*def* command\_add(*date*, *event\_details*, *calendar*):  
 '''  
 (str, str, dict) -> str  
 Add event\_details to the list at calendar[date]  
 Create date if it was not there  
  
 date: A string date formatted as "YYYY-MM-DD"  
 event\_details: A string describing the event  
 calendar: The calendar database  
 return: empty string  
  
 >>> calendar = {}  
 >>> command\_add("2017-02-28", "Python class", calendar)  
 ''  
 >>> calendar == {'2017-02-28': ['Python class']}  
 True  
 >>> command\_add("2017-03-11", "CSCA08 test 2", calendar)  
 ''  
 >>> calendar == {'2017-03-11': ['CSCA08 test 2'], '2017-02-28':\  
 ['Python class']}  
 True  
 >>> command\_add("2017-03-11", "go out with friends after test", calendar)  
 ''  
 >>> calendar == {'2017-03-11': ['CSCA08 test 2', \  
 'go out with friends after test'], '2017-02-28': ['Python class']}  
 True  
 >>>  
  
 '''  
  
 *if date in calendar*:  
 *calendar*[*date*].append(*event\_details*)  
 *else*:  
 *calendar*[*date*] = [*event\_details*]  
 *return* ''

*def* command\_show(*calendar*):  
 r'''  
 (dict) -> str  
 Returns the list of events for calendar sorted in increasing date order  
 as a string, see examples below for a sample formatting  
 calendar: the database of events  
  
 Example:  
 >>> calendar = {}  
 >>> command\_add("2017-02-12", "Eye doctor", calendar)  
 ''  
 >>> command\_add("2017-02-12", "lunch with sid", calendar)  
 ''  
 >>> command\_add("2017-03-29", "Change oil in blue car", calendar)  
 ''  
 >>> command\_add("2017-02-12", "dinner with Jane", calendar)  
 ''  
 >>> command\_add("2017-03-29", "Fix tree near front walkway", calendar)  
 ''  
 >>> command\_add("2017-03-29", "Get salad stuff", calendar)  
 ''  
 >>> command\_add("2017-05-06", "Sid's birthday", calendar)  
 ''  
 >>> command\_show(calendar)  
 "\n 2017-02-12:\n 0: Eye doctor\n 1: lunch with sid\n 2: dinner with Jane\n 2017-03-29:\n 0: Change oil in blue car\n 1: Fix tree near front walkway\n 2: Get salad stuff\n 2017-05-06:\n 0: Sid's birthday"  
 '''  
 # Hint: Don't use \t (the tab character) to indent, or DocTest will fail  
 # in the above testcase  
 # Put 4 spaces before the date, 8 spaces before each item.  
  
 *return* list(*calendar*.items())