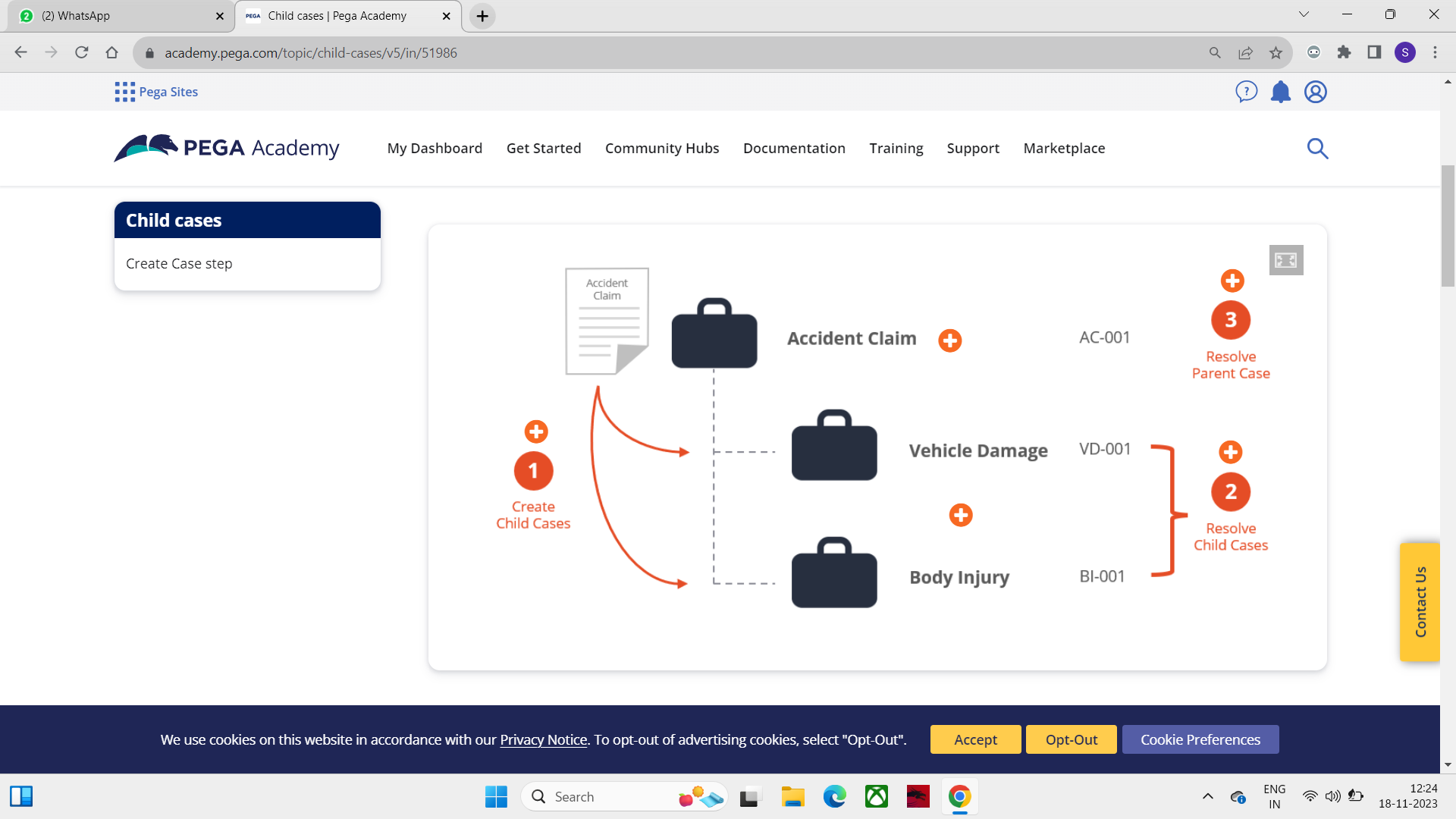
**CHILD CASE:**

* Child case is something which represent work that must be completed to resolve the parent. A parent case that creates multiple child cases allows for work to be processed in parallel.
* Complex business transactions may require the creation of multiple cases that’s why we’ll create child cases.
* Child cases can be created to model work that are performed by different parties or for work that follows different reporting options

for example, an auto insurance accident claim must be able to handle additional claims if the accident resulted in vehicle damage or physical injury. Based on this requirement, the Accident Claim case needs to generate separate Vehicle Damage and Body Injury cases if necessary.



Here Accident Claim is parent Class and Vehicle Damage, Body Injury are child cases of Accident Claim. Accident Claim is Resolved only when the Two child cases are resolved. Parent case waits until the child case is resolved.

**WHEN DO WE USE:**

Child cases are also beneficial in situations for modeling work separately from the parent case when several of the items in the list below are true:

1. A different data model is needed
2. A different life cycle is needed
3. A separate case ID and status for reporting are necessary
4. When separately assigning the case is needed
5. It must complete before the parent completes

**Case Relationships:**

* **Top-level case** a case that does not have a parent case but can become a parent of other cases.
* **Parent case** creates multiple child cases that allow for work to be processed in parallel.
* **Child case** represents work that must be completed to resolve the parent.

**Data Propagation:**

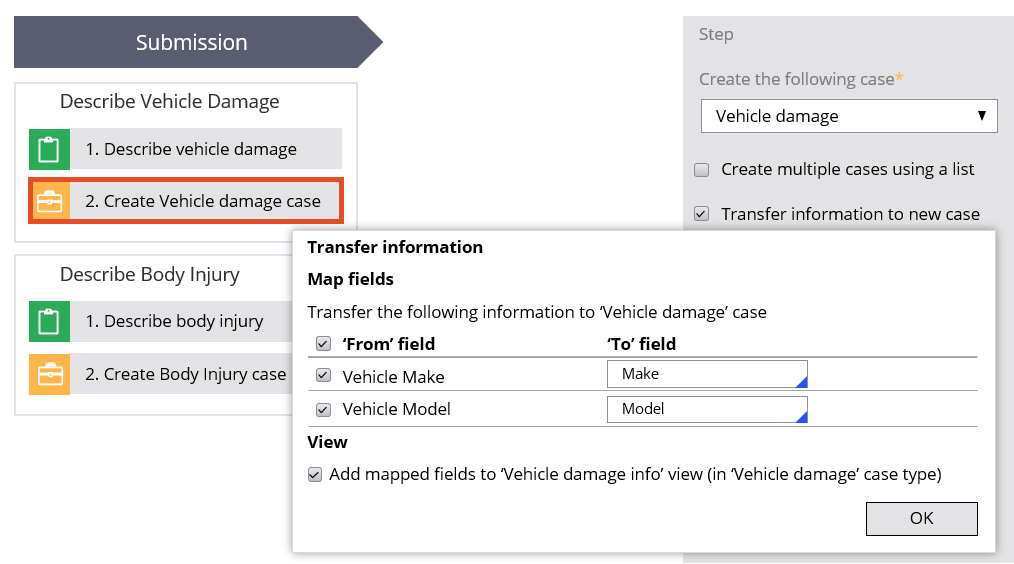
When creating a child case, you can also specify th information to copy from the parent case to the child case through process known as propagation

1. Refer to data page.

2. Copy data from a data page

3. Data transform

4. Directly from Data propagation



**Case Instantiation:**

When a new case is created, it is said to be **instantiated.**

Parent cases are usually instantiated in the Create menu on the Case Manager portal. Subcases are instantiated in the following ways:

* Automatically — By the system when the parent case is created.
* Manually — By the user when the case type is selected from the Other Actions ... menu in the parent work form.

# **Automatic Instantiation:**

To enable automatic instantiation, select the **Automatically by system when** check box and choose one of the following options:

* + **the parent case starts** — The subcase begins when the parent case type is instantiated.  
     Optionally, enter a when rule in the **only if** [rule] **evaluates to true** field.
  + **All/Any of the following dependencies are fulfilled** — The subcase begins when the settings defined in the **Dependencies** item for this case type are met. Choose rules and conditions (**has started**, **has work status**, **has completed**) in the fields below.

# **Manual Instantiation:**

To enable manual instantiation, select **Manually by user**. Optionally, select **only when [rule] evaluates to true** and enter a rule in the field. When it evaluates to true, the user can create a subcase

**Case locking:**

When two or more actions attempt to update a case, the last-performed action may overwrite data written by a previous action. Overwrites can lead to the corruption or loss of data, delaying case processing, and potentially leading to the case's incorrect resolution. If an application supports concurrent users, an appropriate case locking strategy is essential to ensuring data integrity.

Pega Platform™ supports two strategies to balance user access with the need for data protection: Pessimisticlocking and optimisticlocking.

## 

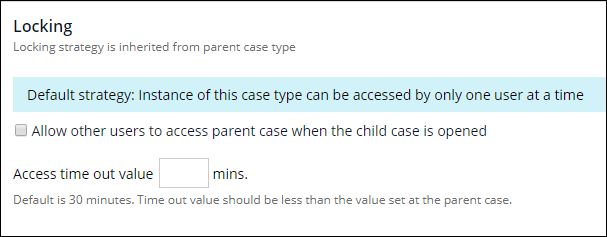
**Pessimistic locking (Allow one user):**

With a **pessimistic locking strategy**, an application applies an exclusive lock when opening an item, such as a case. A user or the system that opens the object gainsexclusive access to the object until the application releases the lock.

Select **Allow one user** to apply a pessimistic locking strategy

By default, Pega Platform selects the **Allow one user** option when creating a case type, which ensures data integrity during case actions.

If we select **Allow one user**, you can modify the default locking time out of **30** mins



User 1 opens the item and begins editing. The application prevents all other users from opening the item for editing. User 1 saves the items and commits their changes. The application removes the lock on the item so that other users can now open the item for editing

* **optimistic locking** (**Allow multiple users):**

With an **optimistic locking strategy**, an application does not apply an exclusive lock when opening an item. Instead, any user — or the system itself — can open and edit the case at any time. When the user attempts to save, the application checks whether the case has changed before committing any changes.

Select **Allow multiple users** to apply an optimistic locking strategy.

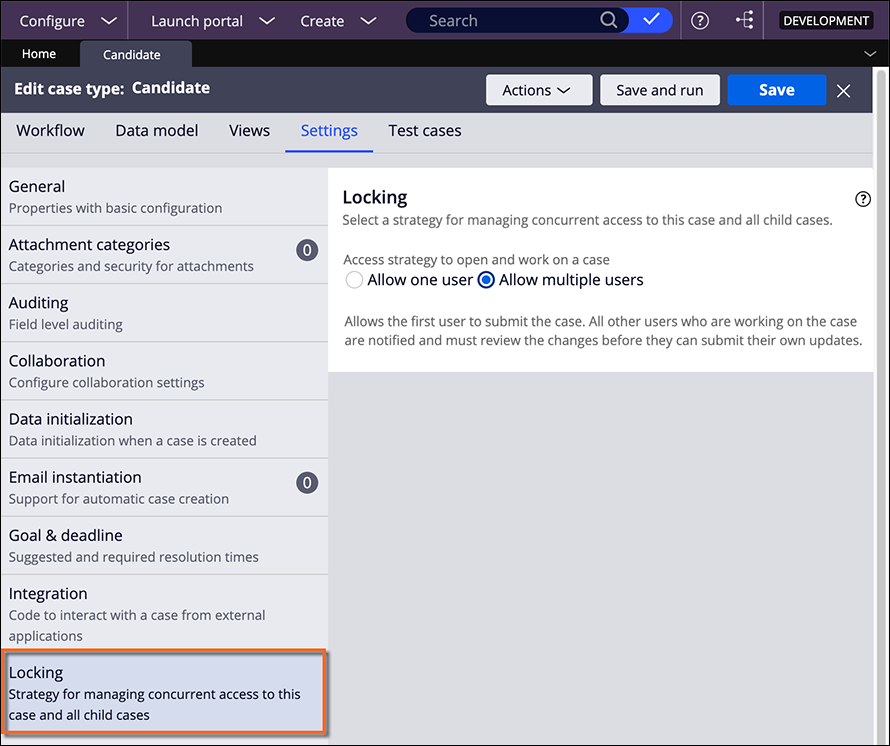
User 1 opens the item and begins editing.

- User 2 opens the item. Because the application does not lock the item for User1, User 2 can edit the item.

- User 2 saves the item and commits their changes.

- User 1 attempts to save the item. The application informs User 1 that the item is stale because User 2 already committed their changes

.



The child case cannot be a temporary case.

– The parent case refers to the child case by its case ID.

Case ID for a child case and a parent case is typically composed of a combination of the Case Type and a unique identifier

* Parent Case ID: ABC-123
* Child Case ID: ABC-123-001

Here, "ABC" represents the Case Type, "123" is the unique identifier for the parent case, and "001" is the unique identifier for the child case. The hyphen (-) is commonly used as a separator.

* While we run the parent case, the parent data is available on pyWorkPage.
* When the child case runs the parent case data is available now on pyWorkCover, the child case data is available on the pyWorkPage.
* After the child case is completed the parent case data is moved to the pyWorkPage.
* The Case Dependency wait type pauses case processing of a parent case until the child case dependency resolves. You create a case dependency by adding a Wait step to the parent case. When a parent case reaches the Wait step, the case pauses until all child cases, or any child case of a given type reach a defined status.

**INTERVIEW QUESTIONS**

1.What is a child case in Pega?

A child case in Pega is associated with another case, known as the parent case. It represents a subprocess or a specific task within the context of the main case.

2. How do you create a child case in Pega?

A child case is typically created using the "Create Case" smart shape in a flow. You configure the smart shape to specify the child case type and provide necessary data.

3.Can a child case have a different case type than its parent case?

Yes, a child case can have a different case type than its parent case. This allows for flexibility in modeling different subprocesses within a parent case.

4.How does data flow between a parent case and its child case?

Data can flow between a parent case and its child case through various means, such as setting properties in the child case based on values from the parent case or using data transforms to pass information.

5. How do you ensure synchronization between a parent and child case?

Synchronization between a parent and child case can be achieved by leveraging the "Wait for Process" and "Resume" shapes in the flow. These shapes help control the flow of execution between the parent and child cases.

6.What is the significance of the "pyWorkCover" property in the context of child cases?

The "pyWorkCover" property is used to link a child case to its parent case. It stores the handle to the parent case, establishing the relationship between the two cases.

7. Can a child case have its own child cases

Yes, a child case can have its own child cases, creating a hierarchical structure of cases. This allows for the modeling of complex business processes.

8. How do you handle errors or exceptions in child case processing?

Error handling in child cases can be managed with exception flows and error handling activities. This ensures that any issues encountered during the processing of a child case are appropriately addressed.

9. What is the purpose of creating Child Cases?

Creating child cases helps model complex business processes by breaking them into manageable and modular components. It allows for better organization, reuse, and maintenance of case logic.

10. How is the association between Parent and Child Cases maintained?

The association between parent and child cases is maintained through the "pzInsKey" property, which holds the unique identifier of the case. This key is used to link child cases to their respective parent cases.

11.Can a Child Case have its own subprocesses or child cases?

Yes, a child case in Pega can have its own subprocesses or child cases, creating a hierarchical structure for modeling complex business scenarios.

12. How can you pass data from a Parent Case to a Child Case?

Data can be passed from a parent case to a child case by mapping properties in the "Create Case" shape or using data transforms to set values before creating the child case.

13. What is the significance of the "Cascade" option when creating Child Cases?

The "Cascade" option determines whether changes made to a parent case propagate down to its child cases. If enabled, modifications in the parent case are reflected in the child cases.

14. How can you dynamically create Child Cases based on conditions?

Dynamic case creation can be achieved using decision shapes or custom logic in flows to determine when and which child cases should be created based on runtime conditions.

Q: What is a child case in Pega?

A: A child case in Pega is a case type that is created and associated with a parent case. It represents a specific task or activity that is part of the overall process managed by the parent case.

Q: How are child cases created in Pega?

A: Child cases can be created in Pega using automation steps in the parent case's life cycle. These automation steps define the conditions and actions required to create and associate a child case with the parent case.

Q: What is the purpose of using child cases in Pega?

A: Child cases are used in Pega to break down complex processes into smaller, manageable units. They allow for better organization, tracking, and collaboration of tasks within the parent case. Child cases also enable reusability and modularity in building applications.

Q: Can a child case have its own child cases?

A: Yes, a child case can have its own child cases, creating a hierarchical structure of cases. This allows for further decomposition of tasks and activities within the overall process.

Q: How are child cases related to the parent case in Pega?

A: Child cases are associated with the parent case through a relationship defined in the data model. This relationship allows for the parent case to track and manage the progress and status of its associated child cases

Q. How are child cases related to the parent case in Pega?

A: Child cases are associated with the parent case through a relationship defined in the data model. This relationship allows for the parent case to track and manage the progress and status of its associated child cases.

Q: How can child cases be accessed and managed in Pega?

A: Child cases can be accessed and managed within the context of the parent case. They can be viewed, updated, and resolved individually or collectively, depending on the requirements of the business process.

. Q: How can child cases be beneficial in complex business transactions?

A: Child cases allow for parallel processing, where different parties with different expertise can handle each child case, leading to faster resolution and improved efficiency.

. Q: How can child cases be reused in different scenarios?

A: By creating modular child case types, you can reuse them in various business processes, saving time and effort in designing and implementing new case types.

. Q: How can you model work that follows different reporting options using child cases?

A: Child cases provide flexibility in modeling work that requires different reporting options, allowing you to define specific reporting requirements for each child case.

. Q: How can you establish dependencies between parent and child cases?

A: By defining case dependencies, you can ensure that child cases are resolved before the parent case reaches resolution, enabling a structured and efficient case management process.

1. Q: What is child case locking in Pega?

A: Child case locking in Pega refers to the mechanism that ensures exclusive access to a child case by a user or a group of users, preventing simultaneous updates and maintaining data integrity.

2. Q: How does child case locking work in Pega?

A: When a user opens a child case in Pega, the system locks the case to prevent other users from making conflicting updates. The lock is released when the user completes their work on the child case.

3. Q: What are the different types of child case locking strategies in Pega?

A: Pega offers two types of child case locking strategies: pessimistic locking and optimistic locking. Pessimistic locking allows only one user to access the child case at a time, while optimistic locking allows multiple users to access the child case simultaneously.

4. Q: How can child case locking be configured in Pega?

A: Child case locking can be configured in Pega by defining the locking strategy in the case type settings. The locking strategy determines whether pessimistic or optimistic locking is used for child cases.

5. Q: What are the considerations for choosing the appropriate child case locking strategy in Pega?

A: The choice of child case locking strategy in Pega depends on factors such as the nature of the work, the number of users involved, and the desired level of concurrency. Pessimistic locking ensures data integrity but may limit concurrency, while optimistic locking allows for greater concurrency but requires conflict resolution mechanisms.

**Scenario Questions:-**

1. Q: In a car insurance claim process, how can child cases be used to handle different types of claims?

A: Child cases can be created for specific claim types such as vehicle damage, bodily injury, or property damage. Each child case can be assigned to the appropriate party or department responsible for handling that specific claim type.

2. Q: How can child cases be used to model work performed by different parties in a customer service process?

A: Child cases can represent different tasks or responsibilities within a customer service process. For example, a child case can be created for a technical support team to handle troubleshooting while another child case can be created for billing inquiries handled by a finance team.

3. Q: In a loan application process, how can child cases be used to handle different stages of the application review?

A: Child cases can be created for each stage of the loan application review process, such as document verification, credit check, and approval. Each child case can be assigned to the appropriate team responsible for that stage, allowing parallel processing and efficient case resolution.

4. Q: How can child cases be used to handle multiple related tasks within a larger business process?

A: Child cases can represent individual tasks or sub-processes within a larger business process. For example, in a customer onboarding process, child cases can be created for identity verification, account setup, and product selection, each handled by different teams or departments.

5. Q: How can child cases be used to handle different reporting options in a complaint management process?

A: Child cases can be created for different types of complaints, such as product-related complaints, service-related complaints, or billing-related complaints. Each child case can be assigned to the appropriate team responsible for handling that specific type of complaint.

6. Q: In a healthcare provider enrollment process, how can child cases be used to handle credentialing and contracting separately?

A: Child cases can be created for the credentialing process and the contracting process, allowing different teams to work on each aspect independently. This ensures that the necessary checks and agreements are completed before the provider enrollment process is finalized.

7. Q: How can child cases be used to handle different time frames in a project management process?

A: Child cases can represent different phases or milestones within a project. Each child case can have its own timeline and assigned resources, allowing parallel processing and efficient project execution.

8. Q: In a customer complaint resolution process, how can child cases be used to handle escalations to different levels of management?

A: Child cases can be created for each level of escalation, such as first-level manager, second-level manager, and executive management. Each child case can be assigned to the appropriate level of management for review and resolution.

9. Q: How can child cases be instantiated in a loan application process?

A: In a loan application process, child cases can be instantiated for various stages, such as document verification, credit check, and approval. Each child case represents a specific task or evaluation that needs to be completed before the loan application can progress.

10. Q: In a customer onboarding process, how can child cases be instantiated for different account types?

A: Child cases can be instantiated for different account types, such as savings account, checking account, and credit card account. Each child case represents the setup and configuration process specific to that account type.

11. Q: How can child cases be instantiated for different product configurations in a manufacturing process?

A: Child cases can be instantiated for different product configurations, such as size, color, and features. Each child case represents the customization and assembly process specific to that product configuration.

12. Q: In a complaint management process, how can child cases be instantiated for different complaint categories?

A: Child cases can be instantiated for different complaint categories, such as product quality, delivery issues, and customer service. Each child case represents the investigation and resolution process specific to that complaint category.

13: In a manufacturing process, how can child cases be instantiated for different production stages?

A: Child cases can be instantiated for different production stages, such as raw material procurement, assembly, quality control, or packaging. Each child case represents a specific stage that needs to be completed to manufacture a product.

14. Q: How can child cases be used in a compliance process?

A: Child cases can be instantiated for different compliance activities, such as risk assessment, audit preparation, or regulatory reporting. Each child case represents a specific compliance task that needs to be addressed to ensure adherence to regulations.

15. Q: In a software development process, how can child cases be instantiated for different development tasks?

A: Child cases can be instantiated for different development tasks, such as requirements gathering, coding, testing, or deployment. Each child case represents a specific task that needs to be completed to develop a software application.

16.How does the parent case interact with child cases in Pega?

A: The parent case in Pega can wait for the completion of child cases before it can be resolved. The data between the parent and child cases can be shared and synchronized as needed.