

```

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
;server-email-test.lisp
;
;Created By: Wesley R. Howell
;
;This file will contain the Theorems and test for the server-email logic
;definitions in the server-email.lisp file
;
;--Change Log-----
;
; 2/28/2013 - File added to SVN and initial theorems proven
;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;

(include-book "server-email")

;(contactStructure-not-nil)
;Theorem to prove that a returned contact structure cannot be nil
;if a correct string is passed into the function
(defthm contactStructure-not-nil
  (implies (stringp str)
    (equal (listp (getContactStructure str))
      t)))

;(contactStructure-contents-are-strings)
;Theorem to prove that the contents of the atoms in a contact structure are
;strings
(defthm contactStructure-contents-are-strings
  (implies (stringp str)
    (AND(equal (stringp (car (getContactStructure str))) t)
      (equal (stringp (car (cdr (getContactStructure str)))) t))
    ))

;(contactStructure-domain-is-a-string)
;Theorem to prove that the contact structure's name is a string, the above
;Theorem contactStructure-contents-are-strings builds off this proof.
(defthm contactStructure-domain-is-a-string
  (implies (stringp str)
    (equal (stringp (car (cdr (getContactStructure str))))
      t)))

;(emailStructure-is-a-list)
;Theorem to prove that the emailStructure is returned from the function
;getEmailStructure
(defthm emailStructure-is-a-list
  (implies (listp xml)
    (equal (listp (getEmailStructure xml)) t)))

;(emailStructure-ToField-a-list)
;Theorem to prove the email's to structure is a list based on the
;data structure based on the project documentation.
(defthm emailStructure-ToField-a-list
  (implies (listp xml)
    (equal (listp (car (getEmailStructure xml)))
      t)))

;(getEmail-takes-list-returns-strings-on-header)
;Theorem to prove that the getEmail functions takes a list and returns
;a list of strings in XML Format.
(defthm getEmail-Takes-list-returns-strings-on-header
  (implies (listp str)
    (equal (stringp
      (car (getEmail str)))
      t)))

;(getEmail-returns-list-of-strings)
;Theorem built off the previous theorem that shows that each element in the
;list is a string.

```

```
(defthm getEmail-Returns-list-of-strings
  (implies (listp str)
    (AND (equal (listp (getEmail str)) t)
      (equal (stringp (car (getEmail str))) t))))
```