

Project Explorer | Navigator (Deprecated) | OriginConfiguration.java | MongoDBConfiguration.java | ConnectionDetails.java

170041\_usrao [170041\_usrao master]  
am-d-servicengorder [boot] [am-d-servicengorder master]  
com.citi.171384.usretailapi.checkbook.component [in rti-pdm-d-checkbook] [com.citi.171384.usretailapi.checkbook.component master]  
CommonUtility [usrao-utility\_repo feature/URS-584]  
fdn-c-eodjobs [boot] [fdn-c-eodjobs bug/eoddefect]  
GdfUtility [usrao-utility\_repo feature/URS-584]  
mongodb [mongodb feature/AugScriptUpdate]  
ogn-c-onboardingmongodb-usretail [boot] [ogn-c-onboardingmongodb-usretail master]

src/main/java  
com.citigroup.usrao.mongodb  
com.citigroup.usrao.mongodb.config  
OriginConfiguration.java  
com.citigroup.usrao.mongodb.configuration  
CommonConfigurations.java  
ConnectionDetails.java  
MongoDbConfiguration.java  
SwaggerConfiguration.java  
SwaggerController.java  
com.citigroup.usrao.mongodb.controller  
MongoDBController.java  
MongoDBController  
com.citigroup.usrao.mongodb.domain  
com.citigroup.usrao.mongodb.event  
com.citigroup.usrao.mongodb.exception  
com.citigroup.usrao.mongodb.model  
com.citigroup.usrao.mongodb.resttemplate  
com.citigroup.usrao.mongodb.scheduler  
com.citigroup.usrao.mongodb.service  
com.citigroup.usrao.mongodb.service.ratesheet  
com.citigroup.usrao.mongodb.validator  
src/main/resources  
src/test/java  
JRE System Library [JavaSE-1.8]  
Maven Dependencies  
target/generated-sources/annotations  
target/generated-test-sources/test-annotations  
api-gateway  
api-tests  
app-configs  
app-info  
manifests  
performance-tests  
src  
target  
hotfix\_pcdConfig\_pipeline.yml  
hotfix\_springboot\_pipeline.yml  
master\_pcdConfig\_pipeline.yml  
master\_springboot\_pipeline.yml  
pom.xml

```
2* Copyright (C) 2016 by Citigroup. All rights reserved.
package com.citigroup.usrao.mongodb.configuration;

import org.springframework.boot.context.properties.ConfigurationProperties;

@Configuration
@ConfigurationProperties(prefix = "connection")
public class ConnectionDetails {

    private String username;

    private String password;

    private String certPath;

    private String dbName;

    private String sourceName;

    private String serverAddress;

    private String serverAddress1;

    private String serverAddress2;

    private String port;

    private String sslKeyStore;

    private String sslKeyStorePwd;

    public String getUsername() {
        return username;
    }

    public void setUsername(String username) {
        this.username = username;
    }

    public String getPassword() {
        return password;
    }

    public void setPassword(String password) {
        this.password = password;
    }

    public String getCertPath() {
        return certPath;
    }
}
```

Markers | Properties | Servers | Snippets | Console | Search | JUnit | Coverage | History | Call Hierarchy | Boot Dashboard | Debug

@EnableAspectAutoProxy - 1 match in workspace ("java")

am-d-servicengorder [boot] [am-d-servicengorder master]  
src  
main

Project Explorer | Navigator (Deprecated) | OriginConfiguration.java | MongoDBConfiguration.java | ConnectionDetails.java

170041\_usrao master  
am-d-servicengorder [boot] [am-d-servicengorder master]  
usretailapi.checkbook.component [in rti-pdm-d-checkbook] [com.citi.171384.usretailapi.checkbook.component master]  
usrao-utility\_repo feature/URS-584  
fdn-c-eodjobs [boot] [fdn-c-eodjobs bug/eoddefect]  
usrao-utility\_repo feature/URS-584  
mongodb feature/AugScriptUpdate  
rdingmongodb-usretail [boot] [ogn-c-onboardingmongodb-usretail master]  
java  
group.usrao.mongodb  
group.usrao.mongodb.config  
Configuration.java  
group.usrao.mongodb.configuration  
CommonConfigurations.java  
ConnectionDetails.java  
MongoDbConfiguration.java  
SwaggerConfiguration.java  
SwaggerController.java  
group.usrao.mongodb.controller  
MongoDBController.java  
MongoDBController  
group.usrao.mongodb.domain  
group.usrao.mongodb.event  
group.usrao.mongodb.exception  
group.usrao.mongodb.model  
group.usrao.mongodb.resttemplate  
group.usrao.mongodb.scheduler  
group.usrao.mongodb.service  
group.usrao.mongodb.service.ratesheet  
group.usrao.mongodb.validator  
resources  
ra  
Library [JavaSE-1.8]  
dependencies  
generated-sources/annotations  
generated-test-sources/test-annotations  
ry  
gs  
nce-tests  
Config\_pipeline.yml  
ingBoot\_pipeline.yml  
dConfig\_pipeline.yml  
pringBoot\_pipeline.yml

```
import java.io.File;

@Configuration
public class MongoDBConfiguration {

    private static final Logger LOG = LoggerFactory.getLogger(MongoDbConfiguration.class.getName());

    @Autowired
    CertificateConfig certificateConfig;

    @Autowired
    ConnectionDetails mongoConfig;

    @Bean
    public MongoClientDatabaseFactory mongoDatabaseFactory() {
        MongoCredential credential = MongoCredential.createScramSha256Credential(mongoConfig.getUsername(),
            mongoConfig.getSourceName(), mongoConfig.getPassword().toCharArray());
        MongoClientSettings settings = MongoClientSettings.builder()
            .applyToClusterSettings(builder -> builder.hosts(Arrays.asList(
                new ServerAddress(mongoConfig.getServerAddress(), Integer.parseInt(mongoConfig.getPort())),
                new ServerAddress(mongoConfig.getServerAddress1(), Integer.parseInt(mongoConfig.getPort())),
                new ServerAddress(mongoConfig.getServerAddress2(), Integer.parseInt(mongoConfig.getPort()))))
            .credential(credential).applyToSslSettings(builder -> {
                builder.enabled(true);
                try {
                    builder.context(getSSLContext());
                } catch (KeyManagementException | UnrecoverableKeyException | KeyStoreException |
                    NoSuchAlgorithmException | CertificateException | IOException e) {
                    LOG.error("Exception occurred {}", e);
                    LOG.debug("{}{} ", "Error in ", e.getStackTrace());
                }
            }).build();
        return new SimpleMongoClientDatabaseFactory(MongoClients.create(settings), mongoConfig.getDbName());
    }

    private SSLContext getSSLContext() throws KeyStoreException, NoSuchAlgorithmException, CertificateException,
        IOException, KeyManagementException, UnrecoverableKeyException {
        KeyStore keyStore = KeyStore.getInstance(KeyStore.getDefaultType());
        if (StringUtil.isEmpty(mongoConfig.getCertPath())) {
            File certFile = certificateConfig.getCertificate(mongoConfig.getCertPath(), mongoConfig.getSslKeyStore());
            try (FileInputStream certificate = new FileInputStream(certFile)) {
                keyStore.load(certificate, mongoConfig.getSslKeyStorePwd().toCharArray());
            } catch (Exception e) {
                LOG.error("Exception occurred {}", e);
                LOG.debug("{}{} ", "Error in ", e.getStackTrace());
            }
        }
    }
}
```

Markers | Properties | Servers | Snippets | Console | Search | JUnit | Coverage | History | Call Hierarchy | Boot Dashboard | Debug

@EnableAspectAutoProxy - 1 match in workspace ("java")

am-d-servicengorder [boot] [am-d-servicengorder master]  
src  
main



```
ogn-c-onboardingmongodb-usretail/src/main/java/com/citigroup/usrao/mongodb/configuration/MongoDbConfiguration.java - Spring Tool Suite 4
Refactor Navigate Search Project Run Window Help
Navigator (Deprecated)
rao [170041.usrao master]
chngorder [boot] [am-d-servicingorder master]
1384.usretailapi.checkbook.component [in rti-pdm-d-check]
ty [usrao-utility_repo feature/URS-584]
obs [boot] [fdn-c-eodjobs bug/eoddefect]
rao-utility_repo feature/URS-584]
[mongodb feature/AugScriptUpdate]
ordingmongodb-usretail [boot] [ogn-c-onboardingmong]
n/java
citigroup.usrao.mongodb
itigroup.usrao.mongodb.config
in.Configuration.java
citigroup.usrao.mongodb.configuration
rmon.Configurations.java
nectionDetails.java
MongoDbConfiguration.java
gger.Configuration.java
gger.Controller.java
itigroup.usrao.mongodb.controller
ngoDbController.java
MongoDbController
itigroup.usrao.mongodb.domain
itigroup.usrao.mongodb.event
itigroup.usrao.mongodb.exception
itigroup.usrao.mongodb.model
itigroup.usrao.mongodb.resttemplate
itigroup.usrao.mongodb.scheduler
itigroup.usrao.mongodb.service
itigroup.usrao.mongodb.service.ratesheet
itigroup.usrao.mongodb.validator
/resources
java
m Library [JavaSE-1.8]
dependencies
enerated-sources/annotations
enerated-test-sources/test-annotations
way
s
figs
s
ance-tests
xc/Config_pipeline.yml
springBoot_pipeline.yml
pc/Config_pipeline.yml
springBoot_pipeline.yml

OriginConfiguration.java
MongoDbConfiguration.java
ConnectionDetails.java
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
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000

private SSLContext getSSLContext() throws KeyStoreException, NoSuchAlgorithmException, CertificateException,
IOException, KeyManagementException, UnrecoverableKeyException {
    KeyStore keyStore = KeyStore.getInstance(KeyStore.getDefaultType());

    if (StringUtils.isEmpty(mongoConfig.getCertPath()) {
        File certFile = certificateConfig.getCertificate(mongoConfig.getCertPath(), mongoConfig.getSslKeyStore());
        try (FileInputStream certificate = new FileInputStream(certFile)) {
            keyStore.load(certificate, mongoConfig.getSslKeyStorePwd().toCharArray());
        } catch (Exception e) {
            LOGGER.error("Exception occurred {}", e);
            LOGGER.debug("{} ", "Error in ", e.getStackTrace());
        }
    } else {
        try (FileInputStream certificate = new FileInputStream(mongoConfig.getSslKeyStore())) {
            keyStore.load(certificate, mongoConfig.getSslKeyStorePwd().toCharArray());
        } catch (Exception e) {
            LOGGER.error("Exception occurred {}", e);
            LOGGER.debug("{} ", "Error in ", e.getStackTrace());
        }
    }

    KeyManagerFactory keyManagerFactory = KeyManagerFactory.getInstance("SunX509");
    keyManagerFactory.init(keyStore, mongoConfig.getSslKeyStorePwd().toCharArray());
    KeyManager[] keyManagers = keyManagerFactory.getKeyManagers();

    TrustManagerFactory trustManagerFactory = TrustManagerFactory.getInstance("SunX509");
    trustManagerFactory.init(keyStore);
    TrustManager[] trustManagers = trustManagerFactory.getTrustManagers();

    SSLContext sslContext = SSLContext.getInstance("TLSv1.2");
    sslContext.init(keyManagers, trustManagers, null);

    return sslContext;
}

@Bean
public MongoTemplate mongoTemplate() {
    return new MongoTemplate(mongoDatabaseFactory());
}

Markers Properties Servers Snippets Console Search Unit Coverage History Call Hierarchy Boot Dashboard Debug
@EnableAspect(AutoProxy - 1 match in workspace (*.java))
44 - am-d-servicingorder [boot] [am-d-servicingorder master]
```

```
STSWorkspace - ogn-c-onboardingmongodb-usretail/src/main/java/com/citigroup/usrao/mongodb/controller/MongoDBController.java - Spring Tool Suite 4
Edit Source Refactor Navigate Search Project Run Window Help
Project Explorer
170041.usrao [170041.usrao master]
am-d-servicingorder [boot] [am-d-servicingorder master]
com.citigroup.usrao.usretailapi.checkbook.component [in rti-pdm-d-check]
CommonUtility [usrao-utility_repo feature/URS-584]
fdn-c-eodjobs [boot] [fdn-c-eodjobs bug/eoddefect]
GoUtility [usrao-utility_repo feature/URS-584]
mongodb [mongodb feature/AugScriptUpdate]
ogn-c-onboardingmongodb-usretail [boot] [ogn-c-onboardingmong]
src/main/java
com.citigroup.usrao.mongodb
com.citigroup.usrao.mongodb.config
com.citigroup.usrao.mongodb.configuration
CommonConfigurations.java
ConnectionDetails.java
MongoDbConfiguration.java
SwaggerConfiguration.java
SwaggerController.java
com.citigroup.usrao.mongodb.controller
MongoDbController
com.citigroup.usrao.mongodb.domain
com.citigroup.usrao.mongodb.event
com.citigroup.usrao.mongodb.exception
com.citigroup.usrao.mongodb.model
com.citigroup.usrao.mongodb.scheduler
com.citigroup.usrao.mongodb.service
com.citigroup.usrao.mongodb.service.ratesheet
com.citigroup.usrao.mongodb.validator
src/main/resources
JRE System Library [JavaSE-1.8]
Maven Dependencies
target/generated-sources/annotations
target/generated-test-sources/test-annotations
api-gateway
app-configs
app-info
manifests
performance-tests
src
target
hotfix_pcd/Config_pipeline.yml
hotfix_springBoot_pipeline.yml
master_pcd/Config_pipeline.yml
master_springBoot_pipeline.yml
pom.xml

OriginConfiguration.java
MongoDbConfiguration.java
CommonConfigurations.java
MongoDBController.java
RetrieveCustomerFormService
2** Copyright (C) 2016 by Citigroup. All rights reserved.
15 package com.citigroup.usrao.mongodb.controller;
16
17 import java.text.ParseException;
18
19 @RestController
20 public class MongoDBController {
21
22     @Autowired
23     private MongoTemplate mongoTemplate;
24
25     @Autowired
26     private RetrieveWelcomeKitTransactionService retrieveWelcomeKitTransactionService;
27
28     @Autowired
29     private SaveWelcomeKitTransactionService saveWelcomeKitTransactionService;
30
31     @Autowired
32     private RetrieveCustomerFormService retrieveCustomerFormService;
33
34     @Autowired
35     private RetrieveEnvelopeStatusService envelopeStatusService;
36
37     @Autowired
38     private UpdateEnvelopeStatusService updateEnvelopeStatusService;
39
40     @Autowired
41     private SaveEnvelopeStatusService saveEnvelopeStatusService;
42
43     @Autowired
44     private SecuredDisclosureDownloadScheduler securedDisclosureDownloadScheduler;
45
46     @Autowired
47     private DisclosureDownloadScheduler disclosureDownloadScheduler;
48
49     private static final String RESPONSE_MSG = "Success";
50     private static final String SELECT_FIELD = "applicationId";
51
52     @Autowired
53     private EmployeeDetailsService employeeDetailsService;
54
55     @Autowired
56     private EmployeeDetailsCRUDService employeeDetailsCRUDService;
57
58     @Autowired
59     private EventPublisher eventPublisher;
60
61     @ApiOperation(value = "This API is used to save GCIF transactions while Account opening", notes = "Transactions will be saved in Mong")
62     @ApiResponses(value = { @ApiResponse(code = 200, message = "Successful response", response = String.class),
63         @ApiResponse(code = 201, message = "Application NOT saved in to DB<br>success</td><td>accountSavedNotOpened</td><td>Account could not b")
64     })
65     public String saveTransaction(String transactionDetails) {
66         // TODO: Implement saveTransaction method
67         return null;
68     }
69
70     // Other methods...
71
72     private void logSuccess(String message) {
73         LOGGER.info(message);
74     }
75
76     private void logError(String message, Exception e) {
77         LOGGER.error(message, e);
78     }
79
80     private void logWarning(String message) {
81         LOGGER.warn(message);
82     }
83
84     private void logDebug(String message) {
85         LOGGER.debug(message);
86     }
87
88     private void logInfo(String message) {
89         LOGGER.info(message);
90     }
91
92     private void logTrace(String message) {
93         LOGGER.trace(message);
94     }
95
96     private void logFatal(String message) {
97         LOGGER.fatal(message);
98     }
99
100     private void logWarn(String message) {
101         LOGGER.warn(message);
102     }
103
104     private void logError(String message) {
105         LOGGER.error(message);
106     }
107
108     private void logInfo(String message) {
109         LOGGER.info(message);
110     }
111
112     private void logTrace(String message) {
113         LOGGER.trace(message);
114     }
115
116     private void logFatal(String message) {
117         LOGGER.fatal(message);
118     }
119
120     private void logWarn(String message) {
121         LOGGER.warn(message);
122     }
123
124     private void logError(String message) {
125         LOGGER.error(message);
126     }
127
128     private void logInfo(String message) {
129         LOGGER.info(message);
130     }
131
132     private void logTrace(String message) {
133         LOGGER.trace(message);
134     }
135
136     private void logFatal(String message) {
137         LOGGER.fatal(message);
138     }
139
140     private void logWarn(String message) {
141         LOGGER.warn(message);
142     }
143
144     private void logError(String message) {
145         LOGGER.error(message);
146     }
147
148     private void logInfo(String message) {
149         LOGGER.info(message);
150     }
151
152     private void logTrace(String message) {
153         LOGGER.trace(message);
154     }
155
156     private void logFatal(String message) {
157         LOGGER.fatal(message);
158     }
159
160     private void logWarn(String message) {
161         LOGGER.warn(message);
162     }
163
164     private void logError(String message) {
165         LOGGER.error(message);
166     }
167
168     private void logInfo(String message) {
169         LOGGER.info(message);
170     }
171
172     private void logTrace(String message) {
173         LOGGER.trace(message);
174     }
175
176     private void logFatal(String message) {
177         LOGGER.fatal(message);
178     }
179
180     private void logWarn(String message) {
181         LOGGER.warn(message);
182     }
183
184     private void logError(String message) {
185         LOGGER.error(message);
186     }
187
188     private void logInfo(String message) {
189         LOGGER.info(message);
190     }
191
192     private void logTrace(String message) {
193         LOGGER.trace(message);
194     }
195
196     private void logFatal(String message) {
197         LOGGER.fatal(message);
198     }
199
200     private void logWarn(String message) {
201         LOGGER.warn(message);
202     }
203
204     private void logError(String message) {
205         LOGGER.error(message);
206     }
207
208     private void logInfo(String message) {
209         LOGGER.info(message);
210     }
211
212     private void logTrace(String message) {
213         LOGGER.trace(message);
214     }
215
216     private void logFatal(String message) {
217         LOGGER.fatal(message);
218     }
219
220     private void logWarn(String message) {
221         LOGGER.warn(message);
222     }
223
224     private void logError(String message) {
225         LOGGER.error(message);
226     }
227
228     private void logInfo(String message) {
229         LOGGER.info(message);
230     }
231
232     private void logTrace(String message) {
233         LOGGER.trace(message);
234     }
235
236     private void logFatal(String message) {
237         LOGGER.fatal(message);
238     }
239
240     private void logWarn(String message) {
241         LOGGER.warn(message);
242     }
243
244     private void logError(String message) {
245         LOGGER.error(message);
246     }
247
248     private void logInfo(String message) {
249         LOGGER.info(message);
250     }
251
252     private void logTrace(String message) {
253         LOGGER.trace(message);
254     }
255
256     private void logFatal(String message) {
257         LOGGER.fatal(message);
258     }
259
260     private void logWarn(String message) {
261         LOGGER.warn(message);
262     }
263
264     private void logError(String message) {
265         LOGGER.error(message);
266     }
267
268     private void logInfo(String message) {
269         LOGGER.info(message);
270     }
271
272     private void logTrace(String message) {
273         LOGGER.trace(message);
274     }
275
276     private void logFatal(String message) {
277         LOGGER.fatal(message);
278     }
279
280     private void logWarn(String message) {
281         LOGGER.warn(message);
282     }
283
284     private void logError(String message) {
285         LOGGER.error(message);
286     }
287
288     private void logInfo(String message) {
289         LOGGER.info(message);
290     }
291
292     private void logTrace(String message) {
293         LOGGER.trace(message);
294     }
295
296     private void logFatal(String message) {
297         LOGGER.fatal(message);
298     }
299
300     private void logWarn(String message) {
301         LOGGER.warn(message);
302     }
303
304     private void logError(String message) {
305         LOGGER.error(message);
306     }
307
308     private void logInfo(String message) {
309         LOGGER.info(message);
310     }
311
312     private void logTrace(String message) {
313         LOGGER.trace(message);
314     }
315
316     private void logFatal(String message) {
317         LOGGER.fatal(message);
318     }
319
320     private void logWarn(String message) {
321         LOGGER.warn(message);
322     }
323
324     private void logError(String message) {
325         LOGGER.error(message);
326     }
327
328     private void logInfo(String message) {
329         LOGGER.info(message);
330     }
331
332     private void logTrace(String message) {
333         LOGGER.trace(message);
334     }
335
336     private void logFatal(String message) {
337         LOGGER.fatal(message);
338     }
339
340     private void logWarn(String message) {
341         LOGGER.warn(message);
342     }
343
344     private void logError(String message) {
345         LOGGER.error(message);
346     }
347
348     private void logInfo(String message) {
349         LOGGER.info(message);
350     }
351
352     private void logTrace(String message) {
353         LOGGER.trace(message);
354     }
355
356     private void logFatal(String message) {
357         LOGGER.fatal(message);
358     }
359
360     private void logWarn(String message) {
361         LOGGER.warn(message);
362     }
363
364     private void logError(String message) {
365         LOGGER.error(message);
366     }
367
368     private void logInfo(String message) {
369         LOGGER.info(message);
370     }
371
372     private void logTrace(String message) {
373         LOGGER.trace(message);
374     }
375
376     private void logFatal(String message) {
377         LOGGER.fatal(message);
378     }
379
380     private void logWarn(String message) {
381         LOGGER.warn(message);
382     }
383
384     private void logError(String message) {
385         LOGGER.error(message);
386     }
387
388     private void logInfo(String message) {
389         LOGGER.info(message);
390     }
391
392     private void logTrace(String message) {
393         LOGGER.trace(message);
394     }
395
396     private void logFatal(String message) {
397         LOGGER.fatal(message);
398     }
399
400     private void logWarn(String message) {
401         LOGGER.warn(message);
402     }
403
404     private void logError(String message) {
405         LOGGER.error(message);
406     }
407
408     private void logInfo(String message) {
409         LOGGER.info(message);
410     }
411
412     private void logTrace(String message) {
413         LOGGER.trace(message);
414     }
415
416     private void logFatal(String message) {
417         LOGGER.fatal(message);
418     }
419
420     private void logWarn(String message) {
421         LOGGER.warn(message);
422     }
423
424     private void logError(String message) {
425         LOGGER.error(message);
426     }
427
428     private void logInfo(String message) {
429         LOGGER.info(message);
430     }
431
432     private void logTrace(String message) {
433         LOGGER.trace(message);
434     }
435
436     private void logFatal(String message) {
437         LOGGER.fatal(message);
438     }
439
440     private void logWarn(String message) {
441         LOGGER.warn(message);
442     }
443
444     private void logError(String message) {
445         LOGGER.error(message);
446     }
447
448     private void logInfo(String message) {
449         LOGGER.info(message);
450     }
451
452     private void logTrace(String message) {
453         LOGGER.trace(message);
454     }
455
456     private void logFatal(String message) {
457         LOGGER.fatal(message);
458     }
459
460     private void logWarn(String message) {
461         LOGGER.warn(message);
462     }
463
464     private void logError(String message) {
465         LOGGER.error(message);
466     }
467
468     private void logInfo(String message) {
469         LOGGER.info(message);
470     }
471
472     private void logTrace(String message) {
473         LOGGER.trace(message);
474     }
475
476     private void logFatal(String message) {
477         LOGGER.fatal(message);
478     }
479
480     private void logWarn(String message) {
481         LOGGER.warn(message);
482     }
483
484     private void logError(String message) {
485         LOGGER.error(message);
486     }
487
488     private void logInfo(String message) {
489         LOGGER.info(message);
490     }
491
492     private void logTrace(String message) {
493         LOGGER.trace(message);
494     }
495
496     private void logFatal(String message) {
497         LOGGER.fatal(message);
498     }
499
500     private void logWarn(String message) {
501         LOGGER.warn(message);
502     }
503
504     private void logError(String message) {
505         LOGGER.error(message);
506     }
507
508     private void logInfo(String message) {
509         LOGGER.info(message);
510     }
511
512     private void logTrace(String message) {
513         LOGGER.trace(message);
514     }
515
516     private void logFatal(String message) {
517         LOGGER.fatal(message);
518     }
519
520     private void logWarn(String message) {
521         LOGGER.warn(message);
522     }
523
524     private void logError(String message) {
525         LOGGER.error(message);
526     }
527
528     private void logInfo(String message) {
529         LOGGER.info(message);
530     }
531
532     private void logTrace(String message) {
533         LOGGER.trace(message);
534     }
535
536     private void logFatal(String message) {
537         LOGGER.fatal(message);
538     }
539
540     private void logWarn(String message) {
541         LOGGER.warn(message);
542     }
543
544     private void logError(String message) {
545         LOGGER.error(message);
546     }
547
548     private void logInfo(String message) {
549         LOGGER.info(message);
550     }
551
552     private void logTrace(String message) {
553         LOGGER.trace(message);
554     }
555
556     private void logFatal(String message) {
557         LOGGER.fatal(message);
558     }
559
560     private void logWarn(String message) {
561         LOGGER.warn(message);
562     }
563
564     private void logError(String message) {
565         LOGGER.error(message);
566     }
567
568     private void logInfo(String message) {
569         LOGGER.info(message);
570     }
571
572     private void logTrace(String message) {
573         LOGGER.trace(message);
574     }
575
576     private void logFatal(String message) {
577         LOGGER.fatal(message);
578     }
579
580     private void logWarn(String message) {
581         LOGGER.warn(message);
582     }
583
584     private void logError(String message) {
585         LOGGER.error(message);
586     }
587
588     private void logInfo(String message) {
589         LOGGER.info(message);
590     }
591
592     private void logTrace(String message) {
593         LOGGER.trace(message);
594     }
595
596     private void logFatal(String message) {
597         LOGGER.fatal(message);
598     }
599
600     private void logWarn(String message) {
601         LOGGER.warn(message);
602     }
603
604     private void logError(String message) {
605         LOGGER.error(message);
606     }
607
608     private void logInfo(String message) {
609         LOGGER.info(message);
610     }
611
612     private void logTrace(String message) {
613         LOGGER.trace(message);
614     }
615
616     private void logFatal(String message) {
617         LOGGER.fatal(message);
618     }
619
620     private void logWarn(String message) {
621         LOGGER.warn(message);
622     }
623
624     private void logError(String message) {
625         LOGGER.error(message);
626     }
627
628     private void logInfo(String message) {
629         LOGGER.info(message);
630     }
631
632     private void logTrace(String message) {
633         LOGGER.trace(message);
634     }
635
636     private void logFatal(String message) {
637         LOGGER.fatal(message);
638     }
639
640     private void logWarn(String message) {
641         LOGGER.warn(message);
642     }
643
644     private void logError(String message) {
645         LOGGER.error(message);
646     }
647
648     private void logInfo(String message) {
649         LOGGER.info(message);
650     }
651
652     private void logTrace(String message) {
653         LOGGER.trace(message);
654     }
655
656     private void logFatal(String message) {
657         LOGGER.fatal(message);
658     }
659
660     private void logWarn(String message) {
661         LOGGER.warn(message);
662     }
663
664     private void logError(String message) {
665         LOGGER.error(message);
666     }
667
668     private void logInfo(String message) {
669         LOGGER.info(message);
670     }
671
672     private void logTrace(String message) {
673         LOGGER.trace(message);
674     }
675
676     private void logFatal(String message) {
677         LOGGER.fatal(message);
678     }
679
680     private void logWarn(String message) {
681         LOGGER.warn(message);
682     }
683
684     private void logError(String message) {
685         LOGGER.error(message);
686     }
687
688     private void logInfo(String message) {
689         LOGGER.info(message);
690     }
691
692     private void logTrace(String message) {
693         LOGGER.trace(message);
694     }
695
696     private void logFatal(String message) {
697         LOGGER.fatal(message);
698     }
699
700     private void logWarn(String message) {
701         LOGGER.warn(message);
702     }
703
704     private void logError(String message) {
705         LOGGER.error(message);
706     }
707
708     private void logInfo(String message) {
709         LOGGER.info(message);
710     }
711
712     private void logTrace(String message) {
713         LOGGER.trace(message);
714     }
715
716     private void logFatal(String message) {
717         LOGGER.fatal(message);
718     }
719
720     private void logWarn(String message) {
721         LOGGER.warn(message);
722     }
723
724     private void logError(String message) {
725         LOGGER.error(message);
726     }
727
728     private void logInfo(String message) {
729         LOGGER.info(message);
730     }
731
732     private void logTrace(String message) {
733         LOGGER.trace(message);
734     }
735
736     private void logFatal(String message) {
737         LOGGER.fatal(message);
738     }
739
740     private void logWarn(String message) {
741         LOGGER.warn(message);
742     }
743
744     private void logError(String message) {
745         LOGGER.error(message);
746     }
747
748     private void logInfo(String message) {
749         LOGGER.info(message);
750     }
751
752     private void logTrace(String message) {
753         LOGGER.trace(message);
754     }
755
756     private void logFatal(String message) {
757         LOGGER.fatal(message);
758     }
759
760     private void logWarn(String message) {
761         LOGGER.warn(message);
762     }
763
764     private void logError(String message) {
765         LOGGER.error(message);
766     }
767
768     private void logInfo(String message) {
769         LOGGER.info(message);
770     }
771
772     private void logTrace(String message) {
773         LOGGER.trace(message);
774     }
775
776     private void logFatal(String message) {
777         LOGGER.fatal(message);
778     }
779
780     private void logWarn(String message) {
781         LOGGER.warn(message);
782     }
783
784     private void logError(String message) {
785         LOGGER.error(message);
786     }
787
788     private void logInfo(String message) {
789         LOGGER.info(message);
790     }
791
792     private void logTrace(String message) {
793         LOGGER.trace(message);
794     }
795
796     private void logFatal(String message) {
797         LOGGER.fatal(message);
798     }
799
800     private void logWarn(String message) {
801         LOGGER.warn(message);
802     }
803
804     private void logError(String message) {
805         LOGGER.error(message);
806     }
807
808     private void logInfo(String message) {
809         LOGGER.info(message);
810     }
811
812     private void logTrace(String message) {
813         LOGGER.trace(message);
814     }
815
816     private void logFatal(String message) {
817         LOGGER.fatal(message);
818     }
819
820     private void logWarn(String message) {
821         LOGGER.warn(message);
822     }
823
824     private void logError(String message) {
825         LOGGER.error(message);
826     }
827
828     private void logInfo(String message) {
829         LOGGER.info(message);
830     }
831
832     private void logTrace(String message) {
833         LOGGER.trace(message);
834     }
835
836     private void logFatal(String message) {
837         LOGGER.fatal(message);
838     }
839
840     private void logWarn(String message) {
841         LOGGER.warn(message);
842     }
843
844     private void logError(String message) {
845         LOGGER.error(message);
846     }
847
848     private void logInfo(String message) {
849         LOGGER.info(message);
850     }
851
852     private void logTrace(String message) {
853         LOGGER.trace(message);
854     }
855
856     private void logFatal(String message) {
857         LOGGER.fatal(message);
858     }
859
860     private void logWarn(String message) {
861         LOGGER.warn(message);
862     }
863
864     private void logError(String message) {
865         LOGGER.error(message);
866     }
867
868     private void logInfo(String message) {
869         LOGGER.info(message);
870     }
871
872     private void logTrace(String message) {
873         LOGGER.trace(message);
874     }
875
876     private void logFatal(String message) {
877         LOGGER.fatal(message);
878     }
879
880     private void logWarn(String message) {
881         LOGGER.warn(message);
882     }
883
884     private void logError(String message) {
885         LOGGER.error(message);
886     }
887
888     private void logInfo(String message) {
889         LOGGER.info(message);
890     }
891
892     private void logTrace(String message) {
893         LOGGER.trace(message);
894     }
895
896     private void logFatal(String message) {
897         LOGGER.fatal(message);
898     }
899
900     private void logWarn(String message) {
901         LOGGER.warn(message);
902     }
903
904     private void logError(String message) {
905         LOGGER.error(message);
906     }
907
908     private void logInfo(String message) {
909         LOGGER.info(message);
909     }
910
911     private void logTrace(String message) {
912         LOGGER.trace(message);
913     }
914
915     private void logFatal(String message) {
916         LOGGER.fatal(message);
917     }
918
919     private void logWarn(String message) {
920         LOGGER.warn(message);
921     }
922
923     private void logError(String message) {
924         LOGGER.error(message);
925     }
926
927     private void logInfo(String message) {
928         LOGGER.info(message);
929     }
930
931     private void logTrace(String message) {
932         LOGGER.trace(message);
933     }
934
935     private void logFatal(String message) {
```















```
spring-mongo-usretail/src/main/java/com/citigroup/usrao/mongodb/resttemplate/GlobalDirectoryServiceRestTemplate.java - Spring Tool Suite 4
Edit Search Project Run Window Help
MongoDBConf... ConnectionD... CommonConfig... MongoDBCon... RetrieveCus... CustomerForm... RetrieveWel... SaveEnvelop... EmployeeDet... RestConnect... Employee
64 employeeDetails.getEmailAddress().endsWith("@uoc.citi.com")) {
65     throw new EmployeeDetailsException(employeeDetails, soeId);
66 }
67 return employeeDetails;
68 }
69
70 @HystrixCommand(commandKey = "hystrixCommandGetEmployee", threadPoolKey = "hystrixThreadPoolGetEmployee", fallbackMethod = "getEmployeeDetailsFallback")
71 public GetEmployeeDetailsResponseDTO getEmployeeDetailsResponse(String uuid, String clientId, String businessCode,
72     String countryCode, String channelId, GetEmployeeDetailsRequestDTO getEmployeeDetailsRequest)
73     throws ServiceNotFoundException {
74     GetEmployeeDetailsResponseDTO response = null;
75
76     HttpEntity<GetEmployeeDetailsRequestDTO> requestEntity = new HttpEntity<>(getEmployeeDetailsRequest,
77         getHttpHeaders(uuid, businessCode, countryCode, channelId, clientId));
78     Logger.debug("Request Entity: {} ", requestEntity);
79     URI serviceURI = null;
80     serviceURI = connectionUtil.getURI(employeeService, "/private/v1/foundations/directory/employee/retrieve");
81     Logger.info("EmployeeRestTemplate URI: {} ", serviceURI);
82     if (serviceURI != null) {
83         ResponseEntity<GetEmployeeDetailsResponseDTO> result = restTemplate.exchange(serviceURI.toString(),
84             HttpMethod.POST, requestEntity, GetEmployeeDetailsResponseDTO.class);
85         response = result.getBody();
86         Logger.debug("Response Entity: {} ", response);
87     }
88     return response;
89 }
90
91 public GetEmployeeDetailsResponseDTO getEmployeeDetailsFallback(String uuid, String clientId, String businessCode,
92     String countryCode, String channelId, GetEmployeeDetailsRequestDTO getEmployeeDetailsRequest) {
93     GetEmployeeDetailsResponseDTO response = new GetEmployeeDetailsResponseDTO();
94     Logger.error("Hystrix circuit breaker has tripped in GlobalDirectoryServiceRestTemplate.getEmployeeDetails()");
95     return response;
96 }
97
98 public static HttpHeaders getHttpHeaders(String uuid, String businessCode, String countryCode, String channelId,
99     String clientId) {
100     MultiValueMap<String, String> args = new LinkedMultiValueMap<>();
101     args.add("uuid", uuid);
102     args.add("businessCode", businessCode);
103     args.add("CountryCode", countryCode);
104     args.add("channelId", channelId);
105     args.add("client_id", clientId);
106     args.add("Accept", "application/json");
107     args.add("Content-Type", "application/json");
108     return new HttpHeaders(args);
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 }
926
927 }
928
929 }
929
930 }
931
932 }
933
934 }
935
936 }
937
938 }
939
940 }
941
942 }
943
944 }
945
946 }
947
948 }
949
950 }
951
952 }
953
954 }
955
956 }
957
958 }
959
960 }
961
962 }
963
964 }
965
966 }
967
968 }
969
970 }
971
972 }
973
974 }
975
976 }
977
978 }
979
980 }
981
982 }
983
984 }
985
986 }
987
988 }
989
990 }
991
992 }
993
994 }
995
996 }
997
998 }
999
1000 }
```

```
STSWorkspace - ogn-c-onboardingmongodb-usretail/app-configs/GT1DMSDEV3/OGN-C-OnboardingMongoDB-UsRetail/OGN-C-OnboardingMongoDB-UsRetail.yml - Spring Tool Suite 4
Edit Search Project Run Window Help
MongoDBConf... ConnectionD... CommonConfig... MongoDBCon... RetrieveCus... CustomerForm... RetrieveWel... SaveEnvelop... EmployeeDet...
Project Explorer | Navigator (Deprecated)
com.citigroup.usrao.mongodb.resttemplate
  GlobalDirectoryServiceRestTemplate.java
  RestConnectionUtil.java
com.citigroup.usrao.mongodb.scheduler
  com.citigroup.usrao.mongodb.scheduler
com.citigroup.usrao.mongodb.service
  EmployeeDetailsCRUDService.java
  EmployeeDetailsService.java
  RetrieveCustomerFormService.java
  RetrieveEnvelopeStatusService.java
  RetrieveWelcomeKitTransactionService.java
  SaveEnvelopeStatusService.java
  SaveWelcomeKitTransactionService.java
  UpdateEnvelopeStatusService.java
com.citigroup.usrao.mongodb.service.ratesheet
  com.citigroup.usrao.mongodb.service.ratesheet
com.citigroup.usrao.mongodb.validator
  src/main/resources
  src/test/java
  JRE System Library [JavaSE-1.8]
  Maven Dependencies
  target/generated-sources/annotations
  target/generated-test-sources/test-annotations
  api-gateway
  api-tests
  app-configs
    GT1DMSDEV1
    GT1DMSDEV3
      OGN-C-OnboardingMongoDB-UsRetail
        certs
        OGN-C-OnboardingMongoDB-UsRetail.yml
      GT1DMSPERF
      GT1DMSPROD
      GT1DMSUAT1
      GT1DMSUAT2
      GT1DMSUAT3
      GT1DMSUAT4
      GT1DMSUAT5
      GT1DMSUAT6
      GT1DMSUAT7
      GT1DMSUAT8
      GT1DMSUAT9
      GT1DMSUAT10
      GT1DMSUAT11
      GT1DMSUAT12
      GT1DMSUAT13
      GT1DMSUAT14
      GT1DMSUAT15
      GT1DMSUAT16
      GT1DMSUAT17
      GT1DMSUAT18
      GT1DMSUAT19
      GT1DMSUAT20
      GT1DMSUAT21
      GT1DMSUAT22
      GT1DMSUAT23
      GT1DMSUAT24
      GT1DMSUAT25
      GT1DMSUAT26
      GT1DMSUAT27
      GT1DMSUAT28
      GT1DMSUAT29
      GT1DMSUAT30
      GT1DMSUAT31
      GT1DMSUAT32
      GT1DMSUAT33
      GT1DMSUAT34
      GT1DMSUAT35
      GT1DMSUAT36
      GT1DMSUAT37
      GT1DMSUAT38
      GT1DMSUAT39
      GT1DMSUAT40
      GT1DMSUAT41
      GT1DMSUAT42
      GT1DMSUAT43
      GT1DMSUAT44
      GT1DMSUAT45
      GT1DMSUAT46
      GT1DMSUAT47
      GT1DMSUAT48
      GT1DMSUAT49
      GT1DMSUAT50
      GT1DMSUAT51
      GT1DMSUAT52
      GT1DMSUAT53
      GT1DMSUAT54
      GT1DMSUAT55
      GT1DMSUAT56
      GT1DMSUAT57
      GT1DMSUAT58
      GT1DMSUAT59
      GT1DMSUAT60
      GT1DMSUAT61
      GT1DMSUAT62
      GT1DMSUAT63
      GT1DMSUAT64
      GT1DMSUAT65
      GT1DMSUAT66
      GT1DMSUAT67
      GT1DMSUAT68
      GT1DMSUAT69
      GT1DMSUAT70
      GT1DMSUAT71
      GT1DMSUAT72
      GT1DMSUAT73
      GT1DMSUAT74
      GT1DMSUAT75
      GT1DMSUAT76
      GT1DMSUAT77
      GT1DMSUAT78
      GT1DMSUAT79
      GT1DMSUAT80
      GT1DMSUAT81
      GT1DMSUAT82
      GT1DMSUAT83
      GT1DMSUAT84
      GT1DMSUAT85
      GT1DMSUAT86
      GT1DMSUAT87
      GT1DMSUAT88
      GT1DMSUAT89
      GT1DMSUAT90
      GT1DMSUAT91
      GT1DMSUAT92
      GT1DMSUAT93
      GT1DMSUAT94
      GT1DMSUAT95
      GT1DMSUAT96
      GT1DMSUAT97
      GT1DMSUAT98
      GT1DMSUAT99
      GT1DMSUAT100
      GT1DMSUAT101
      GT1DMSUAT102
      GT1DMSUAT103
      GT1DMSUAT104
      GT1DMSUAT105
      GT1DMSUAT106
      GT1DMSUAT107
      GT1DMSUAT108
      GT1DMSUAT109
      GT1DMSUAT110
      GT1DMSUAT111
      GT1DMSUAT112
      GT1DMSUAT113
      GT1DMSUAT114
      GT1DMSUAT115
      GT1DMSUAT116
      GT1DMSUAT117
      GT1DMSUAT118
      GT1DMSUAT119
      GT1DMSUAT120
      GT1DMSUAT121
      GT1DMSUAT122
      GT1DMSUAT123
      GT1DMSUAT124
      GT1DMSUAT125
      GT1DMSUAT126
      GT1DMSUAT127
      GT1DMSUAT128
      GT1DMSUAT129
      GT1DMSUAT130
      GT1DMSUAT131
      GT1DMSUAT132
      GT1DMSUAT133
      GT1DMSUAT134
      GT1DMSUAT135
      GT1DMSUAT136
      GT1DMSUAT137
      GT1DMSUAT138
      GT1DMSUAT139
      GT1DMSUAT140
      GT1DMSUAT141
      GT1DMSUAT142
      GT1DMSUAT143
      GT1DMSUAT144
      GT1DMSUAT145
      GT1DMSUAT146
      GT1DMSUAT147
      GT1DMSUAT148
      GT1DMSUAT149
      GT1DMSUAT150
      GT1DMSUAT151
      GT1DMSUAT152
      GT1DMSUAT153
      GT1DMSUAT154
      GT1DMSUAT155
      GT1DMSUAT156
      GT1DMSUAT157
      GT1DMSUAT158
      GT1DMSUAT159
      GT1DMSUAT160
      GT1DMSUAT161
      GT1DMSUAT162
      GT1DMSUAT163
      GT1DMSUAT164
      GT1DMSUAT165
      GT1DMSUAT166
      GT1DMSUAT167
      GT1DMSUAT168
      GT1DMSUAT169
      GT1DMSUAT170
      GT1DMSUAT171
      GT1DMSUAT172
      GT1DMSUAT173
      GT1DMSUAT174
      GT1DMSUAT175
      GT1DMSUAT176
      GT1DMSUAT177
      GT1DMSUAT178
      GT1DMSUAT179
      GT1DMSUAT180
      GT1DMSUAT181
      GT1DMSUAT182
      GT1DMSUAT183
      GT1DMSUAT184
      GT1DMSUAT185
      GT1DMSUAT186
      GT1DMSUAT187
      GT1DMSUAT188
      GT1DMSUAT189
      GT1DMSUAT190
      GT1DMSUAT191
      GT1DMSUAT192
      GT1DMSUAT193
      GT1DMSUAT194
      GT1DMSUAT195
      GT1DMSUAT196
      GT1DMSUAT197
      GT1DMSUAT198
      GT1DMSUAT199
      GT1DMSUAT200
      GT1DMSUAT201
      GT1DMSUAT202
      GT1DMSUAT203
      GT1DMSUAT204
      GT1DMSUAT205
      GT1DMSUAT206
      GT1DMSUAT207
      GT1DMSUAT208
      GT1DMSUAT209
      GT1DMSUAT210
      GT1DMSUAT211
      GT1DMSUAT212
      GT1DMSUAT213
      GT1DMSUAT214
      GT1DMSUAT215
      GT1DMSUAT216
      GT1DMSUAT217
      GT1DMSUAT218
      GT1DMSUAT219
      GT1DMSUAT220
      GT1DMSUAT221
      GT1DMSUAT222
      GT1DMSUAT223
      GT1DMSUAT224
      GT1DMSUAT225
      GT1DMSUAT226
      GT1DMSUAT227
      GT1DMSUAT228
      GT1DMSUAT229
      GT1DMSUAT230
      GT1DMSUAT231
      GT1DMSUAT232
      GT1DMSUAT233
      GT1DMSUAT234
      GT1DMSUAT235
      GT1DMSUAT236
      GT1DMSUAT237
      GT1DMSUAT238
      GT1DMSUAT239
      GT1DMSUAT240
      GT1DMSUAT241
      GT1DMSUAT242
      GT1DMSUAT243
      GT1DMSUAT244
      GT1DMSUAT245
      GT1DMSUAT246
      GT1DMSUAT247
      GT1DMSUAT248
      GT1DMSUAT249
      GT1DMSUAT250
      GT1DMSUAT251
      GT1DMSUAT252
      GT1DMSUAT253
      GT1DMSUAT254
      GT1DMSUAT255
      GT1DMSUAT256
      GT1DMSUAT257
      GT1DMSUAT258
      GT1DMSUAT259
      GT1DMSUAT260
      GT1DMSUAT261
      GT1DMSUAT262
      GT1DMSUAT263
      GT1DMSUAT264
      GT1DMSUAT265
      GT1DMSUAT266
      GT1DMSUAT267
      GT1DMSUAT268
      GT1DMSUAT269
      GT1DMSUAT270
      GT1DMSUAT271
      GT1DMSUAT272
      GT1DMSUAT273
      GT1DMSUAT274
      GT1DMSUAT275
      GT1DMSUAT276
      GT1DMSUAT277
      GT1DMSUAT278
      GT1DMSUAT279
      GT1DMSUAT280
      GT1DMSUAT281
      GT1DMSUAT282
      GT1DMSUAT283
      GT1DMSUAT284
      GT1DMSUAT285
      GT1DMSUAT286
      GT1DMSUAT287
      GT1DMSUAT288
      GT1DMSUAT289
      GT1DMSUAT290
      GT1DMSUAT291
      GT1DMSUAT292
      GT1DMSUAT293
      GT1DMSUAT294
      GT1DMSUAT295
      GT1DMSUAT296
      GT1DMSUAT297
      GT1DMSUAT298
      GT1DMSUAT299
      GT1DMSUAT300
      GT1DMSUAT301
      GT1DMSUAT302
      GT1DMSUAT303
      GT1DMSUAT304
      GT1DMSUAT305
      GT1DMSUAT306
      GT1DMSUAT307
      GT1DMSUAT308
      GT1DMSUAT309
      GT1DMSUAT310
      GT1DMSUAT311
      GT1DMSUAT312
      GT1DMSUAT313
      GT1DMSUAT314
      GT1DMSUAT315
      GT1DMSUAT316
      GT1DMSUAT317
      GT1DMSUAT318
      GT1DMSUAT319
      GT1DMSUAT320
      GT1DMSUAT321
      GT1DMSUAT322
      GT1DMSUAT323
      GT1DMSUAT324
      GT1DMSUAT325
      GT1DMSUAT326
      GT1DMSUAT327
      GT1DMSUAT328
      GT1DMSUAT329
      GT1DMSUAT330
      GT1DMSUAT331
      GT1DMSUAT332
      GT1DMSUAT333
      GT1DMSUAT334
      GT1DMSUAT335
      GT1DMSUAT336
      GT1DMSUAT337
      GT1DMSUAT338
      GT1DMSUAT339
      GT1DMSUAT340
      GT1DMSUAT341
      GT1DMSUAT342
      GT1DMSUAT343
      GT1DMSUAT344
      GT1DMSUAT345
      GT1DMSUAT346
      GT1DMSUAT347
      GT1DMSUAT348
      GT1DMSUAT349
      GT1DMSUAT350
      GT1DMSUAT351
      GT1DMSUAT352
      GT1DMSUAT353
      GT1DMSUAT354
      GT1DMSUAT355
      GT1DMSUAT356
      GT1DMSUAT357
      GT1DMSUAT358
      GT1DMSUAT359
      GT1DMSUAT360
      GT1DMSUAT361
      GT1DMSUAT362
      GT1DMSUAT363
      GT1DMSUAT364
      GT1DMSUAT365
      GT1DMSUAT366
      GT1DMSUAT367
      GT1DMSUAT368
      GT1DMSUAT369
      GT1DMSUAT370
      GT1DMSUAT371
      GT1DMSUAT372
      GT1DMSUAT373
      GT1DMSUAT374
      GT1DMSUAT375
      GT1DMSUAT376
      GT1DMSUAT377
      GT1DMSUAT378
      GT1DMSUAT379
      GT1DMSUAT380
      GT1DMSUAT381
      GT1DMSUAT382
      GT1DMSUAT383
      GT1DMSUAT384
      GT1DMSUAT385
      GT1DMSUAT386
      GT1DMSUAT387
      GT1DMSUAT388
      GT1DMSUAT389
      GT1DMSUAT390
      GT1DMSUAT391
      GT1DMSUAT392
      GT1DMSUAT393
      GT1DMSUAT394
      GT1DMSUAT395
      GT1DMSUAT396
      GT1DMSUAT397
      GT1DMSUAT398
      GT1DMSUAT399
      GT1DMSUAT400
      GT1DMSUAT401
      GT1DMSUAT402
      GT1DMSUAT403
      GT1DMSUAT404
      GT1DMSUAT405
      GT1DMSUAT406
      GT1DMSUAT407
      GT1DMSUAT408
      GT1DMSUAT409
      GT1DMSUAT410
      GT1DMSUAT411
      GT1DMSUAT412
      GT1DMSUAT413
      GT1DMSUAT414
      GT1DMSUAT415
      GT1DMSUAT416
      GT1DMSUAT417
      GT1DMSUAT418
      GT1DMSUAT419
      GT1DMSUAT420
      GT1DMSUAT421
      GT1DMSUAT422
      GT1DMSUAT423
      GT1DMSUAT424
      GT1DMSUAT425
      GT1DMSUAT426
      GT1DMSUAT427
      GT1DMSUAT428
      GT1DMSUAT429
      GT1DMSUAT430
      GT1DMSUAT431
      GT1DMSUAT432
      GT1DMSUAT433
      GT1DMSUAT434
      GT1DMSUAT435
      GT1DMSUAT436
      GT1DMSUAT437
      GT1DMSUAT438
      GT1DMSUAT439
      GT1DMSUAT440
      GT1DMSUAT441
      GT1DMSUAT442
      GT1DMSUAT443
      GT1DMSUAT444
      GT1DMSUAT445
      GT1DMSUAT446
      GT1DMSUAT447
      GT1DMSUAT448
      GT1DMSUAT449
      GT1DMSUAT450
      GT1DMSUAT451
      GT1DMSUAT452
      GT1DMSUAT453
      GT1DMSUAT454
      GT1DMSUAT455
      GT1DMSUAT456
      GT1DMSUAT457
      GT1DMSUAT458
      GT1DMSUAT459
      GT1DMSUAT460
      GT1DMSUAT461
      GT1DMSUAT462
      GT1DMSUAT463
      GT1DMSUAT464
      GT1DMSUAT465
      GT1DMSUAT466
      GT1DMSUAT467
      GT1DMSUAT468
      GT1DMSUAT469
      GT1DMSUAT470
      GT1DMSUAT471
      GT1DMSUAT472
      GT1DMSUAT473
      GT1DMSUAT474
      GT1DMSUAT475
      GT1DMSUAT476
      GT1DMSUAT477
      GT1DMSUAT478
      GT1DMSUAT479
      GT1DMSUAT480
      GT1DMSUAT481
      GT1DMSUAT482
      GT1DMSUAT483
      GT1DMSUAT484
      GT1DMSUAT485
      GT1DMSUAT486
      GT1DMSUAT487
      GT1DMSUAT488
      GT1DMSUAT489
      GT1DMSUAT490
      GT1DMSUAT491
      GT1DMSUAT492
      GT1DMSUAT493
      GT1DMSUAT494
      GT1DMSUAT495
      GT1DMSUAT496
      GT1DMSUAT497
      GT1DMSUAT498
      GT1DMSUAT499
      GT1DMSUAT500
      GT1DMSUAT501
      GT1DMSUAT502
      GT1DMSUAT503
      GT1DMSUAT504
      GT1DMSUAT505
      GT1DMSUAT506
      GT1DMSUAT507
      GT1DMSUAT508
      GT1DMSUAT509
      GT1DMSUAT510
      GT1DMSUAT511
      GT1DMSUAT512
      GT1DMSUAT513
      GT1DMSUAT514
      GT1DMSUAT515
      GT1DMSUAT516
      GT1DMSUAT517
      GT1DMSUAT518
      GT1DMSUAT519
      GT1DMSUAT520
      GT1DMSUAT521
      GT1DMSUAT522
      GT1DMSUAT523
      GT1DMSUAT524
      GT1DMSUAT525
      GT1DMSUAT526
      GT1DMSUAT527
      GT1DMSUAT528
      GT1DMSUAT529
      GT1DMSUAT530
      GT1DMSUAT531
      GT1DMSUAT532
      GT1DMSUAT533
      GT1DMSUAT534
      GT1DMSUAT535
      GT1DMSUAT536
      GT1DMSUAT537
      GT1DMSUAT538
      GT1DMSUAT539
      GT1DMSUAT540
      GT1DMSUAT541
      GT1DMSUAT542
      GT1DMSUAT543
      GT1DMSUAT544
      GT1DMSUAT545
      GT1DMSUAT546
      GT1DMSUAT547
      GT1DMSUAT548
      GT1DMSUAT549
      GT1DMSUAT550
      GT1DMSUAT551
      GT1DMSUAT552
      GT1DMSUAT553
      GT1DMSUAT554
      GT1DMSUAT555
      GT1DMSUAT556
      GT1DMSUAT557
      GT1DMSUAT558
      GT1DMSUAT559
      GT1DMSUAT560
      GT1DMSUAT561
      GT1DMSUAT562
      GT1DMSUAT563
      GT1DMSUAT564
      GT1DMSUAT565
      GT1DMSUAT566
      GT1DMSUAT567
      GT1DMSUAT568
      GT1DMSUAT569
      GT1DMSUAT570
      GT1DMSUAT571
      GT1DMSUAT572
      GT1DMSUAT573
      GT1DMSUAT574
      GT1DMSUAT575
      GT1DMSUAT576
      GT1DMSUAT577
      GT1DMSUAT578
      GT1DMSUAT579
      GT1DMSUAT580
      GT1DMSUAT581
      GT1DMSUAT582
      GT1DMSUAT583
      GT1DMSUAT584
      GT1DMSUAT585
      GT1DMSUAT586
      GT1DMSUAT587
      GT1DMSUAT588
      GT1DMSUAT589
      GT1DMSUAT590
      GT1DMSUAT591
      GT1DMSUAT592
      GT1DMSUAT593
      GT1DMSUAT594
      GT1DMSUAT595
      GT1DMSUAT596
      GT1DMSUAT597
      GT1DMSUAT598
      GT1DMSUAT599
      GT1DMSUAT600
      GT1DMSUAT601
      GT1DMSUAT602
      GT1DMSUAT603
      GT1DMSUAT604
      GT1DMSUAT605
      GT1DMSUAT606
      GT1DMSUAT607
      GT1DMSUAT608
      GT1DMSUAT609
      GT1DMSUAT610
      GT1DMSUAT611
      GT1DMSUAT612
      GT1DMSUAT613
      GT1DMSUAT614
      GT1DMSUAT615
      GT1DMSUAT616
      GT1DMSUAT617
      GT1DMSUAT618
      GT1DMSUAT619
      GT1DMSUAT620
      GT1DMSUAT621
      GT1DMSUAT622
      GT1DMSUAT623
      GT1DMSUAT624
      GT1DMSUAT625
      GT1DMSUAT626
      GT1DMSUAT627
      GT1DMSUAT628
      GT1DMSUAT629
      GT1DMSUAT630
      GT1DMSUAT631
      GT1DMSUAT632
      GT1DMSUAT633
      GT1DMSUAT634
      GT1DMSUAT635
      GT1DMSUAT636
      GT1DMSUAT637
      GT1DMSUAT638
      GT1DMSUAT639
      GT1DMSUAT640
      GT1DMSUAT641
      GT1DMSUAT642
      GT1DMSUAT643
      GT1DMSUAT644
      GT1DMSUAT645
      GT1DMSUAT646
      GT1DMSUAT647
      GT1DMSUAT648
      GT1DMSUAT649
      GT1DMSUAT650
      GT1DMSUAT651
      GT1DMSUAT652
      GT1DMSUAT653
      GT1DMSUAT654
      GT1DMSUAT655
      GT1DMSUAT656
      GT1DMSUAT657
      GT1DMSUAT658
      GT1DMSUAT659
      GT1DMSUAT660
      GT1DMSUAT661
      GT1DMSUAT662
      GT1DMSUAT663
      GT1DMSUAT664
      GT1DMSUAT665
      GT1DMSUAT666
      GT1DMSUAT6
```