instructor

EXAMPLE_DATABASE_TESTS

feature --setup
 d: **DICTIONARY[STRING, INTEGER]**

feature -- Test for array comparison
test_array_comparison: BOOLEAN

feature -- Tests
test_setup: **BOOLEAN**

test_remove: **BOOLEAN**

test_add: **BOOLEAN**

test_iterable_dictionary: **BOOLEAN**

test_iteration_cursor: **BOOLEAN**

student

STUDENT_DATABASE_TESTS

feature --setup
 d:**DICTIONARY[STRING, INTEGER]**

feature-- Test for array comparison
test_array_comparison: BOOLEAN

feature-- Tests
test_setup:**BOOLEAN**

test_remove:**BOOLEAN**

test_add:**BOOLEAN**

test_iterable_dictionary:**BOOLEAN**

test_iteration_cursor:**BOOLEAN**

Model (Supplier)

ITERABLE *

new_cursor*: ITERATION_CURSOR[G]
! Result =/ Void

ITERATION_CURSOR[G]*

after*: BOOLEAN
item*: G
forth*

new_cursor*

DICTIONARY[V, K]+

class
 DICTIONARY[V, K]
 inherit
  ITERABLE[TUPLE[V, K]]

feature
model: FUN[k, v]
ensure
consistent_model_imp_counts:
Result.count = count

feature -- feature(s) required by ITERABLE
 new_cursor: TUPLE_ITERATION_CURSOR[TUPLE[V, K]]

feature --commands
 add_entry (v: **V**; k: **K**)
 -- Add a new entry into current dictionary
 require
non_existing_key:
  ¬ (k ∈ model.domain)
 ensure
  entry_added_to_model:
   model = (old model).extended(k, v)

ITERABLE_COLLECTION

ARRAY[G]+   LINKED_LIST[G]+

keys, values

new_cursor+

k, v

TUPLE_ITERATION_CURSOR[K, V]+

feature -- Constructor: initialize from array, linked list and linked list
make**(v: LINKED_LIST[V]**; **k: ARRAY[K]**)
feature -- Cursor Operations
item: **TUPLE[V, K]**
after: **BOOLEAN**
forth

d

d