

Unparalleled InsuranceSuite-Developer Dumps Reviews Help You to Get Acquainted with Real InsuranceSuite-Developer Exam Simulation

27/12/2024, 14:00 InsuranceSuite Developer Fundamentals Test 2025 (Questions With 100% Correct Answers) A+ Graded Verified Flashcards | Quizlet

InsuranceSuite Developer Fundamentals Test 2025 (Questions With 100% Correct Answers) A+ Graded Verified

Practice questions for this set

Learn

Studied 7 terms

Nice work. You're doing brilliantly.

Continue studying in Learn

Terms in this set (41)

Logging	Process of recording application actions and state to a secondary interface used for: Application maintenance and troubleshooting Creating statistics relating to application usage Auditing by capturing significant events.
---------	---

<https://quizlet.com/968441558/insurance-suite-developer-fundamentals-test-2025-questions-with-100-correct-answers-a-graded-verified-flash-cards/7...> 1/12

P.S. Free & New InsuranceSuite-Developer dumps are available on Google Drive shared by Itcertkey:
<https://drive.google.com/open?id=1VhTs6emlPaMUyI3hG5yYrmlItkdrHr00>

Success in the test of the Associate Certification - InsuranceSuite Developer - Mammoth Proctored Exam (InsuranceSuite-Developer) certification proves your technical knowledge and skills. The InsuranceSuite-Developer exam credential paves the way toward landing high-paying jobs or promotions in your organization. Many people who attempt the Associate Certification - InsuranceSuite Developer - Mammoth Proctored Exam (InsuranceSuite-Developer) exam questions don't find updated practice questions. Due to this they don't prepare as per the current InsuranceSuite-Developer examination content and fail the final test.

Dear every one, please come on and check out free demo of Itcertkey exam dumps in PDF test files. Do you see the Guidewire InsuranceSuite-Developer free demo? Do not hesitate, go and free download it. You may be surprised to see the questions are very valuable. InsuranceSuite-Developer online test engine is a test soft for simulating the actual test environment which can offer you the interactive and interesting experience. Besides, InsuranceSuite-Developer online test engine is virus-free, so you can rest assured to install it and use it. You will be more confident to face your InsuranceSuite-Developer exam test with InsuranceSuite-Developer online test engine.

>> InsuranceSuite-Developer Dumps Reviews <<

Overcome Exam Challenges with InsuranceSuite-Developer Guidewire

InsuranceSuite-Developer Exam Questions

The quality of our InsuranceSuite-Developer exam questions is very high and we can guarantee to you that you will have no difficulty to pass the exam. The content of the questions and answers of InsuranceSuite-Developer study braindumps is refined and focuses on the most important information. To let the clients be familiar with the atmosphere and pace of the real exam we provide the function of stimulating the exam. Our expert team updates the InsuranceSuite-Developer training guide frequently to let the clients practice more. Every detail of our InsuranceSuite-Developer learning prep is perfect.

Guidewire Associate Certification - InsuranceSuite Developer - Mammoth Proctored Exam Sample Questions (Q103-Q108):

NEW QUESTION # 103

In ClaimCenter, the Desktop- > Claims page contains a ListView that is backed by a View Entity ClaimDesktopView.eti. The company would like to add a column to the Claim Validation Level in this List View. Following best practice, which of the following steps are required to fulfill this requirement?

- A. Add a typekey in ClaimDesktopView.etx with name: ValidationLevel_Ext and path: Claim ValidationLevel.
- B. Add a computedTypekey in ClaimDesktopView.etx with name: ValidationLevel_Ext and path: Claim ValidationLevel.
- C. Add a computedColumn in ClaimDesktopView.etx with name: ValidationLevel and path: Claim ValidationLevel.
- **D. Add a viewEntityTypekey in ClaimDesktopView.etx with name: ValidationLevel and path: Claim ValidationLevel.**

Answer: D

Explanation:

View Entities (.eti files ending in View) are specialized data model objects used to provide high-performance, read-only data for ListViews. They function similarly to a SQL View, joining multiple tables to retrieve only the specific fields needed for a display grid. When extending a base view entity like ClaimDesktopView, standard data model tags like column or typekey (Option B) are not used. Instead, Guidewire provides a specific set of "View Entity" tags. Because ValidationLevel is a typelist on the Claim entity, the correct tag to use in the .etx file is viewEntityTypekey (Option C). This tag tells the platform that the field is derived from a typelist and ensures that the UI renders it with the correct localized display names and icons.

The path attribute is equally critical; it defines the "traversal" from the view entity to the source data. Since ClaimDesktopView typically has a link to the Claim entity, the path Claim ValidationLevel correctly instructs the system to pull the value from the related Claim record. While best practices for standard entities usually mandate the _Ext suffix, view entities often mirror the base field name for clarity unless a collision occurs.

However, the use of the viewEntityTypekey tag is the fundamental architectural requirement for this task, as computedTypekey and computedColumn (Options A and D) are intended for logic-heavy Gosu expressions rather than simple database-backed path traversals.

NEW QUESTION # 104

What type of Assessment Check ensures that applications have monitoring and logging frameworks in place?

- A. Performance
- B. Security
- **C. Operations**
- D. Upgrades

Answer: C

Explanation:

As part of the Guidewire Cloud Standards and the SurePath methodology, various "Quality Gates" and Assessment Checks are performed to ensure a successful implementation. These checks are categorized into specific domains: Security, Performance, Upgrades, and Operations.

The Operations assessment check focuses on the long-term maintainability and "observability" of the application in a production environment. For a system to be considered operationally ready in Guidewire Cloud, it must have robust monitoring and logging

frameworks in place. This includes ensuring that the code utilizes the `gw.api.util.Logger` properly, that logs do not contain sensitive PII (as discussed in previous best practices), and that the system is correctly integrated with observability tools like Datadog for real-time alerting.

While Performance checks (Option C) might look at the speed of logging or the impact of profiling, the actual requirement to have these frameworks present and correctly configured falls under the Operations umbrella.

Security checks (Option B) focus on vulnerabilities and data encryption, and Upgrades (Option A) focus on the readiness of the codebase to be migrated to newer versions of the platform. Therefore, the "Operations" check is the specific gate that validates whether the technical support teams will have the necessary data (via logs and metrics) to manage the application effectively once it is live on the Cloud.

NEW QUESTION # 105

What configuration item is needed to add `ABCContact.Notes` to `PendingContactChangeView` following best practices?

- A. Set the value on the input widget to `PendingContactChangeView.ABCContact.Note`
- B. Create a viewEntity that includes Note
- C. Add a viewEntityType for Note to `PendingContactChangeView.eti`
- D. Add a viewEntityType for Note to `PendingContactChangeView.etx`

Answer: D

Explanation:

In Guidewire InsuranceSuite, View Entities are specialized data model objects used primarily to provide flattened, high-performance data for ListViews (LVs). They function similarly to a database view, joining multiple related entities into a single virtual record to minimize the number of database queries required when rendering a page.

The screenshot provided shows `PendingContactChangeView.eti`. As indicated by the "This is a read-only file

" warning in the Studio interface, this is a Base Application File. According to the Data Model Architecture and InsuranceSuite

Developer Fundamentals, developers must never modify base files directly. Direct modifications to `.eti` files are not upgrade-safe and would be overwritten during any future Guidewire platform update, leading to significant maintenance issues.

To extend the metadata of a base view entity, the best practice is to use an Extension File, which carries the

`.etx` suffix. By adding a viewEntityType for Note within a `PendingContactChangeView.etx` file, the developer instructs the system to merge this custom field into the existing base view entity at runtime. This allows the new field to be accessible in Gosu and PCFs as if it were part of the original definition, while keeping the custom code isolated for easy upgrades.

Option C is incorrect because bypassing the View Entity by traversing through `ABCContact.Note` directly in a PCF widget defeats the performance purpose of the View Entity and can lead to "N+1" query performance bottlenecks. Option D is incorrect because creating a separate view entity is unnecessary and redundant when the current view entity can be easily extended. Therefore, Option A is the only verified best practice for maintaining a scalable and upgradeable Guidewire configuration.

NEW QUESTION # 106

In the screenshot below

A developer has added a tab labeled `Delinquencies` to the tab bar of `BillingCenter`. This tab will contain several pages. The first page in the tab will display a summary of the currently-selected delinquency, the second page will show the associated policy, and the third page will show the associated account.

What PCF container will be used to configure this requirement?

- A. A location group
- B. A location
- C. A location ref
- D. A location ref iterator

Answer: A

Explanation:

In the Guidewire Page Configuration Framework (PCF), locations are organized into a hierarchical structure to manage navigation and user context. When a requirement involves grouping multiple related pages under a single entry point—such as a Tab in the Tab Bar or a sidebar menu—the correct container to use is a Location Group.

1. The Role of a Location Group

A Location Group is a "non-leaf" node in the PCF navigation tree. It does not display content itself; instead, it contains other locations (Pages, Popups, or even other Location Groups). In the context of the "Delinquencies" tab, the Location Group serves as

the parent container that defines:

- * The Tab entry in the top-level navigation.
- * The list of child pages (Summary, Policy, Account).
- * The navigation menu (usually appearing on the left side of the screen) that allows users to switch between these three pages.

2. Why Other Options are Incorrect

- * Option A (Location Ref): This is a widget inside a container (like a Location Group or a Section) that simply points to another location. It is a "pointer," not the organizational container itself.
- * Option C (Location Ref Iterator): This is used to dynamically generate a set of links or locations based on an array of data (e.g., a tab for each Open Claim). It is not the standard way to define a static three-page tab structure.
- * Option D (Location): This is a generic term that encompasses Pages, Popups, and Worksheets. A single "Location" (specifically a Page) can only display one set of data. It cannot manage the multiple-page navigation required by the "Delinquencies" tab.

According to the InsuranceSuite Developer Fundamentals guide, using a Location Group ensures that the user's context (like the selected Delinquency ID) is maintained as they click through the different sub-pages within that group. This provides a seamless UX where the application "remembers" which record is being inspected even as the view changes.

NEW QUESTION # 107

Which rule is written in the correct form for a rule which sets the claim segment and leaves the ruleset?

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

Answer: A

Explanation:

In the GuidewireGosu Rules engine, managing the logic flow within a ruleset is a fundamental skill for any developer. A ruleset is essentially a collection of "If-Then" statements that the application evaluates sequentially. When a business requirement dictates that an action should be taken—such as categorizing a claim by setting its Segment property—and then no further rules in that specific set should be processed, the developer must use the `actions.utility` object.

The correct method to terminate the current ruleset execution is `actions.exit()`. As shown in Option A, the logic must be ordered procedurally: first, the state of the entity is modified (`claim.Segment = TC_AUTO_LOW`), and then the `exit()` command is called to stop the engine from evaluating subsequent rules. Using the typecode constant (`TC_AUTO_LOW`) is the best practice for assignment, as it provides compile-time checking, whereas using a hardcoded string (Option B) is error-prone and discouraged in Guidewire development.

Furthermore, the placement of the exit command is critical. In Option C, the `actions.exit()` is placed before the assignment; this results in the rule terminating immediately, and the claim segment is never actually updated.

Option D is incorrect because `actions.stop()` is not the standard method for exiting a ruleset in the Gosu rule architecture. By following the pattern in Option A, developers ensure that once a "mutually exclusive" business condition is met and handled, the system efficiently moves to the next ruleset or stage in the claim lifecycle, preventing redundant processing or accidental overwrites of the segment value by lower-priority rules.

NEW QUESTION # 108

.....

The job with high pay requires they boost excellent working abilities and profound major knowledge. Passing the InsuranceSuite-Developer exam can help you find the job you dream about, and we will provide the best InsuranceSuite-Developer question torrent to the client. We are aimed that candidates can pass the exam easily. The study materials what we provide is to boost pass rate and hit rate, you only need little time to prepare and review, and then you can pass the InsuranceSuite-Developer Exam. It costs you little time and energy, and you can download the software freely and try out the product before you buy it.

InsuranceSuite-Developer Reliable Test Cram: https://www.itcertkey.com/InsuranceSuite-Developer_braindumps.html

Itcertkey InsuranceSuite-Developer Reliable Test Cram Data Saving, Do you want to obtain InsuranceSuite-Developer certification quickly at first attempt, You can download Guidewire Certified Associate InsuranceSuite-Developer free demo dump as you like, Once you purchase our InsuranceSuite-Developer: Associate Certification - InsuranceSuite Developer - Mammoth Proctored Exam braindumps PDF You can always download our latest dumps any time within one year, Guidewire InsuranceSuite-Developer Dumps Reviews The only thing you need to do is to download the software, log into your account, and start learning.

