

[illegible]

```

cardinal {
    padding: 5px 5px 5px;
    border: 1px solid black;
    display: inline-block;
    padding: 5px 10px 5px 10px;
    width: 100px;
    height: 100px;
    margin: 5px;
}

cardinal h1 {
    font-size: 24px;
    margin: 0;
}

cardinal p {
    margin: 0;
}

cardinal h2 {
    font-size: 18px;
    margin: 0;
}

cardinal h3 {
    font-size: 16px;
    margin: 0;
}

cardinal h4 {
    font-size: 14px;
    margin: 0;
}

cardinal h5 {
    font-size: 12px;
    margin: 0;
}

cardinal h6 {
    font-size: 10px;
    margin: 0;
}

cardinal h7 {
    font-size: 8px;
    margin: 0;
}

cardinal h8 {
    font-size: 6px;
    margin: 0;
}

cardinal h9 {
    font-size: 4px;
    margin: 0;
}

cardinal h10 {
    font-size: 2px;
    margin: 0;
}

cardinal h11 {
    font-size: 1px;
    margin: 0;
}

cardinal h12 {
    font-size: 0.5px;
    margin: 0;
}

cardinal h13 {
    font-size: 0.25px;
    margin: 0;
}

cardinal h14 {
    font-size: 0.125px;
    margin: 0;
}

cardinal h15 {
    font-size: 0.0625px;
    margin: 0;
}

cardinal h16 {
    font-size: 0.03125px;
    margin: 0;
}

cardinal h17 {
    font-size: 0.015625px;
    margin: 0;
}

cardinal h18 {
    font-size: 0.0078125px;
    margin: 0;
}

cardinal h19 {
    font-size: 0.00390625px;
    margin: 0;
}

cardinal h20 {
    font-size: 0.001953125px;
    margin: 0;
}

cardinal h21 {
    font-size: 0.0009765625px;
    margin: 0;
}

cardinal h22 {
    font-size: 0.00048828125px;
    margin: 0;
}

cardinal h23 {
    font-size: 0.000244140625px;
    margin: 0;
}

cardinal h24 {
    font-size: 0.0001220703125px;
    margin: 0;
}

cardinal h25 {
    font-size: 6.103515625e-05px;
    margin: 0;
}

cardinal h26 {
    font-size: 3.0517578125e-05px;
    margin: 0;
}

cardinal h27 {
    font-size: 1.52587890625e-05px;
    margin: 0;
}

cardinal h28 {
    font-size: 7.62939453125e-06px;
    margin: 0;
}

cardinal h29 {
    font-size: 3.814697265625e-06px;
    margin: 0;
}

cardinal h30 {
    font-size: 1.9073486328125e-06px;
    margin: 0;
}

cardinal h31 {
    font-size: 9.5367431640625e-07px;
    margin: 0;
}

cardinal h32 {
    font-size: 4.76837158203125e-07px;
    margin: 0;
}

cardinal h33 {
    font-size: 2.384185791015625e-07px;
    margin: 0;
}

cardinal h34 {
    font-size: 1.1920928955078125e-07px;
    margin: 0;
}

cardinal h35 {
    font-size: 5.9604644775390625e-08px;
    margin: 0;
}

cardinal h36 {
    font-size: 2.98023223876953125e-08px;
    margin: 0;
}

cardinal h37 {
    font-size: 1.490116119384765625e-08px;
    margin: 0;
}

cardinal h38 {
    font-size: 7.450580596923828125e-09px;
    margin: 0;
}

cardinal h39 {
    font-size: 3.7252902984619140625e-09px;
    margin: 0;
}

cardinal h40 {
    font-size: 1.86264514923095703125e-09px;
    margin: 0;
}

cardinal h41 {
    font-size: 9.31322574615478515625e-10px;
    margin: 0;
}

cardinal h42 {
    font-size: 4.656612873077392578125e-10px;
    margin: 0;
}

cardinal h43 {
    font-size: 2.3283064365386962890625e-10px;
    margin: 0;
}

cardinal h44 {
    font-size: 1.16415321826934814453125e-10px;
    margin: 0;
}

cardinal h45 {
    font-size: 5.82076609134674072265625e-11px;
    margin: 0;
}

cardinal h46 {
    font-size: 2.910383045673370361328125e-11px;
    margin: 0;
}

cardinal h47 {
    font-size: 1.4551915228366851806640625e-11px;
    margin: 0;
}

cardinal h48 {
    font-size: 7.2759576141834259033203125e-12px;
    margin: 0;
}

cardinal h49 {
    font-size: 3.63797880709171295166015625e-12px;
    margin: 0;
}

cardinal h50 {
    font-size: 1.818989403545856475830078125e-12px;
    margin: 0;
}

cardinal h51 {
    font-size: 9.094947017729282379150390625e-13px;
    margin: 0;
}

cardinal h52 {
    font-size: 4.5474735088646411895751953125e-13px;
    margin: 0;
}

cardinal h53 {
    font-size: 2.27373675443232059478759765625e-13px;
    margin: 0;
}

cardinal h54 {
    font-size: 1.136868377216160297393798828125e-13px;
    margin: 0;
}

cardinal h55 {
    font-size: 5.684341886080801486968994140625e-14px;
    margin: 0;
}

cardinal h56 {
    font-size: 2.8421709430404007434844970703125e-14px;
    margin: 0;
}

cardinal h57 {
    font-size: 1.42108547152020037174224853515625e-14px;
    margin: 0;
}

cardinal h58 {
    font-size: 7.10542735760100185871124267578125e-15px;
    margin: 0;
}

cardinal h59 {
    font-size: 3.552713678800500929355621337890625e-15px;
    margin: 0;
}

cardinal h60 {
    font-size: 1.7763568394002504646778106689453125e-15px;
    margin: 0;
}

cardinal h61 {
    font-size: 8.8817841970012523233890533447265625e-16px;
    margin: 0;
}

cardinal h62 {
    font-size: 4.44089209850062616169452667236328125e-16px;
    margin: 0;
}

cardinal h63 {
    font-size: 2.220446049250313080847263336181640625e-16px;
    margin: 0;
}

cardinal h64 {
    font-size: 1.1102230246251565404236316680908203125e-16px;
    margin: 0;
}

cardinal h65 {
    font-size: 5.5511151231257827021181583404541015625e-17px;
    margin: 0;
}

cardinal h66 {
    font-size: 2.77555756156289135105907917022705078125e-17px;
    margin: 0;
}

cardinal h67 {
    font-size: 1.387778780781445675529539585113525390625e-17px;
    margin: 0;
}

cardinal h68 {
    font-size: 6.938893903907228377647697925567626953125e-18px;
    margin: 0;
}

cardinal h69 {
    font-size: 3.4694469519536141888238489627838134765625e-18px;
    margin: 0;
}

cardinal h70 {
    font-size: 1.73472347597680709441192448139190673828125e-18px;
    margin: 0;
}

cardinal h71 {
    font-size: 8.67361737988403547205962240695953369140625e-19px;
    margin: 0;
}

cardinal h72 {
    font-size: 4.336808689942017736029811203479766845703125e-19px;
    margin: 0;
}

cardinal h73 {
    font-size: 2.1684043449710088680149056017398834228515625e-19px;
    margin: 0;
}

cardinal h74 {
    font-size: 1.08420217248550443400745280086994171142578125e-19px;
    margin: 0;
}

cardinal h75 {
    font-size: 5.4210108624275221700122640004349585571265625e-20px;
    margin: 0;
}

cardinal h76 {
    font-size: 2.71050543121376108500613200021747927856328125e-20px;
    margin: 0;
}

cardinal h77 {
    font-size: 1.355252715606880542503066000108739639281640625e-20px;
    margin: 0;
}

cardinal h78 {
    font-size: 6.776263578034402712515330000543698196408203125e-21px;
    margin: 0;
}

cardinal h79 {
    font-size: 3.3881317890172013562576650002718490982041015625e-21px;
    margin: 0;
}

cardinal h80 {
    font-size: 1.69406589450860067812883250013592454910205078125e-21px;
    margin: 0;
}

cardinal h81 {
    font-size: 8.4703294725
```

```

1  @param {String} [url] - URL to fetch
2  @param {Object} [options] - Options
3  @param {Boolean} [options.cache] - Whether to use the browser's cache
4  @param {Boolean} [options.credentials] - Whether to include credentials
5  @param {Boolean} [options.headers] - Headers to include
6  @param {Boolean} [options.method] - Method to use
7  @param {Boolean} [options.mode] - Mode to use
8  @param {Boolean} [options.referrer] - Referrer to use
9  @param {Boolean} [options.timeout] - Timeout in milliseconds
10 @param {Boolean} [options.withCredentials] - Whether to include credentials
11 @return {Promise} - Promise that resolves to the response
12 @throws {Error} - If the request fails
13
14 @example
15 // Fetch a page
16 fetch(url, {
17   headers: {
18     'Content-Type': 'application/json'
19   },
20   method: 'POST',
21   body: JSON.stringify({ name: 'John' })
22 })
23
24 // Fetch a page with credentials
25 fetch(url, {
26   credentials: 'include'
27 })
28
29 // Fetch a page with a timeout
30 fetch(url, {
31   timeout: 5000
32 })
33
34 // Fetch a page with a referrer
35 fetch(url, {
36   referrer: 'https://example.com'
37 })
38
39 // Fetch a page with a mode
40 fetch(url, {
41   mode: 'cors'
42 })
43
44 // Fetch a page with a cache
45 fetch(url, {
46   cache: 'no-cache'
47 })
48
49 // Fetch a page with a method
50 fetch(url, {
51   method: 'GET'
52 })
53
54 // Fetch a page with a mode
55 fetch(url, {
56   mode: 'cors'
57 })
58
59 // Fetch a page with a cache
60 fetch(url, {
61   cache: 'no-cache'
62 })
63
64 // Fetch a page with a method
65 fetch(url, {
66   method: 'GET'
67 })
68
69 // Fetch a page with a mode
70 fetch(url, {
71   mode: 'cors'
72 })
73
74 // Fetch a page with a cache
75 fetch(url, {
76   cache: 'no-cache'
77 })
78
79 // Fetch a page with a method
80 fetch(url, {
81   method: 'GET'
82 })
83
84 // Fetch a page with a mode
85 fetch(url, {
86   mode: 'cors'
87 })
88
89 // Fetch a page with a cache
90 fetch(url, {
91   cache: 'no-cache'
92 })
93
94 // Fetch a page with a method
95 fetch(url, {
96   method: 'GET'
97 })
98
99 // Fetch a page with a mode
100 fetch(url, {
101   mode: 'cors'
102 })
103
104 // Fetch a page with a cache
105 fetch(url, {
106   cache: 'no-cache'
107 })
108
109 // Fetch a page with a method
110 fetch(url, {
111   method: 'GET'
112 })
113
114 // Fetch a page with a mode
115 fetch(url, {
116   mode: 'cors'
117 })
118
119 // Fetch a page with a cache
120 fetch(url, {
121   cache: 'no-cache'
122 })
123
124 // Fetch a page with a method
125 fetch(url, {
126   method: 'GET'
127 })
128
129 // Fetch a page with a mode
130 fetch(url, {
131   mode: 'cors'
132 })
133
134 // Fetch a page with a cache
135 fetch(url, {
136   cache: 'no-cache'
137 })
138
139 // Fetch a page with a method
140 fetch(url, {
141   method: 'GET'
142 })
143
144 // Fetch a page with a mode
145 fetch(url, {
146   mode: 'cors'
147 })
148
149 // Fetch a page with a cache
150 fetch(url, {
151   cache: 'no-cache'
152 })
153
154 // Fetch a page with a method
155 fetch(url, {
156   method: 'GET'
157 })
158
159 // Fetch a page with a mode
160 fetch(url, {
161   mode: 'cors'
162 })
163
164 // Fetch a page with a cache
165 fetch(url, {
166   cache: 'no-cache'
167 })
168
169 // Fetch a page with a method
170 fetch(url, {
171   method: 'GET'
172 })
173
174 // Fetch a page with a mode
175 fetch(url, {
176   mode: 'cors'
177 })
178
179 // Fetch a page with a cache
180 fetch(url, {
181   cache: 'no-cache'
182 })
183
184 // Fetch a page with a method
185 fetch(url, {
186   method: 'GET'
187 })
188
189 // Fetch a page with a mode
190 fetch(url, {
191   mode: 'cors'
192 })
193
194 // Fetch a page with a cache
195 fetch(url, {
196   cache: 'no-cache'
197 })
198
199 // Fetch a page with a method
200 fetch(url, {
201   method: 'GET'
202 })
203
204 // Fetch a page with a mode
205 fetch(url, {
206   mode: 'cors'
207 })
208
209 // Fetch a page with a cache
210 fetch(url, {
211   cache: 'no-cache'
212 })
213
214 // Fetch a page with a method
215 fetch(url, {
216   method: 'GET'
217 })
218
219 // Fetch a page with a mode
220 fetch(url, {
221   mode: 'cors'
222 })
223
224 // Fetch a page with a cache
225 fetch(url, {
226   cache: 'no-cache'
227 })
228
229 // Fetch a page with a method
230 fetch(url, {
231   method: 'GET'
232 })
233
234 // Fetch a page with a mode
235 fetch(url, {
236   mode: 'cors'
237 })
238
239 // Fetch a page with a cache
240 fetch(url, {
241   cache: 'no-cache'
242 })
243
244 // Fetch a page with a method
245 fetch(url, {
246   method: 'GET'
247 })
248
249 // Fetch a page with a mode
250 fetch(url, {
251   mode: 'cors'
252 })
253
254 // Fetch a page with a cache
255 fetch(url, {
256   cache: 'no-cache'
257 })
258
259 // Fetch a page with a method
260 fetch(url, {
261   method: 'GET'
262 })
263
264 // Fetch a page with a mode
265 fetch(url, {
266   mode: 'cors'
267 })
268
269 // Fetch a page with a cache
270 fetch(url, {
271   cache: 'no-cache'
272 })
273
274 // Fetch a page with a method
275 fetch(url, {
276   method: 'GET'
277 })
278
279 // Fetch a page with a mode
280 fetch(url, {
281   mode: 'cors'
282 })
283
284 // Fetch a page with a cache
285 fetch(url, {
286   cache: 'no-cache'
287 })
288
289 // Fetch a page with a method
290 fetch(url, {
291   method: 'GET'
292 })
293
294 // Fetch a page with a mode
295 fetch(url, {
296   mode: 'cors'
297 })
298
299 // Fetch a page with a cache
300 fetch(url, {
301   cache: 'no-cache'
302 })
303
304 // Fetch a page with a method
305 fetch(url, {
306   method: 'GET'
307 })
308
309 // Fetch a page with a mode
310 fetch(url, {
311   mode: 'cors'
312 })
313
314 // Fetch a page with a cache
315 fetch(url, {
316   cache: 'no-cache'
317 })
318
319 // Fetch a page with a method
320 fetch(url, {
321   method: 'GET'
322 })
323
324 // Fetch a page with a mode
325 fetch(url, {
326   mode: 'cors'
327 })
328
329 // Fetch a page with a cache
330 fetch(url, {
331   cache: 'no-cache'
332 })
333
334 // Fetch a page with a method
335 fetch(url, {
336   method: 'GET'
337 })
338
339 // Fetch a page with a mode
340 fetch(url, {
341   mode: 'cors'
342 })
343
344 // Fetch a page with a cache
345 fetch(url, {
346   cache: 'no-cache'
347 })
348
349 // Fetch a page with a method
350 fetch(url, {
351   method: 'GET'
352 })
353
354 // Fetch a page with a mode
355 fetch(url, {
356   mode: 'cors'
357 })
358
359 // Fetch a page with a cache
360 fetch(url, {
361   cache: 'no-cache'
362 })
363
364 // Fetch a page with a method
365 fetch(url, {
366   method: 'GET'
367 })
368
369 // Fetch a page with a mode
370 fetch(url, {
371   mode: 'cors'
372 })
373
374 // Fetch a page with a cache
375 fetch(url, {
376   cache: 'no-cache'
377 })
378
379 // Fetch a page with a method
380 fetch(url, {
381   method: 'GET'
382 })
383
384 // Fetch a page with a mode
385 fetch(url, {
386   mode: 'cors'
387 })
388
389 // Fetch a page with a cache
390 fetch(url, {
391   cache: 'no-cache'
392 })
393
394 // Fetch a page with a method
395 fetch(url, {
396   method: 'GET'
397 })
398
399 // Fetch a page with a mode
400 fetch(url, {
401   mode: 'cors'
402 })
403
404 // Fetch a page with a cache
405 fetch(url, {
406   cache: 'no-cache'
407 })
408
409 // Fetch a page with a method
410 fetch(url, {
411   method: 'GET'
412 })
413
414 // Fetch a page with a mode
415 fetch(url, {
416   mode: 'cors'
417 })
418
419 // Fetch a page with a cache
420 fetch(url, {
421   cache: 'no-cache'
422 })
423
424 // Fetch a page with a method
425 fetch(url, {
426   method: 'GET'
427 })
428
429 // Fetch a page with a mode
430 fetch(url, {
431   mode: 'cors'
432 })
433
434 // Fetch a page with a cache
435 fetch(url, {
436   cache: 'no-cache'
437 })
438
439 // Fetch a page with a method
440 fetch(url, {
441   method: 'GET'
442 })
443
444 // Fetch a page with a mode
445 fetch(url, {
446   mode: 'cors'
447 })
448
449 // Fetch a page with a cache
450 fetch(url, {
451   cache: 'no-cache'
452 })
453
454 // Fetch a page with a method
455 fetch(url, {
456   method: 'GET'
457 })
458
459 // Fetch a page with a mode
460 fetch(url, {
461   mode: 'cors'
462 })
463
464 // Fetch a page with a cache
465 fetch(url, {
466   cache: 'no-cache'
467 })
468
469 // Fetch a page with a method
470 fetch(url, {
471   method: 'GET'
472 })
473
474 // Fetch a page with a mode
475 fetch(url, {
476   mode: 'cors'
477 })
478
479 // Fetch a page with a cache
480 fetch(url, {
481   cache: 'no-cache'
482 })
483
484 // Fetch a page with a method
485 fetch(url, {
486   method: 'GET'
487 })
488
489 // Fetch a page with a mode
490 fetch(url, {
491   mode: 'cors'
492 })
493
494 // Fetch a page with a cache
495 fetch(url, {
496   cache: 'no-cache'
497 })
498
499 // Fetch a page with a method
500 fetch(url, {
501   method: 'GET'
502 })
503
504 // Fetch a page with a mode
505 fetch(url, {
506   mode: 'cors'
507 })
508
509 // Fetch a page with a cache
510 fetch(url, {
511   cache: 'no-cache'
512 })
513
514 // Fetch a page with a method
515 fetch(url, {
516   method: 'GET'
517 })
518
519 // Fetch a page with a mode
520 fetch(url, {
521   mode: 'cors'
522 })
523
524 // Fetch a page with a cache
525 fetch(url, {
526   cache: 'no-cache'
527 })
528
529 // Fetch a page with a method
530 fetch(url, {
531
```

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]


```

1 // @file: /src/.../vuln/cv2_style_style_common_funcs.h | Definition
2
3 #include <...>
4
5 struct... {
6     ...
7     ...
8     ...
9     ...
10 }
11
12 ...
13
14 ...
15
16 ...
17
18 ...
19
20 ...
21
22 ...
23
24 ...
25
26 ...
27
28 ...
29
30 ...
31
32 ...
33
34 ...
35
36 ...
37
38 ...
39
40 ...
41
42 ...
43
44 ...
45
46 ...
47
48 ...
49
50 ...
51
52 ...
53
54 ...
55
56 ...
57
58 ...
59
60 ...
61
62 ...
63
64 ...
65
66 ...
67
68 ...
69
70 ...
71
72 ...
73
74 ...
75
76 ...
77
78 ...
79
80 ...
81
82 ...
83
84 ...
85
86 ...
87
88 ...
89
90 ...
91
92 ...
93
94 ...
95
96 ...
97
98 ...
99
100 ...
101
102 ...
103
104 ...
105
106 ...
107
108 ...
109
110 ...
111
112 ...
113
114 ...
115
116 ...
117
118 ...
119
120 ...
121
122 ...
123
124 ...
125
126 ...
127
128 ...
129
130 ...
131
132 ...
133
134 ...
135
136 ...
137
138 ...
139
140 ...
141
142 ...
143
144 ...
145
146 ...
147
148 ...
149
150 ...
151
152 ...
153
154 ...
155
156 ...
157
158 ...
159
160 ...
161
162 ...
163
164 ...
165
166 ...
167
168 ...
169
170 ...
171
172 ...
173
174 ...
175
176 ...
177
178 ...
179
180 ...
181
182 ...
183
184 ...
185
186 ...
187
188 ...
189
190 ...
191
192 ...
193
194 ...
195
196 ...
197
198 ...
199
200 ...
201
202 ...
203
204 ...
205
206 ...
207
208 ...
209
210 ...
211
212 ...
213
214 ...
215
216 ...
217
218 ...
219
220 ...
221
222 ...
223
224 ...
225
226 ...
227
228 ...
229
230 ...
231
232 ...
233
234 ...
235
236 ...
237
238 ...
239
240 ...
241
242 ...
243
244 ...
245
246 ...
247
248 ...
249
250 ...
251
252 ...
253
254 ...
255
256 ...
257
258 ...
259
260 ...
261
262 ...
263
264 ...
265
266 ...
267
268 ...
269
270 ...
271
272 ...
273
274 ...
275
276 ...
277
278 ...
279
280 ...
281
282 ...
283
284 ...
285
286 ...
287
288 ...
289
290 ...
291
292 ...
293
294 ...
295
296 ...
297
298 ...
299
300 ...
301
302 ...
303
304 ...
305
306 ...
307
308 ...
309
310 ...
311
312 ...
313
314 ...
315
316 ...
317
318 ...
319
320 ...
321
322 ...
323
324 ...
325
326 ...
327
328 ...
329
330 ...
331
332 ...
333
334 ...
335
336 ...
337
338 ...
339
340 ...
341
342 ...
343
344 ...
345
346 ...
347
348 ...
349
350 ...
351
352 ...
353
354 ...
355
356 ...
357
358 ...
359
360 ...
361
362 ...
363
364 ...
365
366 ...
367
368 ...
369
370 ...
371
372 ...
373
374 ...
375
376 ...
377
378 ...
379
380 ...
381
382 ...
383
384 ...
385
386 ...
387
388 ...
389
390 ...
391
392 ...
393
394 ...
395
396 ...
397
398 ...
399
400 ...
401
402 ...
403
404 ...
405
406 ...
407
408 ...
409
410 ...
411
412 ...
413
414 ...
415
416 ...
417
418 ...
419
420 ...
421
422 ...
423
424 ...
425
426 ...
427
428 ...
429
430 ...
431
432 ...
433
434 ...
435
436 ...
437
438 ...
439
440 ...
441
442 ...
443
444 ...
445
446 ...
447
448 ...
449
450 ...
451
452 ...
453
454 ...
455
456 ...
457
458 ...
459
460 ...
461
462 ...
463
464 ...
465
466 ...
467
468 ...
469
470 ...
471
472 ...
473
474 ...
475
476 ...
477
478 ...
479
480 ...
481
482 ...
483
484 ...
485
486 ...
487
488 ...
489
490 ...
491
492 ...
493
494 ...
495
496 ...
497
498 ...
499
500 ...
501
502 ...
503
504 ...
505
506 ...
507
508 ...
509
510 ...
511
512 ...
513
514 ...
515
516 ...
517
518 ...
519
520 ...
521
522 ...
523
524 ...
525
526 ...
527
528 ...
529
530 ...
531
532 ...
533
534 ...
535
536 ...
537
538 ...
539
540 ...
541
542 ...
543
544 ...
545
546 ...
547
548 ...
549
550 ...
551
552 ...
553
554 ...
555
556 ...
557
558 ...
559
560 ...
561
562 ...
563
564 ...
565
566 ...
567
568 ...
569
570 ...
571
572 ...
573
574 ...
575
576 ...
577
578 ...
579
580 ...
581
582 ...
583
584 ...
585
586 ...
587
588 ...
589
590 ...
591
592 ...
593
594 ...
595
596 ...
597
598 ...
599
600 ...
601
602 ...
603
604 ...
605
606 ...
607
608 ...
609
610 ...
611
612 ...
613
614 ...
615
616 ...
617
618 ...
619
620 ...
621
622 ...
623
624 ...
625
626 ...
627
628 ...
629
630 ...
631
632 ...
633
634 ...
635
636 ...
637
638 ...
639
640 ...
641
642 ...
643
644 ...
645
646 ...
647
648 ...
649
650 ...
651
652 ...
653
654 ...
655
656 ...
657
658 ...
659
660 ...
661
662 ...
663
664 ...
665
666 ...
667
668 ...
669
670 ...
671
672 ...
673
674 ...
675
676 ...
677
678 ...
679
680 ...
681
682 ...
683
684 ...
685
686 ...
687
688 ...
689
690 ...
691
692 ...
693
694 ...
695
696 ...
697
698 ...
699
700 ...
701
702 ...
703
704 ...
705
706 ...
707
708 ...
709
710 ...
711
712 ...
713
714 ...
715
716 ...
717
718 ...
719
720 ...
721
722 ...
723
724 ...
725
726 ...
727
728 ...
729
730 ...
731
732 ...
733
734 ...
735
736 ...
737
738 ...
739
740 ...
741
742 ...
743
744 ...
745
746 ...
747
748 ...
749
750 ...
751
752 ...
753
754 ...
755
756 ...
757
758 ...
759
760 ...
761
762 ...
763
764 ...
765
766 ...
767
768 ...
769
770 ...
771
772 ...
773
774 ...
775
776 ...
777
778 ...
779
780 ...
781
782 ...
783
784 ...
785
786 ...
787
788 ...
789
790 ...
791
792 ...
793
794 ...
795
796 ...
797
798 ...
799
800 ...
801
802 ...
803
804 ...
805
806 ...
807
808 ...
809
810 ...
811
812 ...
813
814 ...
815
816 ...
817
818 ...
819
820 ...
821
822 ...
823
824 ...
825
826 ...
827
828 ...
829
830 ...
831
832 ...
833
834 ...
835
836 ...
837
838 ...
839
840 ...
841
842 ...
843
844 ...
845
846 ...
847
848 ...
849
850 ...
851
852 ...
853
854 ...
855
856 ...
857
858 ...
859
860 ...
861
862 ...
863
864 ...
865
866 ...
867
868 ...
869
870 ...
871
872 ...
873
874 ...
875
876 ...
877
878 ...
879
880 ...
881
882 ...
883
884 ...
885
886 ...
887
888 ...
889
890 ...
891
892 ...
893
894 ...
895
896 ...
897
898 ...
899
900 ...
901
902 ...
903
904 ...
905
906 ...
907
908 ...
909
910 ...
911
912 ...
913
914 ...
915
916 ...
917
918 ...
919
920 ...
921
922 ...
923
924 ...
925
9
```


[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible][illegible]

```
-- File Content End --

❖ tasks_updateTask.js

-- File Content Start --

import {functionName} from './updateTask.js'
case {
  values: {id, admin,
    category, productId, startDate, endDate,
    getActiveWindows, callFunction,
    addLog } = require('.../common')
  case (sequence) { 'require('.../auth);

  async function test() {
    let {data} = await functionName({
      id: id,
      admin: admin,
      productId: productId,
      startDate: startDate,
      endDate: endDate,
      getActiveWindows: getActiveWindows,
      callFunction: callFunction,
      addLog: addLog
    })
    return {data}
  }
}
```

[illegible]

[illegible]

[illegible]