the Web Service Consumer throws a WSC:BAD\_REQUEST error. What is the next step to fix this error?

[Larger image](javascript:void(0))

* ​

Set a SOAP payload before the Consume operation that contains the destination query parameter

**(Correct)**

* ​

Set a JSON payload before the Consume operation that contains the destination query parameter

* ​

Set a property in Consume operation equal to destination query parameter

* ​

Set a header in Consume operation equal to destination query parameter

**Explanation**

As can be seen in error message , SOAP service findFlights expects the SOAP paylaod. This can be set using transform message processor which forms SOAP payload before the Consume operation that contains the destination query parameter.

Correct answer is ***Set a SOAP payload before the Consume operation that contains the destination query parameter***

Bottom of Form

Top of Form

Question 15: **Correct**

Refer to the exhibit. What payload is returned from a request to http://localhost:8081/?

[Larger image](javascript:void(0))

* ​

1

* ​

2

* ​

3

**(Correct)**

* ​

4

**Explanation**

The flow can be described as below.  1) First HTTP POST request is made in which payload is set to 1 and it gets returned to our mail flow.   2) Second call is initiated for JMS Publish Consume JMS: num1 which add 1 to the payload which makes it as 2. Note that publish consume is a synchronous operation. Hence payload is returned to main flow.  3) Third  call is initiated for JMS Publish JMS: num2 which add 1 to the payload . Note that publish  is asynchronous operation. Hence payload is never returned to main flow. So payload in main flow is still 2.  4) Finally Set Payload increments payload by 1 making payload as 3 which is returned by the flow.  Hence correct answer is 3

Bottom of Form

Top of Form

Question 16: **Incorrect**

Refer to the exhibits. What Dataweave expression transforms the input to the output?

[Larger image](javascript:void(0))

* ​
  1. payload.order.\*item map {(value, index) -> {
  2. index: index,
  3. orderId: value.@orderId,
  4. itemName: value.item,
  5. lineItemPrice: (value.price as Number) \* (value.quantity as Number)
  6. }
  7. }

**(Correct)**

* ​
  1. payload.order.\*item map {(value, index) -> {
  2. index: index,
  3. orderId: value.orderId,
  4. itemName: value.item,
  5. lineItemPrice: (value.price as Number) \* (value.quantity as Number)
  6. }
  7. }
* ​
  1. payload.order.\*item map {(value, index) -> {
  2. index: index,
  3. orderId: value.@orderId,
  4. itemName: value.item,
  5. lineItemPrice: (value.price as :number) \* (value.quantity as :number)
  6. }
  7. }

**(Incorrect)**

* ​
  1. payload.order.\*item map {(value, index) -> {
  2. index: index,
  3. orderId: value.orderId,
  4. itemName: value.item,
  5. lineItemPrice: (value.price as :number) \* (value.quantity as :number)
  6. }
  7. }

**Explanation**

This kind of questions are best solved with filtering technique. Key thing to note here is XML attributes are always access by @ notation. So in this example orderId can be accessed as @OrderId which makes two of the options incorrect.  Correct syntax to format a String in Number is value.price as Number.

Hence correct answer is

payload.order.\*item map {(value, index) -> {

index: index,

orderId: value.@orderId,

itemName: value.item,

lineItemPrice: (value.price as Number) \* (value.quantity as Number)

}

}

Bottom of Form

Top of Form

Question 17: **Correct**

Mule application contains two HTTP Listeners, each configured for different API endpoints: http://trainingdemo.com/apis/orders and

http: //trainingdemo.com/apis/customers

What base path value should be set in an HTTP Listener config element so that it can be used to configure both HTTP Listeners?

* ​

/apis/\*

**(Correct)**

* ​

/apis/?

* ​

/apis/orders|customers

* ​

/apis/

**Explanation**

Correct answer as  /api/\*   It accepts everything starting with /api/

While configuring HTTP Listener , you can use wildcards in the value for **Path**, to listen for all incoming requests to any path within the specified base path. You can also specify a partial path that ends in /\*, pointing to any path that begins as defined but that could also be extended with anything else.

MuleSoft Doc Reference : <https://docs.mulesoft.com/http-connector/0.3.8/http-listener-connector#using-wildcards-in-the-path>

Bottom of Form

Top of Form

Question 18: **Correct**

Refer to the exhibit. What is the correct DataWeave expression for the Set Payload transformer to call the createCustomerObject flow with values for the first and last names of a new customer?

[Larger image](javascript:void(0))

* ​

lookup( "createCustomerObject", {first: "Aice, last: "Green"})

**(Correct)**

* ​

lookup( "createCustomerObject", ("Alice", "Green"))

* ​

createCustomerObject("Alice", "Green")

* ​

createCustomerObject({first: "Alice", last' "Green"})

**Explanation**

**lookup(String, Any, Number)**

This function enables you to execute a flow within a Mule app and retrieve the resulting payload.

It works in Mule apps that are running on Mule Runtime version 4.1.4 and later.

Similar to the Flow Reference component (recommended), the lookup function enables you to execute another flow within your app and to retrieve the resulting payload. It takes the flow’s name and an input payload as parameters. For example, lookup("anotherFlow", payload) executes a flow named anotherFlow.

Correct answer is

lookup( "createCustomerObject", {first: "Aice, last: "Green"})

MuleSoft Documentation Reference : <https://docs.mulesoft.com/mule-runtime/4.3/dw-mule-functions-lookup>

Bottom of Form

Top of Form

Question 19: **Correct**

A web client submits a request to http://localhost:8081?accountType=personal.The query parameter is captured using a Set Variable transformer to a variable named accountType. What is the correct Dataweave expression to log accountType?

* ​

Account Type : #[flowvars.accountType]

* ​

Account Type : #[vars.accountType]

**(Correct)**

* ​

Account Type : #[message.inboundProperties.accountType]

* ​

Account Type : #[attributes.accountType]

**Explanation**

Correct answer is Account Type : #[vars.accountType]

**vars**: Keyword for accessing a variable, for example, through a DataWeave expression in a Mule component, such as the Logger, or from an Input or Output parameter of an operation. If the name of your variable is myVar, you can access it like this: vars.myVar

MuleSoft Documentation Reference : <https://docs.mulesoft.com/mule-runtime/4.3/about-mule-variables>

Bottom of Form

Top of Form

Question 20: **Correct**

Refer to the exhibits. What is valid text to set the user field in the Database connector configuration to the username value specified in the config.yaml file?

[Larger image](javascript:void(0))

* ​

${db.username}

**(Correct)**

* ​

${db:username}

* ​

#[db.username]

* ​

#[db:username>

**Explanation**

Correct way to access configuration properties in DB connector is

${db.username}

Bottom of Form

Top of Form

Question 21: **Correct**

Refer to the exhibits. The validation component in the private flow throws an error. What response message is returned to a client request to the main flow's HTTP Listener?

[Larger image](javascript:void(0))

[Larger image](javascript:void(0))

* ​

Error - main flow

* ​

Success - main flow

**(Correct)**

* ​

Error - private flow

* ​

Validation Error

**Explanation**

Note that private flow has error scope defined as On Error Continue . So when error occurs in private flow , it is handled by this  On Error Continue scope which sends success response back to main flow and does not throw back an error. So main continues normally and payload is set to Success - main flow.

Hence correct answer is **Success - main flow**

Bottom of Form

Top of Form

Question 22: **Incorrect**

Refer to the exhibits. What is the response when a client submits a request to http://localhost:8081?

[Larger image](javascript:void(0))

* ​

Before

* ​

After

**(Incorrect)**

* ​

Validation error

**(Correct)**

* ​

null

**Explanation**

Correct answer is **Validation error**

First payload is being set to value "Before" and then Is null validation is used which will pass the message only if payload is null. In this case as payload is not null , it will throw validation error as configured. Hence validation error would be returned to client.

Bottom of Form

Top of Form

Question 23: **Correct**

**What is correct syntax for a Logger component to output a message with the contents of a JSON Object payload?**

* ​

The  payload is: #[payload]

**(Correct)**

* ​

#["The  payload is " + payload]

* ​

The payload is: $(payload)

* ​

#["The  payload is " ++ payload]

**Explanation**

The correct answer is  The  payload is: #[payload]

You may get confused with one more solution which is #["The  payload is " ++ payload] .

But this solution will throw an error as concatenation operator expects string as the arguments. In this case as payload is JSON

Bottom of Form

Top of Form

Question 24: **Incorrect**

Refer to the exhibits. A web client submits the request to the HTTP Listener. What response message would be returned to web client?

[Larger image](javascript:void(0))

[Larger image](javascript:void(0))

* ​

No response would be sent back to client and request will get errored out in Mule

* ​

Start

* ​

End

**(Incorrect)**

* ​

String is not blank

**(Correct)**

**Explanation**

Correct answer is  **String is not blank.**

As no error handling is provided. Default error is sent back which is String is not blank

Bottom of Form

Top of Form

Question 25: **Incorrect**

What module and operation will throw an error if a Mule events payload is not number?

* ​

None of these

* ​

Filter modules Is Number operation

**(Incorrect)**

* ​

Validation modules Is Number operation

**(Correct)**

* ​

Validation modules Is not Number operation

**Explanation**

Correct answer is **Validation modules Is Number operation.**

Mule 4 does not use filters anymore. The functionality provided by filters in Mule 3 can be achieved by using the [Validation Module](https://docs.mulesoft.com/validation-connector/1.4/).

Bottom of Form

Top of Form

Question 26: **Correct**

Mule application contains ActiveMQ JMS dependency. Mule application was  compiled and run successfully in Anypoint Studio. The mule application must now be exported from Anypoint Studio and shared with other developer. What export options should be selected to create the smallest JAR file that can be imported into other developer's Anypoint Studio and run successfully?

* ​

De-select both **Attach Project Sources**and  **Include project modules and dependencies** option

* ​

Select only **Attach Project Sources only**

**(Correct)**

* ​

Select the **Include project modules and dependencies** option only

* ​

Select both **Attach Project Sources**and  **Include project modules and dependencies** option

**Explanation**

Correct answer is Select only **Attach Project Sources only**

You must keep the **Attach Project Sources** option selected to be able to import the packaged JAR file back into a Studio workspace.

Deselect the **Include project modules and dependencies** option.

This option skips bundling the actual modules and external dependencies required to run the Mule application in a Mule runtime engine, creating a lightweight JAR file package that does not include any dependencies specified in the Mule application’s pom.xml file.

The generated JAR file is not a functional deployable archive and cannot be deployed to a Mule runtime engine, but instead offers a way to archive only the source files that make up the Mule application. This is the same as using the -lightWeightPackage flag when packaging using the Mule Maven Plugin and is useful if you want to distribute your project to other Studio installations because it only keeps a reference to all its dependencies. When you import a lightweight package into Studio, all your dependencies are automatically downloaded

MuleSoft Doc Ref : <https://docs.mulesoft.com/studio/7.8/import-export-packages>

Bottom of Form

Top of Form

Question 27: **Incorrect**

As a part of requirement , application property defined below needs to be accessed as Dataweave expression. What is the correct expression to map it to port value?

* ​

{ port : {db:port}}

**(Incorrect)**

* ​

Application property cannot be accessed in Dataweave

* ​

{ port : p('db.port')}

**(Correct)**

* ​

{ port : p['db.port']}

**Explanation**

Correct answer is **{ port : p('db.port')}**

This function returns a string that identifies the value of one of these input properties: Mule property placeholders, System properties, or Environment variables.

The p function returns a null value if the property is not set or if the function does not find the property.

MuleSoft Documentation Reference : <https://docs.mulesoft.com/mule-runtime/4.3/dw-mule-functions-p>

Bottom of Form

Top of Form

Question 28: **Correct**

Refer to the exhibits. Mule application has an HTTP request configuration where host name is hardcoded. Organization is looking to move host and port values to configuration file. What valid expression can be used to so that HTTP configuration can pick the value from configuration file?

[Larger image](javascript:void(0))

[Larger image](javascript:void(0))

* ​

${training.host}

**(Correct)**

* ​

#[training.host]

* ​

${http.host}

* ​

#{training.host}

**Explanation**

Correct answer is **${training.host}**

Bottom of Form

Top of Form

Question 29: **Correct**

**What is the correct syntax to define and and call a function in Dataweave?**

* ​
  1. %function addKV( object: Object, key: String, value: Any) =
  2. object ++ {(key):(value)}
  3. ---
  4. { hello: "world"} addKV ( "hola","mundo" )
* ​
  1. fun addKV( object: Object, key: String, value: Any) =
  2. object ++ {(key):(value)}
  3. ---
  4. addKV ( {"hello': "world"}, "hola", "mundo" )

**(Correct)**

* ​
  1. fun addKV( object: Object, key: String, value: Any) =
  2. object ++ {(key):(value)}
  3. ---
  4. { hello: "world"} addKV ( "hola","mundo" )
* ​
  1. %function addKV( object: Object, key: String, value: Any) =
  2. object ++ {(key):(value)}
  3. ---
  4. addKV ( {"hello': "world"}, "hola", "mundo" )

**Explanation**

Keyword to ad function in Dataweave transformation is fun.  Also parameters needs to be passed exactly in same order as defined in function definition.

Hence correct answer is

fun addKV( object: Object, key: String, value: Any) =

object ++ {(key):(value)}

---

addKV ( {"hello': "world"}, "hola", "mundo" )

MuleSoft Documentation Reference : <https://docs.mulesoft.com/mule-runtime/4.3/dataweave-functions>

Bottom of Form