PDFVCE

Choose the version that fits your needs	PDF Version	Desktop Test Engine	Online Test Engine
Latest and Up-to-Date exam dumps with real exam questions answers.	✓	✓	✓
Get 12-Months free updates without any extra charges.	✓	✓	✓
Experience same exam environment before appearing in the certification exam.	✗	✓	✓
100% exam passing guarante in the first attempt.	✓	✓	✓
20% discount on more than one license and 30% discount on 5+ license purchases.	✗	✓	✓
100% secure purchase on SSL.	✓	✓	✓
Completely private purchase without sharing your personal info with anyone.	✓	✓	✓

Exam : **OMG-OCUP2-ADV300**

Title : OMG Certified UML Professional 2 (OCUP 2) - Advanced Level

Vendor : OMG

Version : DEMO

QUESTION NO: 1
Choose the correct answer: What is the value of EMOF?
A. It provides the ability to define instance models without defining Slots.
B. It is the metamodel used to specify other metamodels including UML 2.
C. It allows implementations, but these are not considered MOF-compliant.
D. It enables mapping of MOF models to implementations such as XMI for simple metamodels.
Answer: D
Explanation:
The value of Essential MOF (EMOF) lies in its ability to provide a straightforward framework for mapping MOF models to implementations such as XML Metadata Interchange (XMI) for simple metamodels2. This allows for easier integration and manipulation of MOF models in various platforms and tools.

QUESTION NO: 2
Choose the correct answer:
In addition to stereotypes, which UML elements can a profile define?
A. all of the UML elements
B. all of the UML Structured Classifiers
C. Class. DataType. Enumeration
D. Class. DataType. Enumeration. Object. Component
Answer: C
Explanation:
A UML profile can define several different elements to extend the UML for a specific domain, platform, or purpose. Besides stereotypes, a profile can define DataTypes and Enumerations, which can be used within Stereotypes to type the tagged values. A profile can also define Classes, which can be used as base classes for stereotypes. However, a profile does not typically define instances of elements, such as Objects, nor does it define composite structures, such as Components. The intent of a profile is to adapt existing metamodel elements through the use of stereotypes, tag definitions, and constraints rather than to create instances or define system architectures. This definition and use of a profile are aligned with the guidelines in the UML 2.x Superstructure Specification, which outlines what elements a profile can define within the UML framework.

QUESTION NO: 3
Choose the correct answer:
What is the difference between a ValueSpecificattonActlon and a CreateObjectAction?
A. There is no difference between the possible results of the CreateObjectAction and the ValueSpecificationAction
B. A ValueSpecificationAction cannot produce instances of Classifiers, but a CreateObjectAction only produces instances of Classifiers
C. When an InstanceValue is produced by a CreateObjectAction. the Structural Features can be assigned values, but when an InstanceValue is produced by a ValueSpecificationAction, the Structural Features have no values.

D. When an InstanceValue is produced by a ValueSpecificationAction, the Structural Features can be assigned values, but when an InstanceValue is produced by a CreateObjectAction, the Structural Features have no values.

Answer: B

Explanation:

The difference between ValueSpecificationAction and CreateObjectAction in UML is fundamental to the kinds of results they produce:

* A.Incorrect, as there is a fundamental difference between the outputs of these two actions.
* B.Correct. A ValueSpecificationAction evaluates to a value but does not produce instances of Classifiers, whereas a CreateObjectAction is specifically designed to instantiate classifiers.
* C.Incorrect, because it reverses the capabilities of the two actions.
* D.Also incorrect, because it provides the wrong association of features with actions.

References:

* UML Specification: Action section, where both ValueSpecificationAction and CreateObjectAction are defined.
* Further clarifications on these actions are provided in the UML 2.5 Documentation, especially in parts discussing Actions and Activities.

QUESTION NO: 4

Choose the correct answer:

Which can be added to a redefined operation?

A. scope

B. templates

C. parameters

D. preconditions

Answer: D

Explanation:

In UML, operations can have preconditions defined, which are constraints that must be true before the operation is invoked. When an operation is redefined in a subclass, it is possible to add new preconditions or alter existing ones. Redefining an operation does not typically allow for changes to its scope or the addition of templates, but the preconditions may be expanded to reflect the semantics of the subclass. This is in line with the behavioral specification of operations in UML, where preconditions are part of the behavioral contract of an operation, as described in the UML 2.x Superstructure Specification.

QUESTION NO: 5

Choose the correct answer:

In a model of a commercial transaction, actors might exchange euros, pesos, and dollars Which figure illustrates compliant use of UML information items for these currency exchanges?

Figure 1

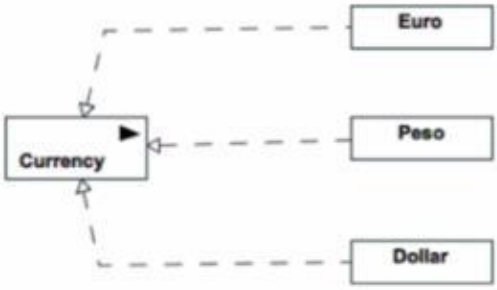


Figure 3

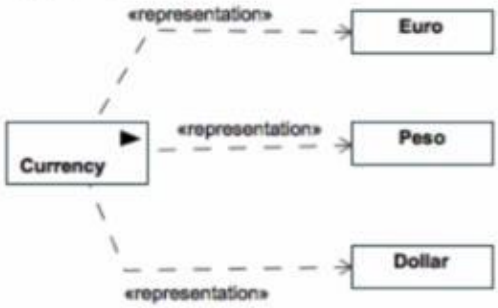


Figure 2

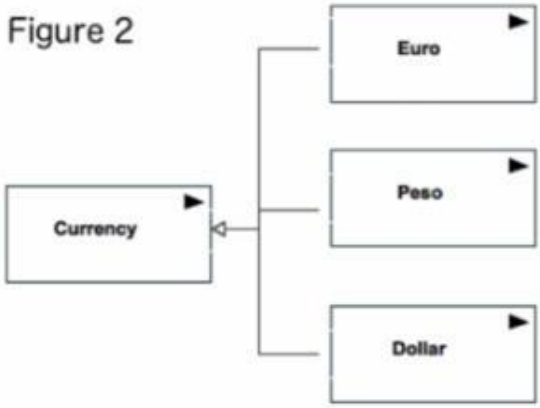
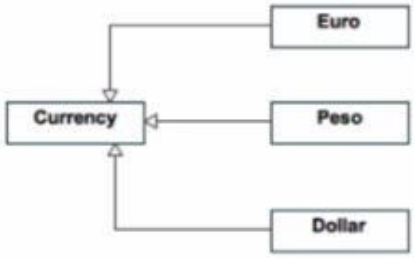


Figure 4



A. Figure 1
B. Figure 2
C. Figure 3
D. Figure 4
Answer: B
Explanation:
In UML, an InformationItem represents an abstraction of all those elements in a UML model that have an information-bearing feature. It is depicted as a classifier with the keyword «informationItem». An InformationItem does not have direct instances and serves as a mechanism to handle unspecified, untyped information in a model. In the context of a commercial transaction model, currencies such as euros, pesos, and dollars can be abstractly represented as InformationItems to signify that they are used as a form of data exchange but without specifying the concrete structure or data type. Figure 2 correctly uses InformationItem notation, with the «informationItem» keyword and the directed association pointing from the Currency InformationItem to the Euro, Peso, and Dollar, which are likely representations or manifestations of the Currency. This complies with the UML specification for representing abstract entities in models that are involved in the exchange or flow of information.

QUESTION NO: 6
Choose the correct answer:
Can State Machine Events have Features?
A. Yes. they can
B. Yes. they can. but only when the modeled Events also extend Classifier.

C. No. they can not. because State Machine Events are not Classifiers.

D. No. they can not. because UML State Machines do not have an Event concept.

Answer: C

Explanation:

State Machine Events in UML are triggers for transitions and do not classify as objects or classifiers; thus they do not inherently have features:

* A.Incorrect, because State Machine Events are not treated as classifiers with features.
* B.Incorrect, though it hints at an extended use case, State Machine Events do not extend classifiers in standard UML usage.
* C.Correct, State Machine Events are not Classifiers and therefore cannot have features such as properties or operations.
* D.Incorrect, because UML State Machines certainly have an Event concept, but these Events are not classified as having features.

References:

* UML Specification: State Machine chapter, specifically sections discussing the nature of events and triggers.
* Further insights can be found in the event and trigger management sections of the UML 2.5 Documentation.

QUESTION NO: 7

Choose the correct answer:

Which interpretation is valid when NamedElement A is the Supplier in a specialized Dependency having NamedElement B as the Client, and a Comment indicates that A and B participate in a transformation?

A. B is the transformation Realization of A.

B. A is the transformation Realization of B.

C. A depends on B.

D. A and B are part of an economic system where A consumes what B transforms.

Answer: C

Explanation:

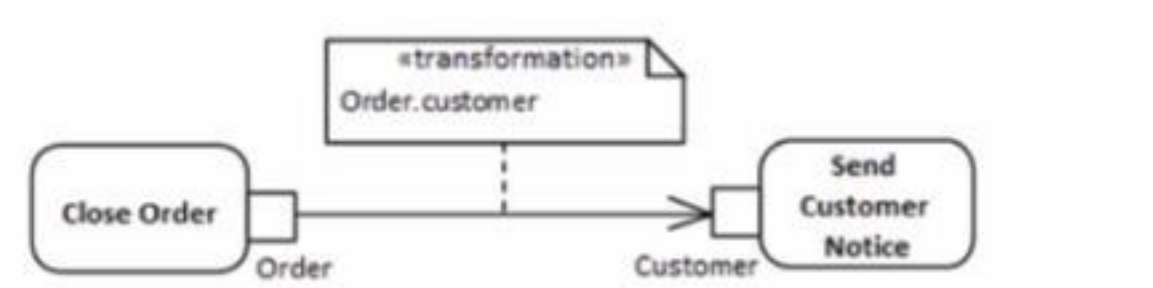
In UML, a Dependency is a relationship that signifies that one NamedElement, the client, depends on another NamedElement, the supplier, meaning that a change in the supplier could affect the client. If NamedElement A is the supplier and NamedElement B is the client in a Dependency relationship, and there is a Comment indicating that both participate in a transformation, the interpretation is that B (the client) depends on A (the supplier) for that transformation. The comment does not necessarily change the nature of the Dependency relationship; it simply adds additional information about the nature of their interaction. A transformation could mean that B transforms A's supplied element in some way, but in terms of UML Dependency relationships,it would still be interpreted as "A depends on B" or "B requires A for its transformation". This interpretation aligns with the UML 2.x Infrastructure and Superstructure specifications, which explain Dependencies and their meanings within the UML metamodel.

QUESTION NO: 8

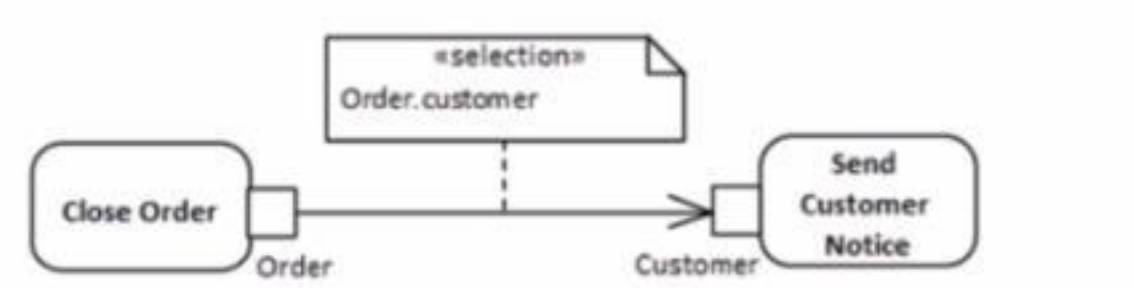
Choose the correct answer:

When an Order is closed, a Customer of that Order must be notified
Which diagram fragment correctly models this scenario?

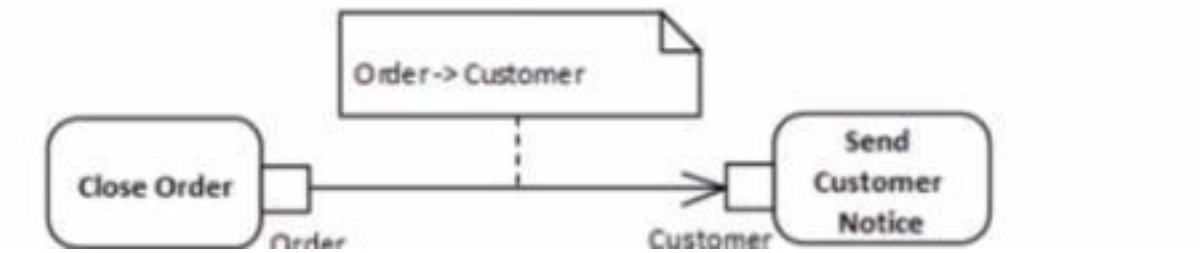
A.



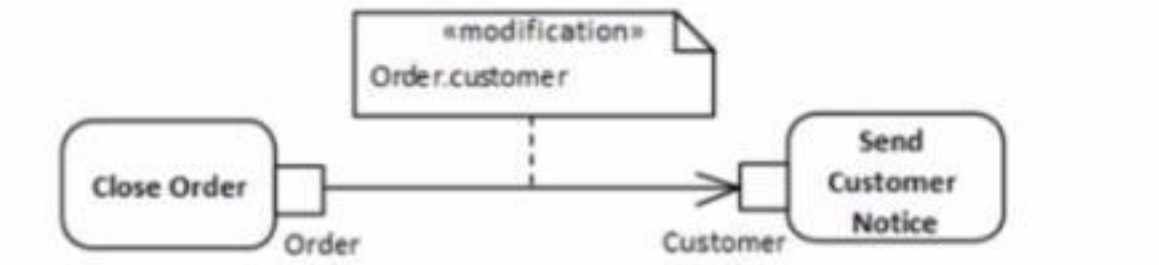
B.



C.



D.



Answer: C

Explanation:

The correct answer is C, which uses a transformation edge to indicate that when an order is closed, information is transformed or processed to notify the customer. This is expressed in UML with an object flow that has a transformation property, which specifies how an object is transformed from one state to another as it flows along the edge. Thisis appropriate for a scenario where closing an order leads to the creation of a customer notice, implying some form of transformation or processing of the order information to generate the notice.

References:

* UML 2.5 Specification: The UML 2.5 specification details how object flows with transformations can be used to show that one piece of data is transformed into another as it

flows from one action to another.

QUESTION NO: 9
Choose the correct answer:
Which statement is correct about redefinition of simple States?
A. A simple State cannot be redefined
B. A simple State can be redefined by excluding it in a redefined StateMachine.
C. A simple State can be redefined as a composite State with one or more Regions.
D. State Redefinition is not a UML capability.
Answer: B
Explanation:
In UML, a simple State can indeed be redefined, but the specifics of how it is redefined matter:
* Ais incorrect as redefinition of States, including simple States, is permitted.
* Bis correct. A simple State can be effectively removed or altered through redefinition in a derived StateMachine, which could include changing its type or characteristics.
* Csuggests that a simple State can be turned into a composite State. While theoretically possible in the context of redefinition, this option might be considered a significant alteration that could be misleading without additional context.
* Dis incorrect; UML does support State Redefinition.
References:
* UML Specification, particularly the sections on Redefinition and StateMachines.
* Detailed rules on state redefinition are outlined in the UML 2.5 Documentation, specifically under StateMachine specializations.