

```
1  ##NO_TEXT
2  class LZCL_ZLOGGER_SRV_E_AU_LOGENTRY definition for testing
3    duration medium
4    risk level harmless.
5
6  private section.
7
8    DATA:
9      lp_tst_logentryset TYPE REF TO zcl_zlogger_srv_e_au_logentrys.
10
11
12    class-data L_LOGHANDLER Type  BALLOGHNDL.
13    class-data LT_LOGNUMBERS  Type  BAL_T_LGNM.
14
15    CLASS-METHODS: class_setup RAISING cx_ecatt_apl_odata_tst.
16    CLASS-METHODS: class_teardown RAISING cx_ecatt_apl_odata_tst.
17
18    METHODS: setup RAISING cx_ecatt_apl_odata_tst.
19    METHODS: teardown RAISING cx_ecatt_apl_odata_tst.
20
21    METHODS: t01_is_metadata__available FOR TESTING RAISING cx_ecatt_apl_odata_tst.
22    METHODS: util_retrieve_set_limited RAISING cx_ecatt_apl_odata_tst.
23    METHODS: t02_is_set__available FOR TESTING RAISING cx_ecatt_apl_odata_tst.
24    METHODS: t03_is_entity__available FOR TESTING RAISING cx_ecatt_apl_odata_tst.
25    METHODS: t11_is_insert_entity_available RAISING cx_ecatt_apl_odata_tst. " FOR TESTING.
26    METHODS: t12_is_update_entity_available RAISING cx_ecatt_apl_odata_tst. " FOR TESTING.
27    METHODS: t13_is_delete_entity_available RAISING cx_ecatt_apl_odata_tst. " FOR TESTING.
28    METHODS: test_get_entity__compare FOR TESTING RAISING cx_ecatt_apl_odata_tst.
29    METHODS: test_get_set__count FOR TESTING RAISING cx_ecatt_apl_odata_tst.
30    METHODS: test_get_set__filter FOR TESTING RAISING cx_ecatt_apl_odata_tst.
31    METHODS: test_get_set__compare FOR TESTING RAISING cx_ecatt_apl_odata_tst.
32
33
34  ENDCLASS.
35
36
37  CLASS LZCL_ZLOGGER_SRV_E_AU_LOGENTRY IMPLEMENTATION.
38
39
40  METHOD class_setup.
41    data ls_logentry type ZCL_ZLOGGER_SRV_E_SC=>TYS_E_LOGENTRY.
42    data ls_message type BAL_S_MSG.
43    data l_object type BALHDR-OBJECT.
44    data l_subobject type BALHDR-SUBOBJECT .
45    data s_log type bal_s_log.
46    data AP_TDC_LOGENTRYSET Type Ref To CL_APL_ECATT_TDC_API.
47    DATA lp_exception TYPE REF TO cx_ecatt_tdc_access.
48    data AT_EXP_LOGENTRYSET  Type  ZCL_ZLOGGER_SRV_E_SC_LOGENTRYS=>TYT_E_LOGENTRYSET.
49    data A_EXPC_LOGENTRYSET Type  I.
50
51    clear LT_LOGNUMBERS.
52    clear l_loghandler.
53
54    s_log-object = 'ZTEST'.
55    s_log-subobject = 'ZTEST'.
56
57    "INITIALIZE TEST DATA CONTAINER
58
59    DATA(l_tdc_error) = `Test Data Access:`.          "#EC NOTEXT
60    IF `ZTD_ZLOGGER_SRV_E_LOGENTRYSET` <> space.
61      TRY.
```

```
62
63     CALL METHOD cl_apl_ecatt_tdc_api=>get_instance
64     EXPORTING
65         i_testdatacontainer      = `ZTD_ZLOGGER_SRV_E_LOGENTRYSET`
66         i_testdatacontainer_version = 00000001
67     RECEIVING
68         e_tdc_ref                = ap_tdc_logentryset.
69
70     CATCH cx_ecatt_tdc_access INTO lp_exception.
71     "exit the complete unit test if opening TDC failed
72     CALL METHOD cl_abap_unit_assert=>fail
73     EXPORTING
74         msg      = l_tdc_error && lp_exception->get_text( )
75         level    = if_aunit_constants=>tolerable
76         quit     = if_aunit_constants=>method
77     *         detail =
78     .
79     ENDTRY.
80 ENDIF.
81
82
83 IF ap_tdc_logentryset IS BOUND.
84     "get the complete set of expected values from testdata container into at_exp_logentryset
85     data l_pname(30) type c.
86     l_pname = `COUNT_logentryset`.
87     TRY.
88         ap_tdc_logentryset->get_value(
89             EXPORTING
90                 i_param_name      = l_pname
91                 i_variant_name    = `TEST_RETRIEVE_ENTITY_SET`
92             CHANGING
93                 e_param_value    = A_EXPC_logentryset ).
94
95         ap_tdc_logentryset->get_value(
96             EXPORTING
97                 i_param_name      = `ENTITYSET_LOGENTRYSET`
98                 i_variant_name    = `TEST_RETRIEVE_ENTITY_SET`
99             CHANGING
100                e_param_value    = at_exp_logentryset ).
101     CATCH cx_ecatt_tdc_access INTO lp_exception.
102     "exit the complete unit test if test expectation is not available
103     CALL METHOD cl_abap_unit_assert=>fail
104     EXPORTING
105         msg      = l_tdc_error && lp_exception->get_text( )
106         quit     = if_aunit_constants=>method
107         level    = if_aunit_constants=>tolerable
108     *         detail =
109     .
110     ENDTRY.
111 ENDIF.
112
113 CALL FUNCTION 'BAL_LOG_CREATE'
114     EXPORTING
115         I_S_LOG          = s_log
116     IMPORTING
117         E_LOG_HANDLE    = l_loghandler
118     EXCEPTIONS
119         LOG_HEADER_INCONSISTENT = 1
120         OTHERS              = 2.
121 IF SY-SUBRC <> 0.
122     CALL METHOD cl_abap_unit_assert=>fail
```

```
123     EXPORTING
124         msg      = 'Cannot create setup data'
125         quit     = if_aunit_constants=>class
126         level    = if_aunit_constants=>tolerable
127 *         detail  =
128     .
129     ENDIF.
130
131     loop at at_exp_logentryset into ls_logentry.
132         clear ls_message.
133         ls_message-msgty = 'I'.
134         ls_message-msgid = '00'.
135         ls_message-msgno = '001'.
136         ls_message-msgvl = ls_logentry-text.
137
138         CALL FUNCTION 'BAL_LOG_MSG_ADD'
139             EXPORTING
140                 I_LOG_HANDLE      = l_loghandler
141                 I_S_MSG           = ls_message
142 *             IMPORTING
143 *                 E_S_MSG_HANDLE  =
144 *                 E_MSG_WAS_LOGGED =
145 *                 E_MSG_WAS_DISPLAYED =
146             EXCEPTIONS
147                 LOG_NOT_FOUND     = 1
148                 MSG_INCONSISTENT = 2
149                 LOG_IS_FULL       = 3
150                 OTHERS            = 4.
151
152         IF SY-SUBRC <> 0.
153             CALL METHOD cl_abap_unit_assert=>fail
154                 EXPORTING
155                     msg      = 'Cannot create setup data'
156                     quit     = if_aunit_constants=>class
157                     level    = if_aunit_constants=>tolerable
158 *                     detail  =
159         .
160         ENDIF.
161     endloop.
162
163     data lt_loghandlers type BAL_T_LOGH.
164
165     append l_loghandler to lt_loghandlers.
166     CALL FUNCTION 'BAL_DB_SAVE'
167         EXPORTING
168 *             I_CLIENT           = SY-MANDT
169 *             I_IN_UPDATE_TASK  = ' '
170 *             I_SAVE_ALL        = 'X'
171 *             I_T_LOG_HANDLE    = lt_loghandlers
172 *             I_2TH_CONNECTION = ' '
173 *             I_2TH_CONNECT_COMMIT = ' '
174 *             I_LINK2JOB        = 'X'
175 *             IMPORTING
176 *                 E_NEW_LOGNUMBERS = lt_lognumbers
177 *                 E_SECOND_CONNECTION =
178 *             EXCEPTIONS
179 *                 LOG_NOT_FOUND     = 1
180 *                 SAVE_NOT_ALLOWED = 2
181 *                 NUMBERING_ERROR  = 3
182 *                 OTHERS            = 4.
183
```

```
184     IF SY-SUBRC <> 0.
185 * Implement suitable error handling here
186     ENDIF.
187
188
189     IF SY-SUBRC <> 0.
190         CALL METHOD cl_abap_unit_assert=>fail
191             EXPORTING
192                 msg      = 'Cannot create setup data'
193                 quit     = if_aunit_constants=>class
194                 level    = if_aunit_constants=>tolerable
195 *                 detail =
196             .
197     ENDIF.
198
199     COMMIT WORK.
200
201 ENDMETHOD.
202
203
204 METHOD class_teardown.
205
206     data lt_numbers type BALHDR_T.
207     data ls_filter type BAL_S_LFIL.
208     data ls_obj type BAL_S_OBJ.
209     data ls_sub type BAL_S_SUB.
210
211     ls_obj-SIGN = 'I'.
212     ls_obj-OPTION = 'EQ'.
213     ls_obj-LOW = 'ZTEST'.
214
215     append ls_obj to ls_filter-object.
216     MOVE-CORRESPONDING ls_obj to ls_sub.
217     append ls_sub to ls_filter-subobject.
218
219     wait up to 5 seconds.
220
221     CALL FUNCTION 'BAL_DB_SEARCH'
222         EXPORTING
223 *             I_CLIENT           = SY-MANDT
224             I_S_LOG_FILTER      = ls_filter
225 *             I_T_SEL_FIELD     =
226 *             I_TZONE           =
227         IMPORTING
228             E_T_LOG_HEADER      = lt_numbers
229         EXCEPTIONS
230             LOG_NOT_FOUND      = 1
231             NO_FILTER_CRITERIA = 2
232             OTHERS             = 3.
233     IF SY-SUBRC <> 0.
234 * Implement suitable error handling here
235     ENDIF.
236
237
238     CALL FUNCTION 'BAL_DB_DELETE'
239         EXPORTING
240             I_T_LOGS_TO_DELETE = lt_numbers
241 *             I_T_LOG_HANDLE    = l_loghandler
242 *             I_T_LOGNUMBER     =
243 *             I_CLIENT          = SY-MANDT
244 *             I_IN_UPDATE_TASK = ' '

```

```
245         I_WITH_COMMIT_WORK = 'X'
246 *       I_PACKAGE_SIZE      = 100
247     EXCEPTIONS
248         NO_LOGS_SPECIFIED = 1
249         OTHERS            = 2.
250     IF SY-SUBRC <> 0.
251 * Implement suitable error handling here
252     ENDIF.
253
254     ENDMETHOD.
255
256     METHOD setup.
257
258         CREATE OBJECT lp_tst_logentryset.
259         lp_tst_logentryset->setup( ).
260
261     ENDMETHOD.
262
263
264     METHOD teardown.
265
266         lp_tst_logentryset->teardown( ).
267
268     ENDMETHOD.
269
270
271     METHOD t01_is_metadata__available.
272
273         lp_tst_logentryset->t01_is_metadata__available( ).
274
275     ENDMETHOD.
276
277
278     METHOD util_retrieve_set_limited.
279
280         lp_tst_logentryset->util_retrieve_set_limited( ).
281
282     ENDMETHOD.
283
284
285     METHOD t02_is_set__available.
286
287         lp_tst_logentryset->t02_is_set__available( ).
288
289     ENDMETHOD.
290
291
292     METHOD t03_is_entity__available.
293
294         lp_tst_logentryset->t03_is_entity__available( ).
295
296     ENDMETHOD.
297
298
299     METHOD t11_is_insert_entity_available.
300
301         lp_tst_logentryset->t11_is_insert_entity_available( ).
302
303     ENDMETHOD.
304
305
```

```
306 METHOD t12_is_update_entity_available.
307
308     lp_tst_logentryset->t12_is_update_entity_available( ).
309
310 ENDMETHOD.
311
312 METHOD t13_is_delete_entity_available.
313
314     lp_tst_logentryset->t13_is_delete_entity_available( ).
315
316 ENDMETHOD.
317
318
319 METHOD test_get_entity__compare.
320
321     lp_tst_logentryset->test_get_entity__compare( ).
322
323 ENDMETHOD.
324
325
326
327
328
329 METHOD test_get_set__count.
330
331     lp_tst_logentryset->test_get_set__count( ).
332
333 ENDMETHOD.
334
335
336 METHOD test_get_set__filter.
337
338     lp_tst_logentryset->test_get_set__filter( ).
339
340 ENDMETHOD.
341
342
343 METHOD test_get_set__compare.
344
345     lp_tst_logentryset->test_get_set__compare( ).
346
347 ENDMETHOD.
348
349
350
351
352
353 endclass.
```