Adding built-in “Pre-Imposed Conditions” in Java. Pre-conditions, post-conditions and invariants are introduced by the Keywords require, ensure & invariant respectively.

Your team is well versed in Java. They “love” the language and will go down passionately fighting for it! Lately though, they have been working on mission critical projects and have been sloppy asserting their code. They blame it on the lack of built-in “Pre-imposed condition” functionality in Java, and frown on your mention of another language. There is only one option (besides firing them!). Extend the language!

class Stack
  invariant (count >= 0) && (count <= maxsize);
  {
    int data[], count, maxsize;
    public Stack (int size) {
      maxsize = size;
      data = new int[size];
      count = 0;
    }
    ...
    public void pop ()
      require !empty ()
      ensure !full () && (count == old count - 1);
      count = count - 1;
    }
    public int top ()
      require !empty ();
      return data[count-1];
    }
    public boolean empty () {
      return count == 0;
    }
    public boolean full () {
      return count == maxsize;
    }
  }

    public void pop () {
      int _old_data[], _old_count, _old_maxsize;
      _old_data = data;
      _old_count = count;
      _old_maxsize = maxsize;
      if (!(count >= 0) && (count <= maxsize)) {
        System.exit (1);
      }
      if (!(! empty ())) {
        System.exit (1);
      }
      if (!(count >= 0) && (count <= maxsize)) {
        System.exit (1);
      }
      if (!(! full () && (count == _old_count - 1))) {
        System.exit (1);
      }
      count = count - 1;
    }

//macro
Invariant: 'invariant' Expression ';

....
SYMBOL Invariant COMPUTE SYNT.IdemPtg = PTGNULL;
END;
RULE: ClassBody ::= Invariant '{' ClassBodyDeclarations '}'
COMPUTE
  ClassBody.Invariant = Invariant.Cond;
END;
SYMBOL ClassBody COMPUTE
  SYNT.Invariant = PTGNULL;