```
 1  /****
 2   *
 3   * LValue is a specialized Value extension to represent the address-valued
 4   * result of evaluating a memory designator.  Any variable or designator
 5   * expression used in an l-value context is represented as an LValue.
 6   *
 7   */
 8  public class LValue extends Value {
 9
10      public LValue(Object val, TypeNode type) {
11          super(val, type);
12      }
13
14      /**
15       * Return the value of this as an integer address.  This method signifies
16       * the key distinction between Value as an r-value, an the LValue extension
17       * as an l-value.  Namely, the val component of an r-Value is an Object,
18       * meaning some form of data value.  As such, it can be accessed directly
19       * via the public .val field, which is of type Object.  In contrast, the
20       * val component of an l-value is an integer memory address, which is
21       * delivered with this getVal method.
22       */
23      public int getVal() {
24          return ((Integer) val).intValue();
25      }
26
27  }
```