**1**. One route in a Scatter-Gather router throws an exception and the exception is not handled by the flow or resource being called. Without additional configuration, how will scatter-Gather handle this situation?

A. The responses from all successful routes will be aggregated and the failed routes response throws an exception on a separate thread; the failed response can be caught and handled.

B. Scatter-Gather retries the failed route once; if the same exception is thrown. Scatter-Gather throws an exception.

C. Scatter-Gather throws an exception; by default Scatter-Gather allows for no failed routes.

D. The responses from all successful routes will be aggregated and the failed routes response is dropped.

**2**. What is the expected behavior of an Inbound FTP endpoint placed within the source of a flow?

A. To connect to an FTP server and check for new files. Every new file creates a new message. By default, Mule deletes the file once it is downloaded.

B. FTP connectors only support outbound endpoints.

C. To connect to an FTP server and check for new files. Every new file creates a new message. Mule doesn't change the file on the FTP server.

D. To create an FTP server and configure it to initiate a message every time a file is uploaded.

**3**. What does the APIKit router not do?

A. Set the baseUri of the application based on the RAML definition.

B. Validate incoming request schemaS based on the RAML definition.

C. Route requests to specific flows where the flow mapping is defined, regardless of naming syntax.

D. Implicitly route requests to flows with the naming syntax (method}:{resource}:{routerConfigName).

**4**. Which of the following configurations is valid?

A. A private flow (a flow without a message source) with an exception strategy

B. A flow with no message processors

C. A subflow with a VM inbound endpoint

D. A subflow with an exception strategy

**5**. Which 'three' statements about DataWeave transformations are true? Choose 3 answers

A. They can be reused for different inputs.

B. They can be reused for different types of outputs.

C. They can be written inline in a Transform Message component.

D. They can be stored in external OWL files.

**6**. A flow has the following configuration:

<flow>

…

<set-propecty doc:name=”Property” propertyName=”size” value=”small”/>

</flow>

What Is the correct MEL expression for retrieving the value of the size property?

A. #[message.outboundProperties(‘size’)]

B. #[message.inboundProperties(‘size’)]

C. #[message.properties(‘size’)]

D. #[message.getProperty(‘size’)]

**7**. Which part of a Mule message is considered immutable?

A. Session variables

B. Outbound Properties

C. Payload

D. Inbound Properties

E. Flow variables

F. Attachments

**8**. A flow needs to be exposed at http://{host}/myPath for users to retrieve data. Assuming the flow logic already exists, what are the minimum elements needed to achieve this?

A. An http:listener-config

B. An http:listener-config and a RAML file

C. An http:listener endpoint and an http:listener-config

D. An http:listener endpoint

**9.** What is the default behavior of a filter?

A. To let messages through or route them to a second flow based on if a condition is not met.

B. To let messages through or route them to a DLQ if a condition is not met.

C. To let parts of a message through based on the filter expression.

D. To let messages through or drop them if a condition is not met.

**10.** What does the payload of a Mule message always contain?

A. A Java object implementing the Collection interface (such as an ArrayList)

B. A JSON representation the data

C. A Java object determined by the messages source and processors

D. A XML representation of the data

**11**. Which case best warrants the use of a subflow?

A. A need to define specific error handling for a group of logic

B. A need to send a Mule message to a different flow via a transport-based endpoint

C. A need to reactor duplicate logic into One reusable call

D. A need to tune the thread pool on a group of message processors

**12**. When separating a mule application into multiple Mule configuration (XML) files, which of the following Is true?

A. All flows must have unique names, regardless of whether the flows exist in different Mule configuration (XML) files.

B. Flows in one Mule configuration (XML) file don't have visibility into flows of another Mule configuration (XML) file.

C. Subflows, because they do not have a message source, cannot be referenced from another Mule configuration (XML) file.

D. Global elements defined in one Mule configuration (XML) file cannot be referenced from another Mule configuration (XML) file.

**13**. Which of the following transports only supports the one-way exchange-pattern?

A. HTTP

B. VM

C. File

D. JMS

**14**. Payload metadata is automatically created for which “two” types of connector endpoints? Choose 2 answers

A. File

B. Database

C. HTTP Request

D. Web Service Consumer

**15.** Mule applications deployed to various environments (QA, Staging, Production) do not contain the same connection attribute values. How can you best ensure the connection attribute values change based on the environment context?

A. Create multiple property placeholders. For example, context:property-placeholder location=”development.properties" and context:property-placeholder location="production.propetties”. On deployment, pass an argument to the Mule runtime to ensure the correct property placeholder is respected.

B. If deploying to Mule ESB, go to MMC and change the values of the deployed application's properties through a property administrator. If deploying to CloudHub, go to the web-based management console and change the values of the deployed application's properties from the property administrator.

C. Create a dynamic property placeholder by using a property in the location attribute. For example, context:property-placehdder location=”myApp-${env}.properties". On deployment, pass an argument to the Mule runtime to ensure the correct property file is loaded.

D. Create multiple XML files, each containing properties for the respective environment. On deployment, pass an argument to the Mule runtime to ensure the correct XML file is used in the application.