

SDFFormatParserI
<ul style="list-style-type: none"> <li>+ virtual void Initialize (const std::string file_path, bool &amp;success)=0</li> <li>+ virtual sdf::SDFPtr GetSDFElement()=0</li> <li>+ virtual std::vector&lt; sdf::ElementPtr &gt; LookupElementsByAttributeType (const std::string &amp;attribute_type)=0</li> <li>+ virtual std::vector&lt; sdf::ElementPtr &gt; LookupElementsByAttributeType AndValue(const std::string &amp;attribute_type, const std::string &amp;attribute_value)=0</li> <li>+ virtual std::vector&lt; sdf::ElementPtr &gt; LookupElementsByType (const std::string &amp;type)=0</li> <li>+ virtual Mentions FindMentions (std::string key)=0</li> <li>+ virtual Mentions FindMentions (std::string key, sdf::ElementPtr element_to_exclude)=0</li> <li>+ virtual Mentions FindMentions (std::string key, sdf::ParamPtr attribute_to_exclude)=0</li> <li>+ virtual Mentions FindMentions (std::string key, sdf::ElementPtr element_to_exclude, sdf::ParamPtr attribute_to_exclude)=0</li> <li>+ virtual std::string GetSDFTreePathToElement (sdf::ElementPtr element)=0</li> <li>+ virtual std::pair&lt; std::vector&lt; ModelViewerI::ModelInfo &gt;, std::vector&lt; ModelViewerI::PresetModelInfo &gt; &gt; GetModelsFromSDFTree(bool render_collisions=false)=0</li> </ul>



SDFFormatParser
<ul style="list-style-type: none"> <li>- sdf::SDFPtr sdfElement</li> <li>- std::string sdf_file_path</li> </ul>
<ul style="list-style-type: none"> <li>- void Initialize(const std::string file_path, bool &amp;success) override</li> <li>- sdf::SDFPtr GetSDFElement () override</li> <li>- std::vector&lt; sdf::ElementPtr &gt; LookupElementsByAttributeType(const std::string &amp;attribute_type) override</li> <li>- std::vector&lt; sdf::ElementPtr &gt; LookupElementsByAttributeTypeAndValue(const std::string &amp;attribute_type, const std::string &amp;attribute_value) override</li> <li>- std::vector&lt; sdf::ElementPtr &gt; LookupElementsInternal (const std::string &amp;attribute_type, const std::string &amp;attribute_value, sdf::ElementPtr scope, sdf::ElementPtr scope_to_exclude)</li> <li>- sdf::ElementPtr FindScope (sdf::ElementPtr element)</li> <li>- sdf::ElementPtr FindCanonical (sdf::ElementPtr element)</li> <li>- std::vector&lt; sdf::ElementPtr &gt; LookupElementsByType (const std::string &amp;type) override</li> <li>- Mentions FindMentions (std::string key) override</li> <li>- Mentions FindMentions (std::string key, sdf::ElementPtr element_to_exclude) override</li> <li>and 8 more...</li> </ul>