{v} exercism Learn Discover Contribute More Insiders ♥ ● Tracks / ABAP / Exercises / Book Store Suh @SuellenMiranda You've completed Book Store. Your Journey Settings Book Store

© completed

medium Sign out Open in editor 🔍 Overview Your iterations 1 Community Solutions Code Review • Passed 🔘 zcl_book_store.clas.abap Analysis D Tests ALL TESTS PASSED 2 PUBLIC 3 FINAL 4 CREATE PUBLIC . √ 18 tests passed √ 6 PUBLIC SECTION. PASSED Test 1
 ONLY_A_SINGLE_BOOK 7 8 "! ID of book to buy from 1 to 5 9 TYPES book_id TYPE i. 11 TYPES basket_type TYPE SORTED TABLE OF book_id WITH NON-UNIQUE KEY table_line. PASSED Test 3
 EMPTY_BASKET PASSED Test 4
 TWO_DIFFERENT_BOOKS 18 IMPORTING
19 basket TYPE basket_type
20 RETURNING
21 VALUE(total) TYPE total. PASSED Test 5
 THREE_DIFFERENT_BOOKS PROTECTED SECTION.
PRIVATE SECTION. PASSED Test 6
 FOUR_DIFFERENT_BOOKS CONSTANTS price_per_book TYPE i VALUE 8. PASSED Test 7
 FIVE_DIFFERENT_BOOKS TYPES:

BEGIN OF ty_count,

book TYPE book_id,

count TYPE i,

END OF ty_count,

ty_counts TYPE TABLE OF ty_count WITH KEY book. PASSED Test 8
 TWO_GROUPS_OF_FOUR_1 PASSED Test 9
 TWO_GROUPS_OF_FOUR_2 TYPES:
BEGIN OF ty_discount, PASSED Test 10
GROUP_OF_FOUR_AND_GROUP_OF_TW books TYPE i,
amount TYPE total,
EMO OF Ty_discount,
ty_discounts TYPE TABLE OF ty_discount WITH KEY books. PASSED Test 11
 TWO_COPIES_OF_EACH_EXCEPT_ONE PASSED Test 12
 TWO_COPIES_OF_EACH_BOOK PASSED Test 13
THREE_OF_FIRST_TWO_EACH_REMAIN 48 METHOD calculate_total. PASSED Test 14
 THREE_OF_FIRST_TWO_TWO_OF_REM 49
50 DATA costs TYPE total.
51 DATA(discounts) = VALUE ty_discounts(
52 (books = 5 amount = '0.75')
53 (books = 4 amount = '0.80')
54 (books = 3 amount = '0.90')
55 (books = 2 amount = '0.95')
56 (books = 1 amount = '1.00')).
57 PASSED Test 15
 FOUR_GROUPS_OF_FOUR PASSED Test 16
 GROUPS_OF_FOUR total = 999.

DATA(counter) = VALUE ty_counts(
 (book = 1)
 (book = 2)
 (book = 3)
 (book = 4)
 (book = 5)). PASSED Test 17
 ONE_GROUP_OF_ONE_AND_FOUR PASSED Test 18
 MIXED_1 < LOOP AT basket INTO DATA(book).

counter[book = book]-count += 1.
ENDLOOP. LOOP AT discounts INTO DATA(discount) WHERE books <= size. remaining_books += 1.
ENDLOOP. IF remaining_books < discount-books.
EXIT.
ENDIF. SORT volumes BY count DESCENDING.

DO discount-books TIMES. volumes[sy-index]-count -= 1.
ENDDO. costs += price_per_book * discount-books * discount-amount.
ENDLOOP. 94 95 total = nmin(val1 = total val2 = costs). 98 ENDWHILE. 100 ENDMETHOD. < > >