

21.

A ListView shows contacts related to a Claim. When a user clicks the contact name in a text cell, the UI should open a Worksheet showing details of that contact. The elementName property in the row iterator is currentContact.

Which is the correct approach?

- Set the actionAvailable property on the atomic widget to ContactWS.push(currentContact)

- Set the Action property on the atomic widget to ContactWS.goInWorkspace(currentC

Set the Action property on the



22.

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.

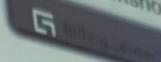
What PCF container will be used to configure this requirement?

- A location
- A location group



22.

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 set 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 location
- A location group
- A location ref iterator
- A location ref



24.

An insurance carrier needs the ability to capture information for different kinds of watercraft, such as power boats, personal water craft, sailboats, etc. The development team has created a Watercraft_Ext entity with subtype entities to store the distinct properties of each type of watercraft.

Which represents the best approach to provide the ability to edit the data for watercraft in the User Interface?

-
- Create a set of Modal Pages for each type of watercraft

- Create a single page for all watercraft types with the visibility of fields distinct to the type of



23.

An insurer imports information used to calculate replacement values for property claims on a monthly basis. The information is summarized in a non-editable list. A business analyst has presented a new requirement to support editing individual records when new information is received between scheduled imports.

What location type will satisfy this requirement?

- Edit buttons
- A location group
- A popup



24.

An insurance carrier needs the ability to capture information for different types of watercraft, such as power boats, personal water craft, sailboats, etc. The development team has created a Watercraft Entity with various references to store the distinct properties of each type of watercraft. Which represents the best approach to provide the ability to edit the data for watercraft in the User Interface?

- Create a set of Modal Pages for each type of watercraft
- Create a single page for all watercraft types with the visibility of Fields defined by the type of watercraft controlled at the widget level
- Create a Detail View for the common properties of all watercraft and a set of InputSets for the distinct property of each watercraft
- Create a Modal Detail View for each type of watercraft, duplicating common fields across each Detail View

Choose 1 option



30°C

Mostly cloudy



Search

24.

An insurance carrier needs the ability to capture information for different kinds of watercraft, such as power boats, personal water craft, sailboats, etc. The development team has created a Watercraft_Ext entity with subtype entities to store the distinct properties of each type of watercraft.

Which represents the best approach to provide the ability to edit the data for watercraft in the User Interface?

-
- Create a set of Modal Pages for each type of watercraft
 - Create a single page for all watercraft types with the visibility of fields distinct to the type of



25. Which scenario follows best practices for user interface field-level validation?

- Store different social media profile addresses, regardless of the social media site involved
- Proposed changes to user passwords contain only alphanumeric characters

- The city and state populate automatically whenever a US user enters their ZIP code

- The interest rate field is 0 whenever a down payment is greater than €5000



26.

An insurer doing business globally wants to use a validation expression to verify that a contact's postal code is a real postal code for the country specified in the contact's address.

A developer has created a method with the signature
`validatePostalCode(anAddress : Address) : boolean`, which returns true if and only if the postal code is valid.

What would be the correct validation expression?

-
- `!validatePostalCode(anAddress)`
 - `validatePostalCode(anAddress) ==`
 - `true`



26. An insurer doing business globally wants to use a validation expression to verify that a contact's postal code is a real postal code for the country mentioned in the contact's address.

A developer has created a method with the signature `validatePostalCode(Address : Address) : boolean`, which returns true if and only if the postal code is valid.

What would be the correct validation expression?

- `!validatePostalCode(anAddress)`
- `validatePostalCode(anAddress) == true`
- `validatePostalCode(anAddress) == null`
- `validatePostalCode(anAddress) ? null : false`

Choose 1 option.



26.

An insurer doing business globally wants to use a validation expression to verify that a contact's postal code is a real postal code for the country specified in the contact's address.

A developer has created a method with the signature

`validatePostalCode(anAddress : Address) : boolean`, which returns true if and only if the postal code is valid.

What would be the correct validation expression?

-
- `!validatePostalCode(anAddress)`
 - `validatePostalCode(anAddress) ==`
 - `true`



30°C

Mostly cloudy

Given the method below:

```
public function FriendlyGreeting  
(name: String) : String {  
    if (name == null or name.length == 0)  
        throw "Requires a non-empty string"  
    return "Hello, " + name + "!"  
}
```

What best practice is violated in the code?

The name of the method should be

- "friendlyGreeting"

A method should not throw an

- exception

"return" cannot return more than one

- object; the method should make and



30°C

Mostly cloudy

27.

Given the method below:

```
public function FriendlyGreeting (name: String): String {  
    if (name == null or name.length == 0) throw "Requires a non-empty name"  
    return "Hello, " + name + "!"  
}
```

What best practice is violated in the code?

- The name of the method should be "friendlyGreeting"
- A method should not throw an exception
 - "return" cannot return more than one object; the method should make and return a String
- The "throw" statement is used without a matching "catch" statement

Choose 1 option.



1 30°C
Mostly cloudy



27. Given the method below:

```
public function FriendlyGreeting  
(name: String) : String {  
    if (name == null or name.length == 0)  
        throw "Requires a non-empty string!"  
    return "Hello, " + name + "!"  
}
```

What best practice is violated in the code?

The name of the method should be

- "friendlyGreeting"

A method should not throw an

- exception

"return" cannot return more than one

- object; the method should make and



28.

An insurer has extended the ABContact entity in ContactManager with an array of Notes to capture information of interest about the contact over time. A developer has been asked to write a function to process all the notes for a given contact.

Which code satisfies the requirement and follows best practices?

```
for ( i =  
1..lanABContact.Notes.length ) {
```

//do something
}

```
for ( note in anABContact.Notes ) {  
    //do something
```



28.

An insurer has extended the ABCContact entity in ContactManager with an array of Notes to capture information of interest about the contact. Over time, a developer has been asked to write a function to process all the notes for a given contact.

Which code satisfies the requirement and follows best practices?

-
- for (i = 1..JanABCContact.Notes.length) {
 //do something
}
 - for (note in anABCContact.Notes) {
 //do something
}
 - while (exists (note in anABCContact.Notes)) {
 //do something
}

var aNote = anABCContact.Notes.firstWhere(\note -> note.Author == null)



28. An insurer has extended the ABContact entity in ContactManager with an array of Notes to capture information of interest about the contact over time. A developer has been asked to write a function to process all the notes for a given contact.
- Which code satisfies the requirement and follows best practices?

```
for (i =
```

```
1..|anABContact.Notes.length| {
```

 //do something
 }

```
    for ( note in anABContact.Notes | {
```

 //do something



29.

What is a purpose of logging in deployed systems that follows best practices?

- Capturing the significant events for auditing
- Recording the details of a user's login credentials
- Saving the runtime database queries and procedures
- Troubleshooting payments by logging bank account numbers

Choose 1 option.



29. What is a purpose of logging in deployed systems that follows best practices?
- Capturing the significant events for auditing
 - Recording the details of a user's login credentials
 - Saving the runtime database queries and procedures
 - Troubleshooting payments by logging bank account numbers

Choose 1 option.

 Previous page



 Search

29. What is a purpose of logging in deployed systems that follows best practices?

- Capturing the significant events for auditing
- Recording the details of a user's login credentials
- Saving the runtime database queries and procedures
- Troubleshooting payments by logging bank account numbers

Choose 1 option.



Search

30.

Which logging statement follows best practice?

- ```
If(_logger.DebugEnabled) {
 _logger.debug(logPrefix +
 someReallyExpensiveOperation())
}
```
- ```
_logger.error(DisplayKey.get("Web.C  
e.Message))
```
- ```
If(_logger.InfoEnabled) {
 _logger.debug("Adding
 '${contact.PublicID}' to
 ContactManager")
}
```
- ```
_logger.info(logPrefix +  
    " [Address#AddressLine1=" +  
    address.getAddressLine1() + "
```



30.

Which logging statement follows best practice?

- `If(_logger.DebugEnabled) {
 _logger.debug(logPrefix + someReallyExpensiveOperation())
}`
- `_logger.error(DisplayKey.get("Web.ContactManager.Error.General.message")
e.Message)`
- `If(_logger.InfoEnabled) {
 _logger.debug("Adding ${contact.PublicID} to ContactManager")
}`
- `_logger.info(logPrefix +
 "[Address#AddressLine1=" + address.AddressLine1 +
 "] [Address#City" + address.City +
 "] [Address#State" + address.State +"]")`

Choose 1 option.



Which logging statement follows best practice?

- ```
If(_logger.DebugEnabled) {
 _logger.debug(logPrefix +
 someReallyExpensiveOperation())
}
```
- ```
_logger.error(DisplayKey.get("Web.C  
e.Message))
```
- ```
If(_logger.InfoEnabled) {
 _logger.debug("Adding
 '${contact.PublicID}' to
 ContactManager")
}
```
- ```
_logger.info(logPrefix +  
    "[Address#AddressLine1=" +  
    contact.AddressLine1 + " " +  
    contact.AddressLine2 + " " +  
    contact.City + ", " +  
    contact.State + ", " +  
    contact.PostalCode + " " +  
    contact.Country + "]")
```



Which log message follows logging best practices in production?

- The claim was created successfully for: Ray Newton, email: rnewton@foo.com, vehicle: white 2022 Toyota Camry.
- ERROR! The claim was not created because the database connection was lost.
- The personal auto claim was created for Jane Smith with driver's license AD45678.
- [Method=ClaimProcess#createClaim
[Claim#PublicID=00001234]
was created.

1 30°C
Mostly cloudy

Q1. Which log message follows logging best practices in production?

- The claim was created successfully for: Ray Newthon, email: raynewthon@xyz.com, vehicle: white 2022 Toyota Camry.
- ERROR! The claim was not created because the database connection was lost.
- The personal auto claim was created for Jane Smith with driver's license 123456789.
- [Method=ClaimProcess#createClaim] [Claim#PublicID->00001234] was created.

Choose 1 option.

◀ Previous page

1 30°C
Cloudy Mostly cloudy

Which log message follows logging best practices in production?

The claim was created successfully for: Ray Newton, email: mewton@foo.com, vehicle: white 2022 Toyota Camry.

ERROR! The claim was not created because the database connection was lost.

The personal auto claim was created for Jane Smith with driver's license AD45678.

[Method=ClaimProcess#createClaim
[Claim#PublicID=00001234]

was created.



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

CONDITION (claim : entity.Claim)
return claim.Segment == null

- ACTION (claim : entity.Claim, actions : gw.rules.ACTION)
claim.Segment == TC_AUTO_LOW

actions.exitRuleset()

CONDITION (claim : entity.Claim)
return claim.Segment == null



32.

Which rule is written in the correct form for a rule which sets the value of the variable 'segment' and leaves the ruleset?

CONDITION (claim : entity.Claim)
return claim.Segment == null

- ACTION (claim : entity.Claim, actions : gw.rules.ACTION)
claim.Segment == TC_AUTO_LOW
actions.exitRuleset()

CONDITION (claim : entity.Claim)
return claim.Segment == null

- ACTION (claim : entity.Claim, actions : gw.rules.ACTION)



```
    claim : entity.Claim, actions : gw.rules.ACTION)
    if (claim.Segment == null)
        (claim.Segment = TC_AUTO_LOW)
    actions.exit()
```

CONDITION (claim : entity.Claim)
return claim.Segment = TC_AUTO_LOW

ACTION (claim : entity.Claim, actions : gw.rules.ACTION)
actions.exit()

Choose 1 option.

◀ Previous page



1 30°C
Mostly cloudy

```
CONDITION (claim : entity.Claim)
return true

ACTION (claim : entity.Claim, actions
: gw.rules.ACTION)
If (claim.Segment == null)
{claim.Segment = TC_AUTO_LOW}
actions.exit()
```

```
CONDITION (claim : entity.Claim)
return claim.Segment =
TC_AUTO_LOW
```

```
O ACTION (claim : entity.Claim, actions
: gw.rules.ACTION)
actions.exit()
```



An insurance carrier requires that a claim be flagged as potential fraud when the Loss Date on a claim is changed, and a review activity and history entry be created.

Which configuration will accomplish this?

Create a Preupdate Rule that flags the claim and creates a history entry;

- a ClaimException Rule will create an escalation activity for the supervisor

Create a Preupdate Rule that checks for a change to the Loss Date field

- and flags the claim and creates the review activity and history entry

Create a PostSetup Rule that checks



```
claim : entity.Claim, actions : gw.rules.ACTIONS  
claim.Segment = TC_AUTO_LOW  
actions.exit()
```

```
CONDITION(claim : entity.Claim)  
return true
```

```
● ACTION (claim : entity.Claim, actions : gw.rules.ACTIONS)  
If (claim.Segment == null)  
{claim.Segment = TC_AUTO_LOW}  
actions.exit()
```

```
CONDITION (claim : entity.Claim)  
return claim.Segment = TC_AUTO_LOW
```

```
○ ACTION (claim : entity.Claim, actions : gw.rules.ACTIONS)
```



An insurance carrier requires that a claim be flagged as potential fraud when the Loss Date on a claim is changed, and a review activity and history entry is created.

Which configuration will accomplish this?

- Create a Preupdate Rule that flags the claim and creates a history entry as ClaimException Rule will create an escalation activity for the supervisor
- Create a Preupdate Rule that checks for a change to the Loss Date field and Flags the claim and creates the review activity and history entry
- Create a PostSetup Rule that checks for a change to the Loss Date field and Flags the claim, which creates a supervisor activity and history entry
- Create a Validation Rule to determine if Policy is in force on the new Loss Date and only take action if the new Loss Date is outside the Policy effective date

Choose 1 option.



33.

An insurance carrier requires that a claim be flagged as potential fraud when the Loss Date on a claim is changed, and a review activity and history entry be created.

Which configuration will accomplish this?

- Create a Preupdate Rule that flags the claim and creates a history entry.
- a ClaimException Rule will create an escalation activity for the supervisor

- Create a Preupdate Rule that checks for a change to the Loss Date field
- and flags the claim and creates the review activity and history entry

Create a PostSetup Rule that checks



34.

A developer is adding an enhancement to Claim in the package: si.cc.enhancement.entity. Claim is an existing entity in ClaimCenter. A getter property is added to calculate the total cost. Which adheres to the best practices naming conventions?

- ClaimEnhancement_Ext.gsx,
totalCost()
- ClaimEnhancement.gsx,
TotalCost_Ext()
- ClaimEnhancement.gsx, TotalCost()
- ClaimEnhancement_Ext.gsx,
TotalCost()



35.

A business analyst has a requirement to use either the EmailAddress1 or EmailAddress2 field on ABCContact as the email address, which will be used in multiple places.

```
{  
    var emailAddress =  
        this.EmailAddress1  
    if(StringUtil.isEmpty(emailAddress))  
        emailAddress = this.EmailAddress2  
    return emailAddress  
}
```

Which enhancement component signature is appropriate for this implementation and follows best practice?



35.

A business analyst has a requirement to use either the EmailAddress or EmailAddress2 field on ABCContact as the email address, which will be used in multiple places.

```
{  
    var emailAddress = this.EmailAddress;  
    if(StringUtil.isEmpty(emailAddress))  
        emailAddress = this.EmailAddress2;  
    return emailAddress  
}
```

Which enhancement component signature is appropriate for this implementation and follows best practice?

- function emailAddress_Ext(): String
- property get EmailAddress_Ext(): String
- function determineEmailAddress_Ext()



Business analyst has a requirement to use either the EmailAddress1 or EmailAddress2 field on ABContact as the email address, when will be used in multiple places.

```
{  
    var emailAddress = this.EmailAddress1  
    if(StringUtil.isEmpty(emailAddress) )  
        emailAddress = this.EmailAddress2  
    return emailAddress  
}
```

Which enhancement component signature is appropriate for this implementation and follows best practice?

- function emailAddress_Ext(): String
- property get EmailAddress_Ext(): String
- function determineEmailAddress_Ext()
- property set EmailAddress_Ext(emailAddress: String)



Q Search

36.

An insurer stores the date a company was established in the company records. A business analyst identified a new requirement to calculate a company's years in business at the time a loss occurred. The years in business will be determined using the date established field and the claim date of loss.

The image below shows the Contact structure in the data model:

Which configuration steps will satisfy the requirement? (Select two)

Create a new enhancement class for the Company entity under the



36.

An insurer stores the date a company was established in the company record. A business analyst identified a new requirement to calculate a company's years in business at the time a loss occurred. The years in business will be determined using the date established field and the claim date of loss.

The image below shows the Contact structure in the data model.

Contact (cc_contact) (delegates to)

Subtypes

Code	Name
Company	Company
Person	Person
Place	Place

Which configuration steps will satisfy the requirement? (Select two)

Create a new enhancement class for the Company entity under the requirement.

Create a function to calculate the years in business in the Contact entity.



- These configuration steps will satisfy the requirement:
- Create a new enhancement class for the Company entity under the package
 - Create a function to calculate the years in business in a Company enhancement
 - Create a setter property to calculate the years in business in the Contact enhancement
 - Create a function to calculate the years in business in a Contact class under the gw package
 - Create a new enhancement class for the Contact entity under the package
 - Create a getter property to calculate the years in business in a Company enhancement

Choose 2 options.

◀ Previous page



37.

A developer is creating an enhancement class for the entity AuditMethod_Ext in PolicyCenter for an insurer, Succeed Insurance. Which package structure of the gosu class and function name follows best practice?

-
- si.pc.enhancements.entity,
 determineAuditType()
 - gw.job.audit, determineAuditType()
 - si.pc.enhancements.entity,
 determineAuditType_Ext()
 - gw.entity.enhancement,
 determineAuditType_Ext()



37. A developer is creating an enhancement class for the entity `enhancement` in PolicyCenter for an insurer, Succeed Insurance. Which package structure of the gosu class and function name is the best practice?

- `si.pc.enhancements.entity.determineAuditType()`
- `gw.job.audit, determineAuditType()`
- `si.pc.enhancements.entity.determineAuditType_Ext()`
- `gw.entity.enhancement, determineAuditType_Ext()`

Choose 1 option.

 Previous page

 30°C
Mostly cloudy

Per is creating an enhancement class for the entity AuditMethod_Ext in PolicyCenter for an insurer, Succeed Insurance.

Which package structure of the gosu class and function name follows best practice?

- si.pc.enhancements.entity,
 determineAuditType()
- gw.job.audit, determineAuditType()
- si.pc.enhancements.entity,
 determineAuditType_Ext()
- gw.entity.enhancement,
 determineAuditType_Ext()



38.

The following Gosu statement is the Action part of a validation rule:

```
claim.rejectField("State",  
    TC_PAYMENT,
```

```
DisplayKey.get("Rules.Validation.Claim  
    null, null)
```

It produces the following compilation error:

Gosu compiler: Wrong number of arguments to function

```
rejectField(java.lang.String,  
    typekey.ValidationLevel,  
    java.lang.String,  
    typekey.ValidationLevel,  
    java.lang.String). Expected 5, got 3
```

What needs to be added to or deleted?



38.

The following Gosu statement is the Action part of a validation rule:
claim.rejectField("State", TO_PAYMENT,
DisplayKey.get("Rules.Validation.Claim.NoteDraft",
null, null))

It produces the following compilation error:

Gosu compiler: Wrong number of arguments to function
rejectField(java.lang.String, typekey.ValidationLevel, java.lang.string,
typekey.ValidationLevel, java.lang.string). Expected 5, got 3.

What needs to be added to or deleted from the statement to clear the error?

- A left parenthesis must be deleted.
- The word "State" must be replaced with a DisplayKey.
- A right parenthesis must be added.
- The two nulls must be replaced with a typekey and a string.



38.

The following Gosu statement is the Action part of a validation rule:

```
claim.rejectField("State",  
TC_PAYMENT,
```

```
DisplayKey.get("Rules.Validation.Claim  
null, null)
```

It produces the following compilation error:

Gosu compiler: Wrong number of arguments to function

```
rejectField(java.lang.String,  
typekey.ValidationLevel,
```

```
java.lang.String,
```

```
typekey.ValidationLevel,
```

```
java.lang.String). Expected 5, got 3
```

What needs to be added to or deleted



30°C

Mostly cloudy

- var notesQuery = gw.api.database.Query.make(Note)
notesQuery.compare(Note#Topic, Equals, NoteTopicType.TC_DENIAL)
var denialNotes = notesQuery.select()
- var notesQuery = gw.api.database.Query.make(Note)
notesQuery.compare(Note#Topic, Equals, NoteTopicType.TC_DENIAL)
notesQuery.compare(Note#Claim, Equals, claim)
var denialNotes = notesQuery.select()
- var denialNotes = claim.Notes.where(elt > elt.Topic == NoteTopicType.TC_DENIAL)
- var notesQuery = gw.api.database.Query.make(Note)
- var denialNotes = notesQuery.select().where(elt > elt.Topic == NoteTopicType.TC_DENIAL)



According to best practices, which logic returns notes with the topic of denial and filters on the database?

- var notesQuery = gw.api.database.Query.make(Note)
notesQuery.compare(Note#Topic, Equals, NoteTopicType.TC_DENIAL)
var denialNotes = notesQuery.select()
- var notesQuery = gw.api.database.Query.make(Note)
notesQuery.compare(Note#Topic, Equals, NoteTopicType.TC_DENIAL)
notesQuery.compare(Note#Claim, Equals, claim)
var denialNotes = notesQuery.select()
- var denialNotes = claim.Notes.where(elt => elt.Topic==NoteTopicType.TC_DENIAL)
- var notesQuery = gw.api.database.Query.make(Note)
var denialNotes = notesQuery.select().where(elt => elt.Topic==NoteTopicType.TC_DENIAL)



39.

Given the following code example:

```
var query = gw.api.database.Query.make(Claim)
query.compare(Claim#ClaimNumber, Equals, "123-45-6789")
var claim = query.select().AtMostOneRow
```

According to best practices, which logic returns notes with the topic of denial and filters on the database?

- var notesQuery = gw.api.database.Query.make(Note)
notesQuery.compare(Note#Topic, Equals, NoteTopicType[Denial])
var denialNotes = notesQuery.select()
- var notesQuery = gw.api.database.Query.make(Note)
notesQuery.compare(Note#Topic, Equals, NoteTopicType[Denial])
notesQuery.compare(Note#Claim, Equals, claim)
var denialNotes = notesQuery.select()
- var denialNotes = claim.Notes.where(\elt -> elt.Topic==NoteTopicType[Denial])



39.

Given the following code example:

```
var query =  
  gw.api.database.Query.make(Claim)  
  query.compare(Claim#ClaimNumber,  
    Equals, "123-45-6798")  
  var claim =  
    query.select().AtMostOneRow
```

According to best practices, which logic returns notes with the topic of denial and filters on the database?

```
var notesQuery =  
  gw.api.database.Query.make(Note)  
  notesQuery.compare(Note#Topic,  
    Equals, NoteTopicType.TC_DENIAL)
```

Equals, NoteTopicType.TC_DENIAL

```
var denialNotes =  
  notesQuery.select()
```



41.

A developer wrote the following query to create a list of activities from the activity pattern, and then returns the first activity for a failed policy test.

```
1  uses gw-api.database.Query  
2  
3  var activitiesList = Query.sortBy(activitiesList, 'activityId', 'asc').firstWhere('activityId', 'eq', '1')  
4  var firstBindFailed = activitiesList.firstWhere('activityId', 'eq', '1').activityName == 'bind failed'
```

This query uses the sortBy() and firstWhere() methods which are anti-patterns. Where should the developer handle the filtering and sorting to follow best practices?

- In the database
- In the application cache
- On the application server
- In a block statement

Choose 1 option.



This code sample performs poorly due to the use of dot notation with multiple array expansions:

What is the recommended best practice to improve the performance of this code?

- Write a query that fetches the addresses into a collection then uses where() to filter the results
- Replace the dot notation syntax in the code with ArrayLoader syntax
- Rewrite the code with a nested for loop to retrieve the results

Break the code into multiple CQL statements



40.

This code sample performs poorly due to the use of dot notation with multiple array expansions:

```
1  
2  
3
```

```
var lineItems = Claim.Expresses.Transaction.lineItems
```

What is the recommended best practice to improve the performance of this code?

- Write a query that fetches the addresses into a collection then uses `where` to filter the results
- Replace the dot notation syntax in the code with `Array.map` syntax
- Rewrite the code with a nested for loop to retrieve the results
- Break the code into multiple gosu queries to retrieve the results for each user

Choose 1 option.



multiple array expansions:
use of dot notation with
forms poorly

What is the recommended best
practice to improve the performance
of this code?

- Write a query that fetches the addresses into a collection then uses where() to filter the results
- Replace the dot notation syntax in the code with ArrayLoader syntax
- Rewrite the code with a nested for loop to retrieve the results

Break the code into multiple gosu



Given the following code sample:

```
var newBundle = gw.transaction.Transaction.newBundle()  
var targetCo = gw.api.database.Query.make(ABCCompany)  
targetCo.compare(ABCCompany#Name, Equals, 'Acme Brick Co')  
var company = targetCo.select().AtMostOneRow  
company.Notes = "TBD"
```

Following best practices, what two items should be changed to create a bundle and commit this data change to the database? (Select two)

- Add company to the bundle
- End with newBundle.commit()
- Add "(\newBundle ->{" to the first line
- Add Notes to the bundle
- Add targetCo to the bundle

Choose 2 options



1 30°C
Mostly cloudy

Q Search

42.

Given the following code sample:

```
var newBundle = gw.transaction.Transaction.newBundle()  
var targetCo = gw.api.database.Query.make(ABCCompany)  
targetCo.compare(ABCCompany#Name, Equals, "Acme Brick Co")  
var company = targetCo.select().AtMostOneRow  
company.Notes = "TBD"
```

Following best practices, what two items should be changed to create a bundle and commit this data change to the database? (Select two)

- Add company to the bundle
- End with newBundle.commit()
- Add "(newBundle ->{" to the first line
- Add Notes to the bundle
- Add targetCo to the bundle



42

Given the following code sample:

```
var newBundle =  
    gw.transaction.Transaction.newBundle()  
var targetCo =  
    gw.api.database.Query.make(ABCComp  
targetCo.compare(ABCCompany#Name)  
    Equals, "Acme Brick Co.")  
var company =  
    targetCo.select().AtMostOneRow  
company.Notes = "TBD"
```

Following best practices, what two items should be changed to create a bundle and commit this data change to the database? (Select two)



30°C

Mostly cloudy

41.

A developer wrote the following query to create a list of activities sorted by activity pattern, and then returns the first activity for a failed policy bind:

```
FOR @activity IN SELECTED  
    ORDER BY @activity.activityPattern DESC  
    FIRST 1 WHERE @activity.activityPattern = 'Failed Policy Bind'
```

This query uses the `sortBy()` and `firstWhere()` methods which are anti-patterns.

Where should the developer handle the filtering and sorting to follow best practices?

-
- In the database
 - In the application cache
 - On the application server



46.

A developer has added multiple related features in an implementation that needs to be tested. For efficiency, all those features need to be tested at the same time.

Which two statements are true about including multiple tests? (Select two)

They must have the same @Suite annotation

They must set TestResultsDir property

They must use the assertTrue() function

They must be in the same GUnit class



30°C

Mostly cloudy

45.

Which statement is correct and recommended for writing GUnit tests?

- Use fluent assertions over conventional assert statements
- Handle any exceptions thrown by test methods in the finally() method
- Use the init() method to set up objects shared by all tests in a test class
- Clear all instance variables of completed test in the tearDown() method

Choose 1 option.



44.

Which GUnit base class is used for tests that involve Gosu queries in PolicyCenter?

- PCServerTestClassBase
- SuiteDBTestClassBase
- PCUnitTestTestClassBase
- GUnitTestTestClassBase

Choose 1 option.

< Previous page

Submit page >



43.

A query is known to return 500,000 rows.

Which two are recommended to process all 500,000 rows efficiently?
(Select two)

- Use a batch process for large result sets
- Use setPageSize()
- Sort the result by entity name
- Chunk the results into page sets
- Use Google Iterables

Choose 2 options.



A screenshot of a software application window. The main area contains a grid of data with columns labeled 'Page', 'Page'. Some cells in the grid are highlighted in blue or purple. Below the grid, there is a section titled 'Scan Times' with a table showing values like '15m 50s'. At the bottom of the window, there is a search bar with the placeholder text 'Search' and a small magnifying glass icon.

...oper has added multiple related features in an implementation that needs to be tested. For efficiency, all those features need to be tested at the same time.

Which two statements are true about including multiple tests? (Select two)

- They must have the same @Suite annotation
- They must set TestResultsDir property
- They must use the assertTrue() function
- They must be in the same GUnit class

They must be based on the same



Search