

```

Object.defineProperty(<ClassName>, "<MemberName>", {
  get: function () {
    <GetAccessorStatements>
  },
  set: function (<ParameterName>) {
    <SetAccessorStatements>
  },
  enumerable: true,
  configurable: true
});

```

where *MemberName* is the name of the member accessor, *GetAccessorStatements* is the code generated for the statements in the get accessor's function body, *ParameterName* is the name of the set accessor parameter, and *SetAccessorStatements* is the code generated for the statements in the set accessor's function body. The 'get' property is included only if a get accessor is declared and the 'set' property is included only if a set accessor is declared.

StaticVariableAssignments is a sequence of statements, one for each static member variable declaration with an initializer, in the order they are declared, of the form

```
<ClassName>.<MemberName> = <InitializerExpression>;
```

where *MemberName* is the name of the static variable, and *InitializerExpression* is the code generated for the initializer expression.

8.6.2 Classes With Extends Clauses

A class with an extends clause generates JavaScript equivalent to the following:

```

var <ClassName> = (function (_super) {
  __extends(<ClassName>, _super);
  function <ClassName>(<ConstructorParameters>) {
    <DefaultValueAssignments>
    <SuperCallStatement>
    <ParameterPropertyAssignments>
    <MemberVariableAssignments>
    <ConstructorStatements>
  }
  <MemberFunctionStatements>
  <StaticVariableAssignments>
  return <ClassName>;
})(<BaseClassName>);

```

In addition, the '`__extends`' function below is emitted at the beginning of the JavaScript source file. It copies all properties from the base constructor function object to the derived constructor function object