else if (num > shares)
{
    cerr << "You can't sell more than you have! "
        << "Transaction is aborted.\n";
}
else
{
    shares -= num;
    share_val = price;
    set_tot();
}
}

void Stock::update(double price)
{
    share_val = price;
    set_tot();
}

void Stock::show()
{
    cout << "Company: " << company
         << " Shares: " << shares << '\n'
         << " Share Price: $" << share_val
         << " Total Worth: $" << total_val << '\n';
}

**Member Function Notes**

The `acquire()` function manages the first acquisition of stock for a given company, whereas `buy()` and `sell()` manage adding to or subtracting from an existing holding. The last two methods make sure that the number of shares bought or sold is not a negative number. Also, if the user attempts to sell more shares than he or she has, the `sell()` function terminates the transaction. The technique of making the data private and limiting access to public functions gives us control over how the data can be used; in this case, it allows us to insert these safeguards against faulty transactions.