

<b>Class Name:</b> Field	
<b>Description:</b> Element of ProtoElement that is editable Rule(s) in a given Protocol	
<p><b>Contracts:</b></p> <p>5. Visualize Field  <i>Show Field</i>  showField(Field)  Description: Receives a field currently hidden and displays it.  Pre: The field belongs to one of A and is not hidden.  Post: The field is displayed.</p> <p><i>Hide Field</i>  hideField(Field)  Description: Receives a field currently displayed and hides it.  Pre: The field belongs to one of A and is not hidden.  Post: The field is hidden.</p> <p>6. Modify Field  <i>Rename Field</i>  renameField(String, Action, Field)  Description: Receives a string and a field to rename using the provided string, adding to the Action  Pre: The Field exists within the PDML  Post:</p> <p><i>Add Hook to Field</i>  addHook(Action, Field)  Description: Adds a Hook to this Field in this Action.  Pre: The Field is editable by this Formatter.  Post: The Hook is added to the Action for this Field with a Script attached to it.</p> <p><i>Annotate Field</i>  addAnnotation(Action, Annotation)  Description: Receives an action and an annotation, adds a new annotation to the action.  Pre: The action is either hiding, renaming, annotating or creating a hook.  Post: The addition of the specified annotation to the action.</p> <p><i>Update Field</i>  updateField(String, Field)  updateField(Boolean, Field)  Description: Receives either a string or a Boolean along with a field, and updates a field value according to the value provided.  Pre: The Field exists within the PDML  Post:</p> <p><b>Private Responsibilities:</b></p>	<p><b>Collaborations:</b></p> <p>Action (2)</p> <p>Action (2)</p> <p>Action (1)</p> <p>Action (1)</p> <p>AnnotationAction (3)</p> <p>Hook (15)</p>
<p><b>Comments:</b>  Contains the following optional and required attributes:  Optional attributes – hidden (Boolean), mask (Bytes), showdtl (String), showmap (String), showvalue (Integer)  Required attributes – longname (String), name (String), pos (Bytes), show, showname (String), size (Bytes), unmaskedvalue (<i>required only if the length of the current field is not an integer number of bytes</i>), value (Hex, String)</p>	

<b>Class Name:</b> Formatter	
<b>Description:</b> Allows to apply rules to Fields in a Protocol for inspection	
<b>Contracts:</b> 10. Manage Formatter <i>Create Formatter</i> createFormatter(Rule<>) Description: Receives a list of rules to build a new formatter. Pre: The list of rules is empty Post:  <i>Apply Rules</i> applyRules(Formatter, Rule<>) Description: Receives a Formatter and a set of Rules and applies the Rules to the Formatter. Pre: Rule<> is not empty, and the Formatter exists Post: 11. Manage Rules <i>Add a New Rule</i> addRule(Formatter, Rule) Description: Receives a formatter and a rule, adds the rule to the formatter. Pre: The Rule is not empty and the Formatter exists. Post:  <i>Delete the Last Rule</i> deleteRule(Rule) Description: Receives a rule and deletes it. Pre: The Rule is the last one added to the Formatter Post:	<b>Collaborations:</b>  PDML (17)   Rule (22, 23)   Rule (22, 23)
<b>Comments:</b> Contains Filter and Action rules specific to one Protocol; type of File	

<b>Class Name:</b> Script	
<b>Description:</b> Subtype of File Class which contains the Python algorithm by which to update Field(s)	
<b>Contracts:</b> 24. Manage Script <i>Create a New Script</i> createScript(Hook) Description: Creates a Script for this Hook Pre: Post: The Script Python File will be created if it does not already exist.  <i>Delete Script</i> deleteScript(Hook) Description: Deletes the Script from this Hook Pre: The Script is associated with this Hook. Post: The Script File is not deleted. 25. Modify Script <i>Update Algorithm</i> updateAlgorithm(String<>) Description: Overwrites the existing algorithm to contain this String<> Pre: The Script Python File exists. Post:  <i>Update File Path</i> updatePath(String) Description: Sets the path to this Script to be this String Pre: This String is a valid file system path to this Algorithm Post: The Script is not deleted. 26. Run Script <i>Get Return Value</i> getReturnValue() Description: Gets the return value from the Script's execution Pre: The Script has completed execution and has a return value Post:  <i>Update Field</i> updateField(Field) Description: Updates the Field associated with this Hook Pre: The Script's return value type matches the Field's value type Post:  <b>Private Responsibilities:</b>	<b>Collaborations:</b>  Hook (13, 14)   Hook (14)   Hook (14)   Hook (14)   Hook (15)   Hook (15)
<b>Comments:</b> Contains editable algorithm and file path; type of File	