

If the declaration of 'M' above had excluded the exported variable 'a', 'M' would be a non-instantiated module and it would be an error to reference 'M' as a *PrimaryExpression*.

An internal module declaration that specifies an *IdentifierPath* with more than one identifier is equivalent to a series of nested single-identifier internal module declarations where all but the outermost are automatically exported. For example:

```
module A.B.C {  
    export var x = 1;  
}
```

corresponds to

```
module A {  
    export module B {  
        export module C {  
            export var x = 1;  
        }  
    }  
}
```

## 10.2 Module Body

The body of an internal module corresponds to a function that is executed once to initialize the module instance.

*ModuleBody:*

*ModuleElements*<sub>opt</sub>

*ModuleElements:*

*ModuleElement*

*ModuleElements* *ModuleElement*

*ModuleElement:*

*Statement*

*export*<sub>opt</sub> *VariableDeclaration*

*export*<sub>opt</sub> *FunctionDeclaration*

*export*<sub>opt</sub> *ClassDeclaration*

*export*<sub>opt</sub> *InterfaceDeclaration*

*export*<sub>opt</sub> *EnumDeclaration*

*export*<sub>opt</sub> *ModuleDeclaration*

*export*<sub>opt</sub> *ImportDeclaration*

Each module body has a declaration space for local variables (including functions, modules, class constructor functions, and enum objects), a declaration space for local named types (classes, interfaces,