

doctest::toString



```
graph LR; A[doctest::toString] --> B[doctest::detail::StringMaker  
Base< detail::has_insertion  
_operator< T >::value||detail  
::types::is_pointer< T >::value|  
|detail::types::is_array< T >::  
value >::convert];
```

The diagram illustrates a function call. A grey rectangular box on the left contains the text 'doctest::toString'. A blue arrow points from this box to a larger white rectangular box on the right. This white box contains the implementation details of the function, showing the 'doctest::detail::StringMaker' struct and its 'Base' template specialization. The implementation uses a series of conditional compilation directives (if, else if, else) to handle different types: 'has\_insertion\_operator' for general types, 'is\_pointer' for pointer types, and 'is\_array' for array types. Each branch uses the 'value' member of the 'Base' struct and the 'convert' static member function to produce the final string representation.

```
doctest::detail::StringMaker  
Base< detail::has_insertion  
_operator< T >::value||detail  
::types::is_pointer< T >::value|  
|detail::types::is_array< T >::  
value >::convert
```