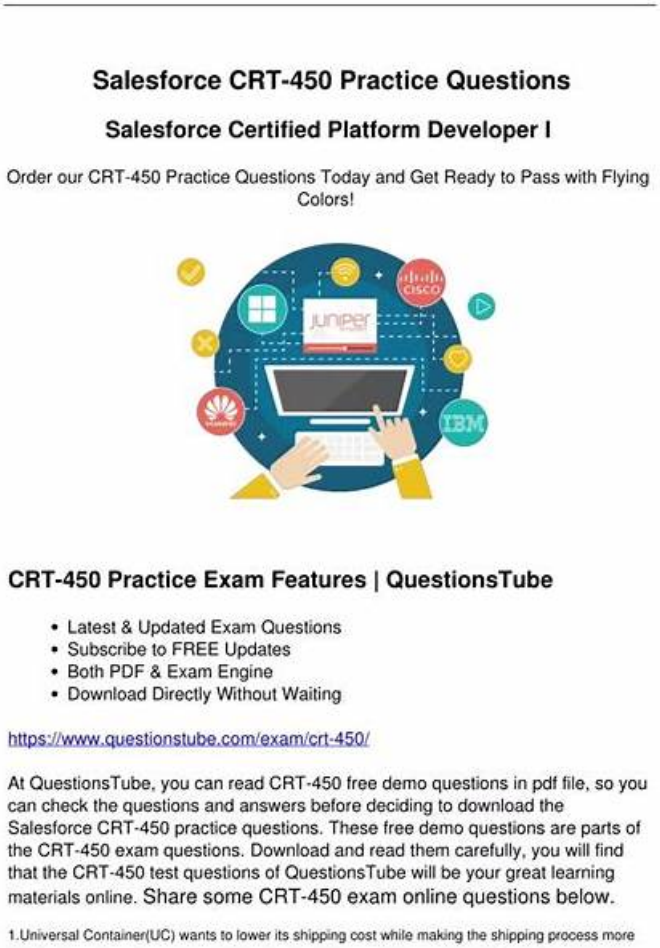
CRT-450 Practice Materials & CRT-450 Test Torrent & CRT-450 Pass King



BTW, DOWNLOAD part of BraindumpsPrep CRT-450 dumps from Cloud Storage: https://drive.google.com/open?id=1xb87B9qo1mZ_qnFEv_IXe3ySjWYEJLey

Our CRT-450 study materials are famous at home and abroad, the main reason is because we have other companies that do not have core competitiveness, there are many complicated similar products on the market, if you want to stand out is the selling point of needs its own. Our CRT-450 Study Materials with other product of different thing is we have the most core expert team to update our CRT-450 study materials , learning platform to changes with the change of the exam outline.

Salesforce CRT-450 exam is a valuable certification for individuals who want to start their career as a Salesforce platform developer. Salesforce Certified Platform Developer I certification is a foundation-level certification that assesses the essential knowledge and skills required to develop custom applications on the Salesforce platform. Passing the Salesforce CRT-450 Exam demonstrates a candidate's proficiency in Salesforce development and validates their expertise in developing custom applications using Apex and Visualforce.

>> New CRT-450 Exam Cram <<

New CRT-450 Exam Cram | Valid Salesforce Certified Platform Developer I 100% Free Test Valid

However, you should keep in mind that to get success in the Salesforce Certified Platform Developer I (CRT-450) exam is not an easy task. It is a challenging exam and not a traditional exam. But complete Salesforce CRT-450 exam preparation can enable you to crack the Salesforce CRT-450 exam easily. For the quick and complete Salesforce Certified Platform Developer I (CRT-450) exam preparation you can trust CRT-450 exam practice test questions. The Salesforce CRT-450 exam practice test questions have already helped many Salesforce CRT-450 exam candidates in their preparation and success and you can also trust "BraindumpsPrep" exam questions and start preparing today.

Passing the Salesforce CRT-450 Exam is a great achievement for any Salesforce developer. It demonstrates that the individual has a deep understanding of the Salesforce platform and is able to develop high-quality applications that meet the needs of their clients or organization. Additionally, earning this certification can help individuals advance their careers and increase their earning potential.

Salesforce Certified Platform Developer I Sample Questions (Q50-Q55):

NEW QUESTION # 50
As part of new feature development, a developer is asked to build a responsive application capable of responding to touch events, that will be executed on stateful clients.
Which two technologies are built on a framework that fully supports the business requirement? Choose 2 answers

- A. Lightning Web Components
- B. Visualforce Pages
- C. Aura Components
- D. Visualforce Components

Answer: A,C

NEW QUESTION # 51
What is the data type returned by the following SOSL search?
[FIND 'Acme*' IN NAME FIELDS RETURNING Account, Opportunity];

- A. Map<Id, sObject>
- B. List<List<Account>, List<Opportunity>>
- C. Map<sObject, sObject>
- D. List<List<sObject>>

Answer: D

NEW QUESTION # 52
Which two scenarios require an Apex method to be called imperatively from a Lightning web component? Choose 2 answer

- A. Calling a method that is not annotated with cacheable-true
- B. Calling a method that makes a web service callout
- C. Calling a method that is external to the main controller for the Lightning web component
- D. Calling a method with the click of a button

Answer: A,C

NEW QUESTION # 53
Refer to the following Apex code:
apex
Copy
Integer x = 0;
do {
x++;
} while (x < 1);
System.debug(x);
What is the value of x when it is written to the debug log?

- A. 0
- B. 1
- C. 2
- D. 3

Answer: C

Explanation:
Comprehensive and Detailed Explanation From Exact Extract:
To determine the value of x when it is written to the debug log, we need to analyze the Apex code step by step, focusing on the behavior of the do-while loop and how it affects the variable x. Let's break down the code execution systematically, referencing Salesforce's official Apex Developer Guide.
Code Analysis:
The given Apex code is:
apex
Copy
Integer x = 0;
do {
x++;
} while (x < 1);
System.debug(x);
Step 1: Initial State
* Integer x = 0;: The variable x is initialized to 0. The Apex Developer Guide states: "An Integer in Apex is a 32-bit number that does not include decimal points, initialized to 0 by default if no value is provided" (Salesforce Apex Developer Guide, Primitive Data Types). Here, x is explicitly set to 0.
Step 2: Understanding the do-while Loop
* A do-while loop in Apex executes the loop body at least once before evaluating the condition. The Apex Developer Guide explains: "The do-while loop executes the block of code in the do statement first, then checks the condition in the while statement. If the condition is true, the loop continues; otherwise, it exits" (Salesforce Apex Developer Guide, Loops).
* The loop body is:
apex
Copy
x++;
This increments x by 1 using the post-increment operator (++). The Apex Developer Guide confirms: "The ++ operator increments the value of the variable by 1" (Salesforce Apex Developer Guide, Expressions and Operators).
* The condition is:
apex
Copy
while (x < 1);
The loop continues as long as x < 1 evaluates to true.
Step 3: Loop Execution
* First Iteration:
* Initial value: x = 0.
* Execute the loop body: x++ # x becomes 1 (0 + 1).
* Evaluate the condition: x < 1 # 1 < 1 # false (since 1 is not less than 1).
* Since the condition is false, the loop exits after the first iteration.
* After the Loop:
* The value of x is now 1.
* The do-while loop guarantees at least one execution, which is why x is incremented once before the condition check fails.
Step 4: Debug Statement
* System.debug(x);: This writes the value of x to the debug log. At this point, x = 1. The Apex Developer Guide states: "System.debug outputs the value of the specified variable to the debug log for troubleshooting" (Salesforce Apex Developer Guide, System Class).
* Therefore, the debug log will show 1.
Evaluating the Options:
* A. 0: Incorrect. The initial value of x is 0, but the do-while loop increments x to 1 in the first iteration, and the loop exits because the condition x < 1 is false. The debug log shows the final value of x, which is 1.
* B. 2: Incorrect. The loop only runs once because after the first iteration, x becomes 1, and the condition x < 1 fails (1 < 1 is false). There is no second iteration to increment x to 2.
* C. 1: Correct. As calculated, the loop executes once, incrementing x from 0 to 1, and then exits because the condition x < 1 is false. The debug log outputs x = 1.

* D. 3: Incorrect. The loop does not run enough times to increment x to 3. It only runs once, setting x to 1.
Why Option C is Correct:
Option C (1) is correct because:
* The do-while loop executes the body (x++) exactly once, incrementing x from 0 to 1.
* The condition x < 1 evaluates to false when x = 1, causing the loop to exit after the first iteration.
* The System.debug(x) statement outputs the final value of x, which is 1.
* This behavior aligns with Apex loop semantics as defined in the Salesforce Apex Developer Guide.
Handling Typos:
* The code in the image contains a typo: "salesforce" is randomly inserted in the middle of the do-while loop. This appears to be an artifact of the image and not part of the intended code. For analysis, we ignore this text and treat the code as:
apex
Copy
Integer x = 0;
do {
x++;
} while (x < 1);
System.debug(x);
* There are no other syntactic issues in the code that affect the execution or outcome.
Example for Clarity:
To illustrate, here's how the code executes:
Integer x = 0; // x is 0
do {
x++; // First iteration: x becomes 1
} while (x < 1); // Condition: 1 < 1 # false, exit loop
System.debug(x); // Outputs: 1
If this code were run in a Salesforce org, the debug log would show:
DEBUG|1
References:
Salesforce Apex Developer Guide:
"Primitive Data Types" section: Defines the Integer type and its initialization.
"Loops" section: Explains the do-while loop's behavior, including guaranteed first execution.
"Expressions and Operators" section: Details the ++ increment operator.
"System Class" section: Describes System.debug for logging variable values.(Available at: https://developer.
salesforce.com/docs/atlas.en-us.apexcode.meta/apexcode/)
Platform Developer I Study Guide:
Section on "Developer Fundamentals": Covers Apex basics, including variables, loops, and debugging techniques.(Available at:
https://trailhead.salesforce.com/en/content/learn/modules/platform-developer-i- certification-study-guide)

NEW QUESTION # 54
Universal Containers decides to use purely declarative development to build out a new Salesforce application.
Which two options can be used to build out the business logic layer for this application?
Choose 2 answers

- A. Remote Actions
- B. Batch Jobs
- C. Record-Triggered Flow
- D. Validation Rules

Answer: C,D

Explanation:
When using purely declarative development to build the business logic layer in Salesforce, the following options are available:
Option A: Validation Rules
Reference:
"Validation rules verify that the data a user enters in a record meets the standards you specify before the user can save the record."
- Salesforce Help: Validation Rules
Option C: Record-Triggered Flow
"Record-triggered flows can be configured to run before or after the record is saved, making them a powerful declarative tool for business logic."
- Salesforce Help: Record-Triggered Flow

Why Other Options Are Incorrect:
Option B: Remote Actions require Apex code (annotated with @RemoteAction) and are used for making Apex methods callable from JavaScript, which is not declarative.
Option D: Batch Jobs are written in Apex and are used for processing large volumes of records asynchronously, which is programmatic development.

NEW QUESTION # 55
.....