

```
1  /****
2  *
3  * BlockEntry extends SymbolTableEntry by adding a data field to support
4  * nested scoping. The field is a reference to a Symbol table for the block's
5  * scope.
6  *
7  * BlockEntry is intended to support anonymous inner blocks, in contrast to <p>
8  * href= FunctionEntry.html> FunctionEntry </a>, which supports the named
9  * scoping blocks defined by a function scope.
10 *
11 */
12 public class BlockEntry extends SymbolTableEntry {
13
14     /**
15     * Construct this with null data fields.
16     */
17     public BlockEntry() {
18     }
19
20     /**
21     * Construct this with the given data field value.
22     */
23     public BlockEntry(SymbolTable scope) {
24         super("Anonymous block", null);
25         this.scope = scope;
26     }
27
28     /**
29     * Return the string rep of this.
30     */
31     public String toString(int level) {
32         return super.toString(level) + scopeString(level);
33     }
34
35     /**
36     * Called by toString to recursively stringify the scope, if non-null.
37     */
38     protected String scopeString(int level) {
39         return scope == null ? "" : "\n " + indentString(level) +
40             scope.toString(level);
41     }
42
43     /** Local scope for this function. */
44     public SymbolTable scope;
45
46 }
```