

class FB JSAPIAuto GetProperty (2)

- Construct
- FireJSEvent
- get_valid
- getAttribute
- getMemberCount
- getMemberNames
- GetMethodObject
- GetProperty**
- GetProperty
- HasMethod
- HasMethodObject
- HasProperty
- HasProperty
- Invoke
- isReserved
- JSAPIAuto
- registerAttribute
- registerMethod
- registerProperty
- RemoveProperty
- RemoveProperty
- setAttribute
- SetProperty
- SetProperty
- setReserved
- ToString
- unregisterMethod
- unregisterProperty

FB::variant FB::JSAPIAuto::GetProperty (int idx) virtual

Gets the value of an indexed property.

This can be used for providing array-style access on your object. For example, the following will result in a call to GetProperty with idx = 12:

```
var i = document.getElementById("plugin")[12];
```

Parameters

idx Zero-based index of the property to get the value of.

Returns

The property value.

Implements **FB::JSAPI**.

Definition at line 276 of file **JSAPIAuto.cpp**.

```
277 {
278     boost::recursive_mutex::scoped_lock lock(m_zoneMutex);
279     if(!m_valid)
280         throw object_invalidated();
281
282     std::string id = boost::lexical_cast<std::string>(idx);
283     AttributeMap::iterator fnd = m_attributes.find(id);
284     if (fnd != m_attributes.end() && memberAccessible(m_zoneMap.find(id)))
285         return fnd->second.value;
286     else if (m_allowDynamicAttributes) {
287         return FB::FBVoid(); // If we allow dynamic attributes then we need to
288         // return void if the property doesn't exist;
289         // otherwise checking a property will throw an exception
290     } else {
291         throw invalid_member(boost::lexical_cast<std::string>(idx));
292     }
293
294     // This method should be overridden to access properties in an array style from
295     // javascript,
296     // i.e. var value = pluginObj[45]; would call GetProperty(45)
297     // Default is to throw "invalid member" unless m_attributes has something matching
298 }
```