l-value:
---
*.In computer science, a value that points to a storage location, potentially allowing new values to be assigned.

r-value:
---
*.In computer science, a value considered independently of its storage location.

***.variable:**
---
*.the variables are called symbolic variables because these are named loctions. the declaration of a variable in c informs about:

*.there are two values associated with a *symbolic variables*:
1.its **data value**, stored at some location in memory. it is sometimes referred to as a variables rvalue .
2.its **location value**; i.e., the address in memory where the data is stored. it is sometimes referred to as variable's lvalue.

Methaphore:

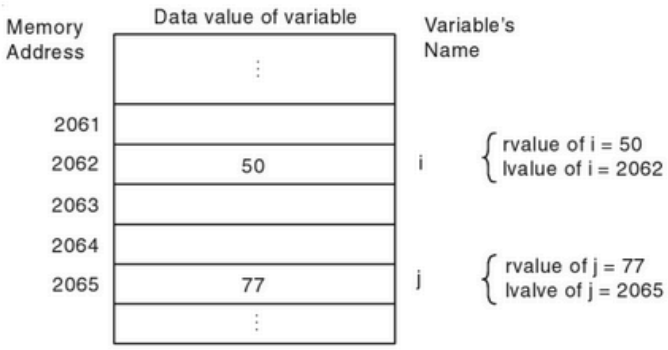
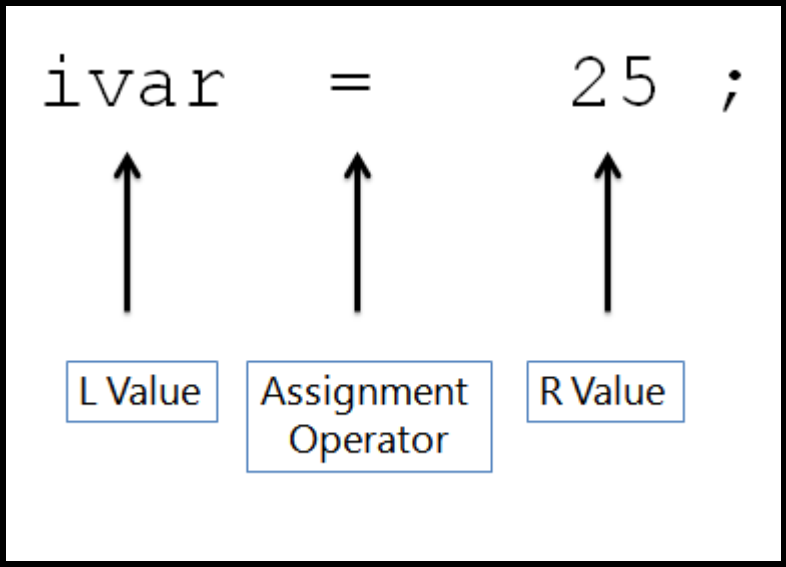


Fig. 5. The rvalue and lvalue of a variable

An *lvalue* is an expression to which you can assign a value. The left of an = (assignment operator) must be an lvalue.

> ***Note :*** *Remember that constant values and constant identifiers (declared using const keyword) are not lvalues and can only appear to the right side of an assignment operator.*

Simple convention:



Compile time errors only:

```
*.int arr[10];
   error: lvalue required as increment operand
  arr++;
      ^
*.5 = 9;
  error: lvalue required as left operand of assignment
  5 = 9;
      ^
*.int a = 9;
  5 = i;
   error: lvalue required as left operand of assignment
  5 = val;
      ^
*.instead of
  if(0 == x)    //works correctly

  if(0 = x)    // lvalue required as left operand of assignment


*.(val+1) = 9;
  error: lvalue required as left operand of assignment
  (val+1) = 9;
            ^
```

```
*.&n = 1000;
  error: lvalue required as left operand of assignment
  &val = 1000;
      ^
```