

Salesforce PDI PDF Dumps - Effective Tips To Pass

materials on impulse, and make little progress even fail the exam unfortunately.

Free PDF Salesforce - PDI - Platform Developer I (PDI) Unparalleled Study Plan

Quality stands as the first priority to Exam-Killer, <https://www.exam-killer.com/platform-developer-i-pdi-material-10041.html> Trust us and you will get what you are dreaming. There are totally three versions of PDI practice materials which are the most suitable versions for you: PDF, Software and APP online versions.

You can free download the demo of our PDI exam questions to check the excellent quality on our website, Exam-Killer provides valid PDI exam questions & answers to help you achieve your goal as soon as possible if you want.

We are continuously promoting our Salesforce PDI exam solved questions after every 90 days.

Download Platform Developer I (PDI) Exam Dumps

NEW QUESTION 43

A developer created a weather app that contains multiple Lightning web components. One of the components, called Toggle, has a toggle for Fahrenheit or Celsius units. Another component, called Temperature, displays the current temperature in the unit selected in the Toggle component. When a user toggles from Fahrenheit to Celsius or vice versa in the Toggle component, the information must be sent to the Temperature component so the temperature can be converted and displayed.

What is the recommend way to accomplish this?

- A. The Toggle component should call a method in the Temperature component.
- B. Use Lightning Message Service to communicate between the components.
- C. Create a custom event to handle the communicate between the components.
- D. Use Lightning Message Service to communicate between the component.

Answer: C

NEW QUESTION 44

A developer created a Visualforce page and custom controller to display the account type field as shown below. Custom controller code:

```
public class customCtrlr{ private Account theAccount; public String actType; public customCtrlr() { theAccount = {SELECT Id, Type FROM Account WHERE Id = :apexPages.currentPage().getParameters().get('id')}; actType = theAccount.Type; } }
```

 Visualforce page snippet:

The Account Type is {!actType} The value of the account type field is not being displayed correctly on the page. Assuming the custom controller is properly referenced on the Visualforce page, what should the developer do to correct the problem?

- A. Add with sharing to the custom controller.
- B. Change theAccount attribute to public.
- C. Convert theAccount.Type to a String.
- D. Add a getter method for the actType attribute.

What's more, part of that Itexamguide PDI dumps now are free: <https://drive.google.com/open?id=1oEBQNIntiruhc3P5YjUV-aslaWf8MUcl>

It is of no exaggeration to say that sometimes a certification is exactly a stepping-stone to success, especially when you are hunting for a job. The PDI study materials are of great help in this sense. People with initiative and drive all want to get a good job, and if someone already gets one, he or she will push for better position and higher salaries. With the PDI test training, you can both have the confidence and gumption to ask for better treatment. To earn such a material, you can spend some time to study our PDI study torrent. No study can be done successfully without a specific goal and a powerful drive, and here to earn a better living by getting promotion is a good one.

Salesforce PDI Certification is an excellent investment for professionals who want to advance their career in the Salesforce ecosystem. It is a comprehensive certification that covers a wide range of topics and validates the skills required to develop custom applications on the Salesforce platform. With the growing demand for Salesforce developers, this certification has become increasingly popular and is highly valued by employers.

>> PDI Certification Materials <<

100% Pass Quiz PDI - Platform Developer I (PDI) Latest Certification Materials

The test software used in our products is a perfect match for Windows' PDI learning material, which enables you to enjoy the best learning style on your computer. Our PDI study materials also use the latest science and technology to meet the new requirements of authoritative research material network learning. Unlike the traditional way of learning, the great benefit of our PDI Study Materials are that when the user finishes the exercise, he can get feedback in the fastest time.

To prepare for the Salesforce PDI Certification Exam, developers can take advantage of various resources provided by Salesforce, such as online training courses, study guides, and practice exams. Developers can also join online communities and forums to connect with other developers and learn from their experiences. It is recommended that developers have at least six months to one year of experience developing applications on the Salesforce platform before attempting the certification exam.

Salesforce Platform Developer I (PDI) Sample Questions (Q165-Q170):

NEW QUESTION # 165

A developer is tasked to perform a security review of the ContactSearch Apex class that exists in the system. Within the class, the developer identifies the following method as a security threat:

```
List<Contact> performSearch(String lastName) {  
    return Database.query('SELECT Id, FirstName, LastName FROM Contact WHERE LastName  
    Like  
    ?'+lastName+'?');  
}
```

What are two ways the developer can update the method to prevent a SOQL injection attack?

Choose 2 answers

- A. Use a regular expression expression on the parameter to remove special characters,
- **B. Use the escapeSingleQuotes method to sanitize the parameter before its use.**
- C. Use the @ReadOnly annotation and the with sharing keyword on the class
- **D. Use variable binding and replace the dynamic query with a static SOQL.**

Answer: B,D

NEW QUESTION # 166

A lead developer creates a virtual class called "OrderRequest". Consider the following code snippet:

```
public class CustomerOrder(  
    //code implementation
```

How can a developer use the OrderRequest class within the CustomerOrder class?

- A. @Implements (class="OrderRequest")
public class Customerorder
- B. public class CustomerOrder implements Order
- **C. public class CustomerOrder extends OrderRequest**
- D. Extends (class="OrderRequest")
public class CustomerOrder

Answer: C

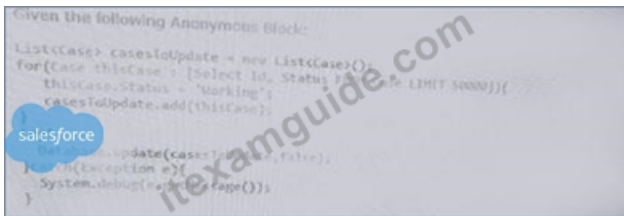
Explanation:

In Apex, a class can extend a virtual class (similar to abstract classes in Java) by using the extends keyword.

This allows the subclass (CustomerOrder) to inherit properties and methods from the parent virtual class (OrderRequest).

NEW QUESTION # 167

Given the following Anonymous Block:



Which one do you like?

What should a developer consider for an environment that has over 10,000 Case records?

- A. The transaction will fail due to exceeding the governor limit.
- B. The try/catch block will handle any DML exceptions thrown.
- C. The try/catch block will handle exceptions thrown by governor limits.
- **D. The transaction will succeed and changes will be committed.**

Answer: D

NEW QUESTION # 168

Universal Containers wants to ensure that all new leads created in the system have a valid email address. They have already created a validation rule to enforce this requirement, but want to add an additional layer of validation using automation.

What would be the best solution for this requirement?

- A. Use an Approval Process to enforce the completion of a valid email address using an outbound message action,
- B. Use a before-save Apex trigger on the Lead object to validate the email address and display an error message if it is invalid
- **C. Use a custom Lightning Web component to make a callout to validate the fields on a third party system.**
- D. Submit a REST API Callout with a JSON payload and validate the fields on a third party system

Answer: C

Explanation:

To add an additional layer of validation for email addresses on Lead records by using automation and possibly an external validation service, the best solution is to use a custom Lightning Web Component (LWC).

Option C: Use a custom Lightning Web Component to make a callout to validate the fields on a third-party system Real-Time

Validation: An LWC can perform client-side validation and make callouts to external systems to validate the email address as the user inputs data.

User Experience: Provides immediate feedback to the user, improving data quality at the point of entry.

Limitations: This solution works when data is entered through the UI using the LWC component.

Reference:

Lightning Web Components

Making Callouts from Lightning Components

Why Other Options are Less Optimal or Incorrect:

Option A: Use an Approval Process to enforce the completion of a valid email address using an outbound message action Not Suitable: Approval Processes are not intended for data validation.

Outbound Messages: Used to send data to external systems but do not enforce validation during record creation.

Option B: Submit a REST API Callout with a JSON payload and validate the fields on a third-party system Vague Implementation:

Does not specify where or how the callout is made.

Technical Constraints: Making synchronous callouts during record save (e.g., from a trigger) is not permitted.

Option D: Use a before-save Apex trigger on the Lead object to validate the email address and display an error message if it is invalid Callout Limitations: Apex triggers cannot make callouts in a before-save context.

Asynchronous Callouts: Using asynchronous methods (e.g., @future) cannot prevent the record from being saved.

Apex Triggers and Callouts

Future Methods

Conclusion:

Using a custom Lightning Web Component is the best solution to add an additional layer of validation during data entry.

It allows for real-time validation via callouts to a third-party system.

While it doesn't cover records created via other means (API, data import), the existing validation rule would still enforce that an email address is provided.

NEW QUESTION # 169

A developer created a Visualforce page and custom controller to display the account type field as shown below. Custom controller code: `public class customCtrl{ private Account theAccount; public String actType; public customCtrl() { theAccount = [SELECT Id, Type FROM Account WHERE Id = :apexPages.currentPage().getParameters().get('id')]; actType = theAccount.Type; } }` Visualforce page snippet: The Account Type is `{!actType}` The value of the account type field is not being displayed correctly on the page. Assuming the custom controller is properly referenced on the Visualforce page, what should the developer do to correct the problem?

- A. Add a getter method for the `actType` attribute.
- B. Convert the `Account.Type` to a `String`.
- C. Change the `Account` attribute to `public`.
- D. Add `with sharing` to the custom controller.

Answer: A

Explanation:

https://developer.salesforce.com/docs/atlas.en-us.pages.meta/pages/pages_controller_custom.htm

NEW QUESTION # 170

• • • • •

PDI Vce Torrent: https://www.itexamguide.com/PDI_braindumps.html

- [illegible]

2026 Latest Iteexamguide PDI PDF Dumps and PDI Exam Engine Free Share: <https://drive.google.com/open?id=1oEBQNIntiruhc3P5YjUV-aslaWf8MUcl>