

```

;; The first four lines of this file were added by Dracula.
;; They tell DrScheme that this is a Dracula Modular ACL2 program.
;; Leave these lines unchanged so that DrScheme can properly load this file.
#reader(planet "reader.ss" ("cce" "dracula.plt") "modular" "lang")
#| Team Steele
   Software Engineering I
   Ipsc

   Interface for the Personal Score Card Module mPSC
|#

(require "Ixmlminidom.lisp")

(defconst *psctableheadpre* "<table id=\"")
(defconst *psctableheadpost* "\" class=\"results\">")
(defconst *psctableheadcontents*
  (concatenate 'string
    "<tr>"
    "<th>Brd</th>"
    "<th colspan=\"2\">Versus</th>"
    "<th>Score</th>"
    "<th>Match Points</th>"
    "</tr>"))
(defconst *psctabletail* "</table>\n")

(interface Ipsc

  (include Ixmlminidom)

  ;Pulls the Personal Score Card data for all players, and put's them all
  ;into html table format
  ;XMLnodes format: Nodes format XXX update this to two args
  ;Output format: String, HTML formatted text comprising the score card
  ;   for one player pair
  (sig serializedPSC (gamenode boardnodes))

  ;Pulls the Name Strings for a given Pair ID
  ;PairID format: (String Direction, String SectionNumber)
  ;Data format: Nodes format
  ;Output format: (String String), Names of the two players
  (sig getNameForID (pairid data))

  ;;
  ;;
  (sig getBoardForPair (rbrds sections sectionlabel dir))

  ;Pulls the match results for a given Pair ID
  ;PairID format: (String Direction, String SectionNumber)
  ;Results format: ?
  ;XXX
  ;Output format: String, HTML formatted text comprising all the boards
  ;   for one player pair
  (sig getBoardsForPair (pairid sectionlabel results sections dir))

  ;;
  ;;
  (sig getAllPairs (results sections gamestring average top gamenode))
#|
  (con getPSC-con1
    (implies ()
      ()))

  (con getNameForID-Gives-Two-Strings
    (implies (and (listp pairid)
      (stringp (car pairid))
      (stringp (cadr pairid))

```

```
      (xml-isodelist data))
    (let (output (getNameForID pairid data))
      (and (listp output)
            (equal (len output) 2)
            (stringp (car output))
            (stringp (cadr output))))))

(con getBoardsForPair-con1
  (implies ()
    ())))
|#
)
```