PassQueen
a.k.a dumpsqueen.com

Pega Certified Business Architect (PCBA) 87V1

Pegasystems PEGAPCBA87V1

Version Demo

Total Demo Questions: 10

Total Premium Questions: 100

Buy Premium PDF

https://passqueen.com

support@passqueen.com

Topic Break Down

Topic	No. of Questions
Topic 1, Application Design	4
Topic 2, Case Management	38
Topic 3, Data Modeling	16
Topic 4, User Experience	16
Topic 5, Reporting	6
Topic 6, Pega Express	19
Topic 7, Mix Questions	1
Total	100

QUESTION NO: 1

A benefits enrollment case contains a Review stage in which a benefits specialist confirms the employee selections. The specialist must be able to request missing documentation only while the case is in the Review stage.

How do you configure this requirement?

A. Add a stage-based optional action to the Review stage.

B. Add a case-wide optional action to the workflow.

C. Add a parallel process to the Enrollment stage.

D. Add a Wait step to the Review stage.

ANSWER: A

Explanation:

Add a stage-based optional action to the Review stage. A stage-based optional action is available only while the case is in the stage where it is configured. In this scenario, the Request documentation process is displayed to the benefits specialist during Review and is unavailable before enrollment reaches Review or after the case advances to another stage. This provides controlled access to an exception process without placing the action on every assignment. See Pega documentation on optional actions.

QUESTION NO: 2

The business process for an automobile insurance claim consists of the following phases:

• Submission: The customer contacts a customer service representative (CSR) to file the claim.

• Review: An adjuster reviews the claim, assesses the damages to each vehicle, and provides an estimate of the cost of repairs.

• Repair: A third party performs the repairs on each vehicle, communicating with the adjuster and customer as necessary.

• Verification: After each vehicle repair, the adjuster closes the claim.

According to Pega best practices, which phase can you implement as a child case?

A. Repair

B. Verification

C. Submission

D. Review

ANSWER: A

Explanation:

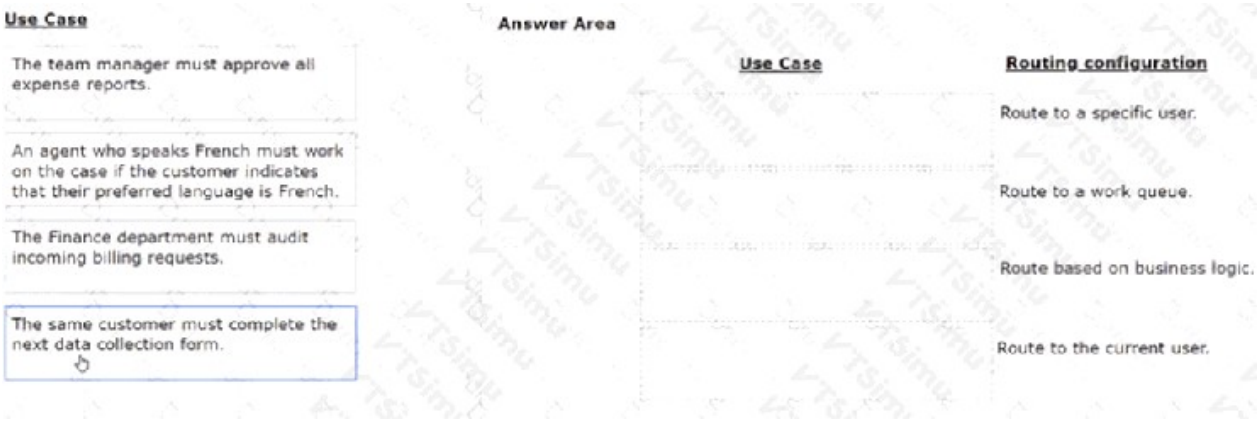
Repair is the appropriate phase to implement as a child case because a single automobile insurance claim can involve repairs for multiple vehicles. Each vehicle repair represents a distinct unit of work with its own repair provider, communications, status, timing, and potentially different completion outcome. Modeling each repair as a child case lets the parent claim create and track one repair case per vehicle while preserving the relationship to the overall claim.

This structure also supports parallel processing: repair work for several vehicles can proceed independently without requiring separate top-level claims. The parent case can coordinate the overall lifecycle, such as waiting for the relevant repair cases to reach completion before progressing to final claim closure activities. Child cases provide their own case data, assignments, SLA processing, history, and participants, while allowing the parent to aggregate status and enforce case-wide business rules. This matches Pega's case-management approach of using child cases for related work that is independently

managed but contributes to completion of a broader business process. See Pega's case type documentation and case hierarchy guidance.

QUESTION NO: 3 - (SIMULATION)

Select each Use Case on the left and drag it to the appropriate Routing configuration on the right.



ANSWER: See the explanation for the answer

Explanation:

Drag the use cases to the routing configurations as follows:

The language requirement requires conditional routing logic, while a department is represented by a work queue. The remaining assignments route directly to the identified manager or back to the current user.

QUESTION NO: 4

Which configuration informs a user by email when an assignment is added to the user's worklist?

A. Configure a service level to send a notification to the assigned user.

B. Configure the case type to send assignment notifications to assigned users.

C. Add a Send Notification step to the process after the assignment to notify the assigned user.

D. Add a Send Email step to the process before the assignment to notify the assigned user.

ANSWER: B

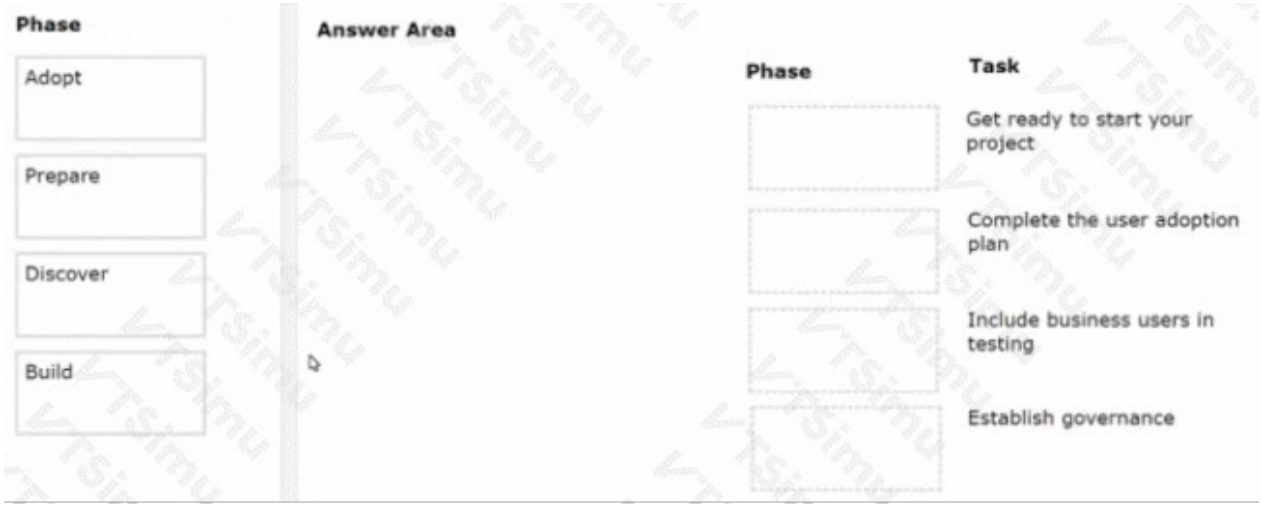
Explanation:

Configure the case type to send assignment notifications to assigned users. Case type settings provide the standard, declarative mechanism for assignment notifications: when an assignment is routed to an operator's worklist, Pega can email that operator to alert them that work is available. This is configured at the case type level in the notification settings, allowing the behavior to be applied consistently to qualifying assignments throughout the case life cycle without adding notification logic individually in each process.

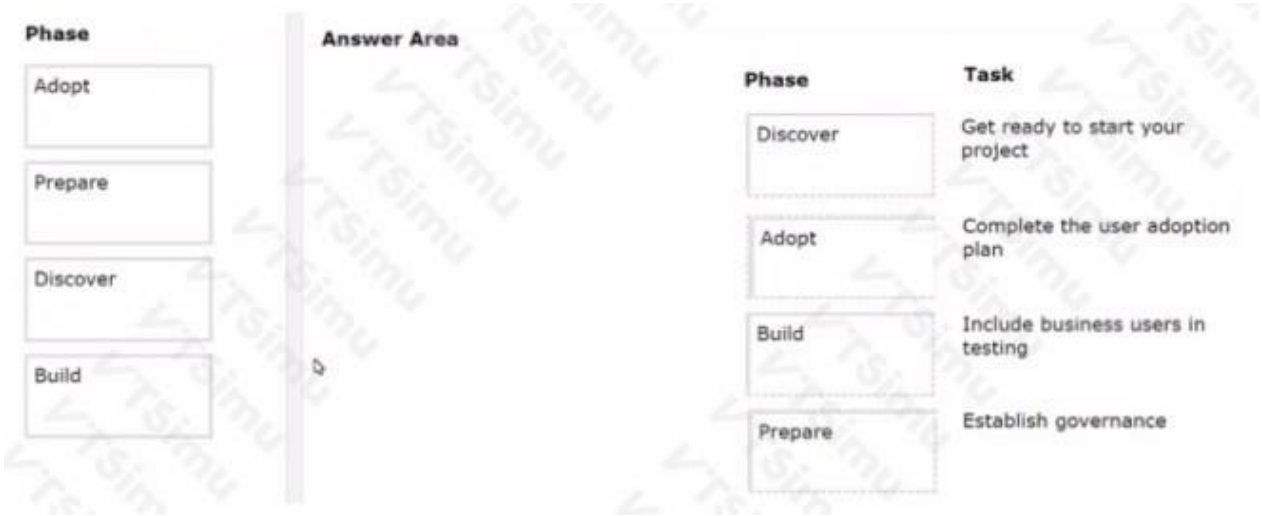
This configuration is specifically intended for worklist routing, where an assignment is owned by an individual user. It supports the common business requirement of promptly notifying users about newly assigned work while keeping the case workflow focused on routing and processing. The notification uses the application's email capability and the recipient information associated with the assigned operator. Administrators can also manage the content and delivery behavior through the application's notification and email configuration. Pega documentation describes case-management configuration and notification capabilities at Pega Case management documentation and provides learning material on assignments and routing at Pega Academy: Routing work.

QUESTION NO: 5 - (DRAG DROP)

Drag each Pega Express'- delivery phase on the left to the matching task on the right.



ANSWER:



Explanation:

Pega Express organizes delivery work into phases that move a project from initial alignment through implementation and user readiness. The correct matching reflects the purpose of each phase in that lifecycle. Discover is the point at which the team gets ready to start the project by aligning on the business problem, desired outcomes, and the initial direction for the work. This early clarity creates the basis for effective delivery planning.

Prepare establishes the delivery foundation, including governance. Governance defines how stakeholders make decisions, how the work is managed, and how the team maintains appropriate oversight throughout the project. With governance in place, the delivery team can work in a controlled and collaborative manner.

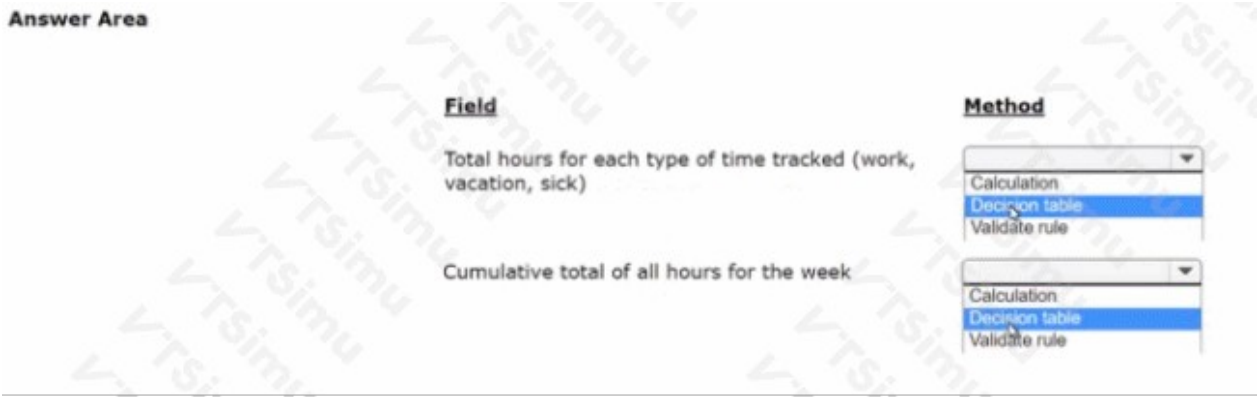
Build is where the application is incrementally developed, validated, and refined. Including business users in testing during this phase ensures that the solution is continuously checked against real business needs and that feedback can be incorporated promptly. This supports Pega's iterative, outcome-focused delivery approach.

Adopt focuses on successful organizational transition to the delivered solution. Completing the user adoption plan ensures that users are prepared to use the application through appropriate communication, training, support, and change-management activities. Together, these mappings represent the expected progression from project readiness, through delivery control and solution validation, to successful user adoption. Pega describes its delivery approach and the importance of iterative collaboration in its Pega Express delivery approach resources. Additional platform and delivery guidance is available through Pega Academy.

QUESTION NO: 6 - (HOTSPOT)

In the following view of a timesheet application, an employee enters their hours for the week. The total hours highlighted are automatically updated with the appropriate values as the employee enters their time.





ANSWER:



Explanation:

Both displayed totals should be configured using a Calculation. In a timesheet, the total for an individual time category is obtained by adding the entered values for that category across the relevant days. For example, the weekly work total is the sum of the employee's daily work-hour entries, while vacation and sick totals are each sums of their own daily entries. This is a direct arithmetic expression over property values, which is precisely the purpose of a calculation.

The cumulative total for the week is also a numeric aggregate. It combines the weekly values across all tracked time categories, or equivalently sums all daily entered hours, to produce one overall hours value. Because it is derived deterministically from current numeric inputs, a calculation keeps the value accurate whenever an employee adds or changes time. The calculation is evaluated from the referenced properties and supplies the updated result without needing a separate business-policy lookup.

Pega calculations are intended for deriving values through expressions, including arithmetic operations and references to case data. This makes them appropriate for rollups such as subtotals and grand totals in a data-entry view. See Pega Academy's overview of calculations and the Pega documentation section on calculating property values. Using a calculation for each field ensures that the totals stay synchronized with the employee's entered hours throughout the week.

QUESTION NO: 7

A customer opens a service request that requires a separate technical investigation. The investigation can have its own assignments, service levels, status, and resolution outcome while remaining associated with the original service request.

Which two statements support modeling the technical investigation as a child case? (Choose Two)

A. A child case has its own case life cycle.

B. A child case can have its own assignments and service levels.

C. A child case must use the same status as its parent case.

D. A child case cannot be associated with a parent case.

ANSWER: A B

Explanation:

A child case has its own life cycle, which allows the technical investigation to progress through its own stages and resolution activities. This is appropriate when the investigation is a distinct unit of work rather than merely a collection of data on the parent service request.

A child case can also have its own assignments and service levels. Technical work can therefore be routed to the appropriate specialists and managed against investigation-specific timing requirements while the parent case tracks the related work. See Pega Academy: Case hierarchy.

QUESTION NO: 8

Before development, your team creates a spreadsheet with work items to populate the backlog. How do you populate the backlog directly from the spreadsheet?

A. Create stones

B. Import stones

C. Create bugs

D. Create feedback

ANSWER: B

Explanation:

Import stones is correct because Pega Agile Workbench supports creating backlog work items in bulk by importing a spreadsheet file. In Agile Workbench, a stone represents a unit of work or requirement that the delivery team can refine, prioritize, and implement. Rather than entering every planned work item individually, the team prepares the required stone details in the supported spreadsheet template and uses the import capability to add those items to the application backlog efficiently. This is particularly useful before development begins, when teams commonly have a large initial set of requirements gathered during discovery and need to establish a workable backlog quickly. After the import, the imported stones are available for backlog management activities such as assigning them to a release, breaking them into smaller work, and tracking their progress. The spreadsheet-import process is specifically intended for transferring a pre-existing list of planned backlog items into Agile Workbench, so it directly fulfills the requirement to populate the backlog from the spreadsheet. Pega documents the process and spreadsheet requirements in its Importing stones from a spreadsheet file guidance. Additional Agile Workbench documentation is available in the Pega Platform Agile Workbench documentation.

QUESTION NO: 9

A customer support case is created when a customer reports a faulty product. The support representative must capture the product serial number and purchase date before troubleshooting begins.

Which two configurations improve the quality of the captured information? (Choose Two)

A. Configure the product serial number as a required field.

B. Configure a validation condition that verifies the purchase date is not in the future.

C. Configure a visibility condition on the product serial number field.

D. Configure the purchase date as a calculated field.

ANSWER: A B

Explanation:

Configure the product serial number as a required field is correct because the serial number is mandatory information for the troubleshooting process. Users cannot submit the view without providing it.

Configure a validation condition that verifies the purchase date is not in the future is also correct because the requirement is a business rule concerning an entered value. Validation prevents invalid future dates from being accepted. Required conditions ensure that mandatory data is supplied, while validation conditions ensure that submitted values satisfy business rules. See Pega Academy: Validating user input.

QUESTION NO: 10

Which three of the following options are application layers in the Situational Layer Cake architecture? (Choose Three)

A. Country

B. Implementation

C. Language

D. Division

E. Organization

ANSWER: B D E

Explanation:

The Situational Layer Cake organizes Pega applications so that reusable rules are separated from rules that are specific to a particular business area or implementation. **Organization** is an application layer used for assets that are common across the organization. **Division** is an application layer for rules shared within a business division while remaining distinct from organization-wide capabilities. **Implementation** is the layer in which an application is tailored for a specific implementation, such as a particular deployment or client solution. This layered structure promotes reuse, supports clearer rule resolution, and reduces duplication by placing shared capabilities at the appropriate level and implementation-specific behavior in the implementation layer. Pega application structure and layering are designed to enable applications to inherit reusable assets while allowing controlled specialization. See Pega Academy's Application layering topic and the Pega documentation on application layering.