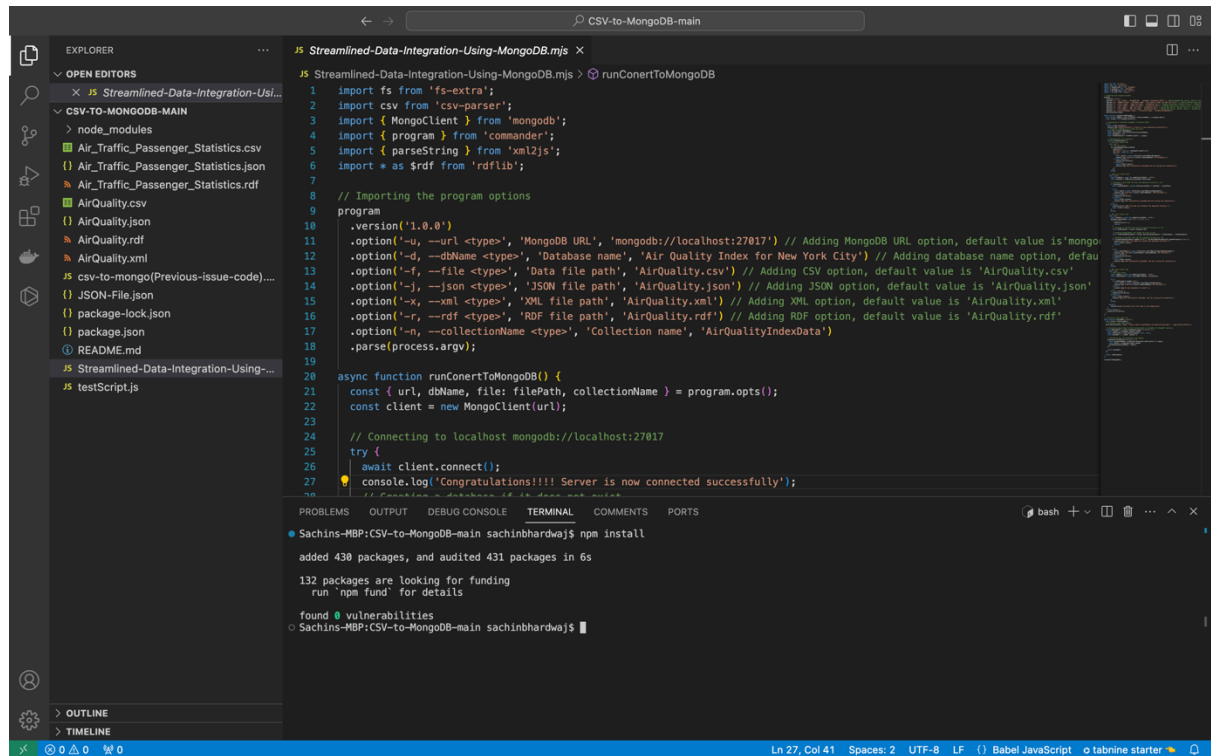


## Project On Streamline Data Integration using MongoDB

Please Find the below step by step screenshots

### 1. Install npm

Command: **npm install**



### 2. Install Dependencies

Command: **npm install fs-extra csv-parser mongodb commander xml2js rdflib**

The screenshot shows a VS Code editor with a file named `Streamlined-Data-Integration-Using-MongoDB.mjs` open. The file contains the following code:

```

1 import fs from 'fs-extra';
2 import csv from 'csv-parser';
3 import { MongoClient } from 'mongodb';
4 import { program } from 'commander';
5 import { parseString } from 'xml2js';
6 import * as $rdf from 'rdflib';
7
8 // Importing the program options
9 program
10   .version('1.0.0')
11   .option('-u, --url <type>', 'MongoDB URL', 'mongodb://localhost:27017') // Adding MongoDB URL option, default value is 'mongodb://localhost:27017'
12   .option('-d, --dbName <type>', 'Database name', 'Air Quality Index for New York City') // Adding database name option, default value is 'Air Quality Index for New York City'
13   .option('-f, --file <type>', 'Data file path', 'AirQuality.csv') // Adding CSV option, default value is 'AirQuality.csv'
14   .option('-j, --json <type>', 'JSON file path', 'AirQuality.json') // Adding JSON option, default value is 'AirQuality.json'
15   .option('-x, --xml <type>', 'XML file path', 'AirQuality.xml') // Adding XML option, default value is 'AirQuality.xml'
16   .option('-r, --rdf <type>', 'RDF file path', 'AirQuality.rdf') // Adding RDF option, default value is 'AirQuality.rdf'
17   .option('-n, --collectionName <type>', 'Collection name', 'AirQualityIndexData')
18   .parse(process.argv);
19
20 async function runConertToMongoDB() {
21   const { url, dbName, file: filePath, collectionName } = program.opts();
22   const client = new MongoClient(url);
23
24   // Connecting to localhost mongodb://localhost:27017
25   try {
26     await client.connect();
27     console.log('Congratulations!!!! Server is now connected successfully');
28   } catch (error) {
29     console.error('Error connecting to MongoDB:', error);
30   }
31 }
32
33 runConertToMongoDB();

```

The terminal window shows the output of `npm install` and `npm install fs-extra csv-parser mongodb commander xml2js rdflib`:

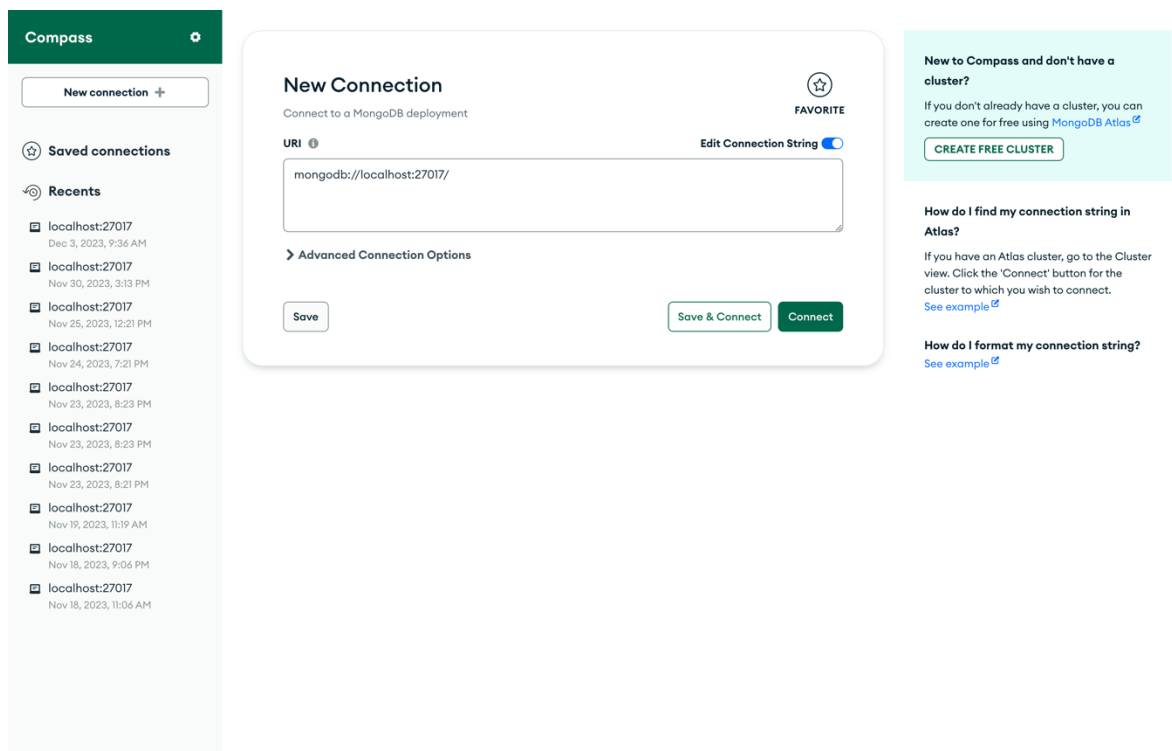
```

Sachins-MBP:CSV-to-MongoDB-main sachinbhardwaj$ npm install
added 430 packages, and audited 431 packages in 6s
132 packages are looking for funding
run 'npm fund' for details

Sachins-MBP:CSV-to-MongoDB-main sachinbhardwaj$ npm install fs-extra csv-parser mongodb commander xml2js rdflib
up to date, audited 431 packages in 3s
132 packages are looking for funding
run 'npm fund' for details

```

### 3. Open MongoDB Compass



### 4. Connect to mongoDB Compass in the URI box **mongodb://localhost:27017**

The image shows two screenshots of the MongoDB Compass interface. The top screenshot displays the 'New Connection' dialog box. On the left, a sidebar shows 'Saved connections' and 'Recents', both containing a list of 'localhost:27017' connections with timestamps. The main dialog area is titled 'New Connection' and includes a 'FAVORITE' star icon. It prompts the user to 'Connect to a MongoDB deployment' and shows the 'URI' as 'mongodb://localhost:27017/'. There is an 'Edit Connection String' toggle and a 'Save' button. At the bottom, there are 'Save & Connect' and 'Connect' buttons. To the right of the dialog, a light blue informational box asks 'New to Compass and don't have a cluster?' and provides a link to 'CREATE FREE CLUSTER'. Below this, two sections provide guidance: 'How do I find my connection string in Atlas?' and 'How do I format my connection string?', both with links to 'See example'. The bottom screenshot shows the 'My Queries' view in Compass. The left sidebar has tabs for 'My Queries', 'Databases', and 'Performance'. Under 'My Queries', there is a search bar and a list of databases: 'admin', 'config', and 'local'. The main area displays a message: 'No saved queries yet. Start saving your aggregations and find queries, you'll see them here.' with a link to 'Visit our Docs'.

## 5. Commands in bash shell

### I. Importing CSV file document to MongoDB database:

#### Command:

node Streamlined-Data-Integration-Using-MongoDB.mjs -u  
 "mongodb://localhost:27017" -d Air Quality Index for New York City -f  
 "AirQuality.csv" -n AirQualityIndexDatainCSV

```

1  README.md
2  Streamlined Data Integration Using MongoDB
3  ## Software Commands (Please run one of these commands based on your file extension):
4  -d, --dbName <type>: The name of the MongoDB database. Default database is 'Air Quality Index for New York City'.
5  -c, --csv <type>: The path of the file format(JSON, CSV, XML, RDF). Default file name is 'AirQuality.csv'.
6  -n, --collectionName <type>: The name of the MongoDB collection to import to. Default is 'AirQualityIndexDatainCSV'.
7
8  ## Software Commands (Please run one of these commands based on your file extension):
9
10  ```bash
11  node Streamlined-Data-Integration-Using-MongoDB.mjs -u <database_url> -d <database_name> -c <file_format_path> -n
12  <collection_name>
13  ```
14
15  2. Importing CSV file document to MongoDB database
16
17  ```bash
18  node Streamlined-Data-Integration-Using-MongoDB.mjs -u "mongodb://localhost:27017" -d Air Quality Index for New York City -f
19  "AirQuality.csv" -n AirQualityIndexDatainCSV
20  ```
21
22  3. Importing JSON file document to MongoDB database
23
24  ```bash
25  node Streamlined-Data-Integration-Using-MongoDB.mjs -u "mongodb://localhost:27017" -d Air Quality Index for New York City -f
26  "AirQuality.json" -n AirQualityIndexDatainJSON
27  ```
28
29  4. Importing XML file document to MongoDB database
30
31  ```bash
32  node Streamlined-Data-Integration-Using-MongoDB.mjs -u "mongodb://localhost:27017" -d Air Quality Index for New York City -f
33  "AirQuality.xml" -n AirQualityIndexDatainXML
34  ```
35
36  PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS
37
38  Inserted 25500 CSV Documents
39  Data successfully uploaded and now closing the connection
40  Sachins-MBP:CSV-to-MongoDB-main sachinbhardwaj$ npm update
41
42  added 17 packages, removed 7 packages, changed 8 packages, and audited 441 packages in 15s
43
44  131 packages are looking for funding
45  run `npm fund` for details
46
47  found 0 vulnerabilities
48  Sachins-MBP:CSV-to-MongoDB-main sachinbhardwaj$ node Streamlined-Data-Integration-Using-MongoDB.mjs -u "mongodb://localhost:27017" -d Air Quality Index f
49  or New York City -f "AirQuality.csv" -n AirQualityIndexDatainCSV
50  (node:75549) [DEP0040] DeprecationWarning: The `punycode` module is deprecated. Please use a userland alternative instead.
51  (Use `node --trace-deprecation ...` to show where the warning was created)
52  Congratulations!!!! Server is now connected successfully
53  Inserted 25500 CSV Documents
54  Data successfully uploaded and now closing the connection
55  Sachins-MBP:CSV-to-MongoDB-main sachinbhardwaj$
  
```

## MongoDB

localhost:27017

Documents  
AirQualityIndexDatainCSV

25.5k 1  
DOCUMENTS INDEXES

AirQualityIndexDatainCSV

Documents Aggregations Schema Indexes Validation

Filter Type a query: { field: 'value' } or [Generate query](#)

EXPLAIN RESET FIND OPTIONS

ADD DATA EXPORT DATA

1 - 20 of 25500

```

_id: ObjectId('656d1b994a9aec36747e7282')
Unique ID: "172653"
Indicator ID: "375"
Name: "Nitrogen dioxide (NO2)"
Measure: "Mean"
Measure Info: "ppb"
Geo Type Name: "UHF34"
Geo Join ID: "203"
Geo Place Name: "Bedford Stuyvesant - Crown Heights"
Time Period: "Annual Average 2011"
Start Date: "12/01/2008"
Data Value: "25.3"
Message: ""

_id: ObjectId('656d1b994a9aec36747e7283')
Unique ID: "172585"
Indicator ID: "375"
Name: "Nitrogen dioxide (NO2)"
Measure: "Mean"
Measure Info: "ppb"
Geo Type Name: "UHF34"
Geo Join ID: "203"
Geo Place Name: "Bedford Stuyvesant - Crown Heights"
Time Period: "Annual Average 2009"
Start Date: "12/01/2008"
Data Value: "26.93"
Message: ""

_id: ObjectId('656d1b994a9aec36747e7284')
  
```

\_MONGOSH

## II. Importing JSON file document to MongoDB database:

### Command:

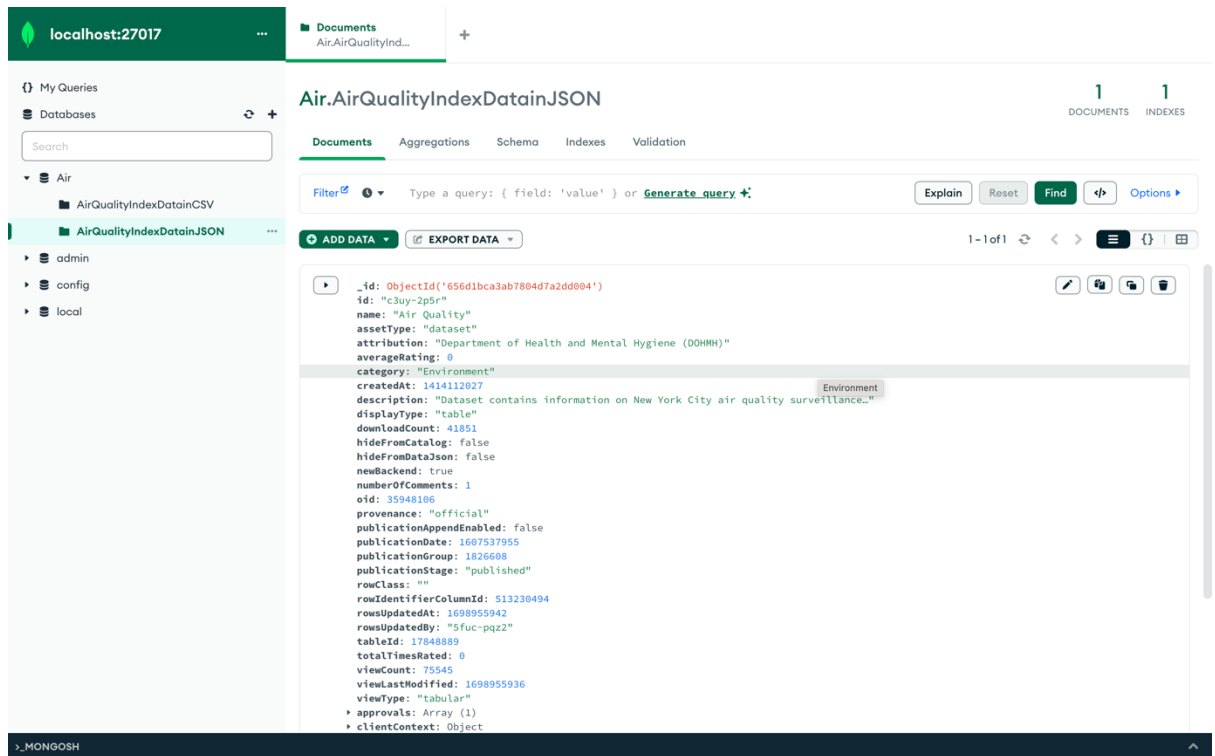
```
node Streamlined-Data-Integration-Using-MongoDB.mjs -u
"mongodb://localhost:27017" -d Air Quality Index for New York City -f
"AirQuality.json" -n AirQualityIndexDatainJSON
```

The screenshot shows a VS Code editor with a project named 'CSV-to-MongoDB-main'. The Explorer sidebar on the left shows the file structure, including a 'README.md' file and a 'testScript.js' file. The main editor area displays the 'README.md' file, which contains instructions for importing CSV, JSON, and XML files into MongoDB. The terminal window at the bottom shows the execution of the command to import the 'AirQuality.json' file into the MongoDB database. The output of the command is visible in the terminal, showing the connection to the MongoDB server and the successful import of the data.

```

1  README.md > # Streamlined Data Integration Using MongoDB > ## Software Commands (Please run one of these commands based on your file extension):
2
3  50
4  51 ```bash
5  52 node Streamlined-Data-Integration-Using-MongoDB.mjs -u <database_url> -d <database_name> -c <file_format_path> -n
6  53 <collection_name>
7  54 ```
8  55
9  56 2. Importing CSV file document to MongoDB database
10 57
11 58 ```bash
12 59 node Streamlined-Data-Integration-Using-MongoDB.mjs -u "mongodb://localhost:27017" -d Air Quality Index for New York City -f
13 60 "AirQuality.csv" -n AirQualityIndexDatainCSV
14 61 ```
15 62
16 63 3. Importing JSON file document to MongoDB database
17 64
18 65 ```bash
19 66 node Streamlined-Data-Integration-Using-MongoDB.mjs -u "mongodb://localhost:27017" -d Air Quality Index for New York City -f
20 67 "AirQuality.json" -n AirQualityIndexDatainJSON
21 68 ```
22 69
23 70 4. Importing XML file document to MongoDB database
24 71
25 72 ```bash
26 73 node Streamlined-Data-Integration-Using-MongoDB.mjs -u "mongodb://localhost:27017" -d Air Quality Index for New York City -f
27 74 "AirQuality.xml" -n AirQualityIndexDatainXML
28 75 ```
29 76
30 77
31 78
32 79
33 80
34 81
35 82
36 83
37 84
38 85
39 86
40 87
41 88
42 89
43 90
44 91
45 92
46 93
47 94
48 95
49 96
50 97
51 98
52 99
53 100
54 101
55 102
56 103
57 104
58 105
59 106
60 107
61 108
62 109
63 110
64 111
65 112
66 113
67 114
68 115
69 116
70 117
71 118
72 119
73 120
74 121
75 122
76 123
77 124
78 125
79 126
80 127
81 128
82 129
83 130
84 131
85 132
86 133
87 134
88 135
89 136
90 137
91 138
92 139
93 140
94 141
95 142
96 143
97 144
98 145
99 146
100 147
101 148
102 149
103 150
104 151
105 152
106 153
107 154
108 155
109 156
110 157
111 158
112 159
113 160
114 161
115 162
116 163
117 164
118 165
119 166
120 167
121 168
122 169
123 170
124 171
125 172
126 173
127 174
128 175
129 176
130 177
131 178
132 179
133 180
134 181
135 182
136 183
137 184
138 185
139 186
140 187
141 188
142 189
143 190
144 191
145 192
146 193
147 194
148 195
149 196
150 197
151 198
152 199
153 200
154 201
155 202
156 203
157 204
158 205
159 206
160 207
161 208
162 209
163 210
164 211
165 212
166 213
167 214
168 215
169 216
170 217
171 218
172 219
173 220
174 221
175 222
176 223
177 224
178 225
179 226
180 227
181 228
182 229
183 230
184 231
185 232
186 233
187 234
188 235
189 236
190 237
191 238
192 239
193 240
194 241
195 242
196 243
197 244
198 245
199 246
200 247
201 248
202 249
203 250
204 251
205 252
206 253
207 254
208 255
209 256
210 257
211 258
212 259
213 260
214 261
215 262
216 263
217 264
218 265
219 266
220 267
221 268
222 269
223 270
224 271
225 272
226 273
227 274
228 275
229 276
230 277
231 278
232 279
233 280
234 281
235 282
236 283
237 284
238 285
239 286
240 287
241 288
242 289
243 290
244 291
245 292
246 293
247 294
248 295
249 296
250 297
251 298
252 299
253 300
254 301
255 302
256 303
257 304
258 305
259 306
260 307
261 308
262 309
263 310
264 311
265 312
266 313
267 314
268 315
269 316
270 317
271 318
272 319
273 320
274 321
275 322
276 323
277 324
278 325
279 326
280 327
281 328
282 329
283 330
284 331
285 332
286 333
287 334
288 335
289 336
290 337
291 338
292 339
293 340
294 341
295 342
296 343
297 344
298 345
299 346
300 347
301 348
302 349
303 350
304 351
305 352
306 353
307 354
308 355
309 356
310 357
311 358
312 359
313 360
314 361
315 362
316 363
317 364
318 365
319 366
320 367
321 368
322 369
323 370
324 371
325 372
326 373
327 374
328 375
329 376
330 377
331 378
332 379
333 380
334 381
335 382
336 383
337 384
338 385
339 386
340 387
341 388
342 389
343 390
344 391
345 392
346 393
347 394
348 395
349 396
350 397
351 398
352 399
353 400
354 401
355 402
356 403
357 404
358 405
359 406
360 407
361 408
362 409
363 410
364 411
365 412
366 413
367 414
368 415
369 416
370 417
371 418
372 419
373 420
374 421
375 422
376 423
377 424
378 425
379 426
380 427
381 428
382 429
383 430
384 431
385 432
386 433
387 434
388 435
389 436
390 437
391 438
392 439
393 440
394 441
395 442
396 443
397 444
398 445
399 446
400 447
401 448
402 449
403 450
404 451
405 452
406 453
407 454
408 455
409 456
410 457
411 458
412 459
413 460
414 461
415 462
416 463
417 464
418 465
419 466
420 467
421 468
422 469
423 470
424 471
425 472
426 473
427 474
428 475
429 476
430 477
431 478
432 479
433 480
434 481
435 482
436 483
437 484
438 485
439 486
440 487
441 488
442 489
443 490
444 491
445 492
446 493
447 494
448 495
449 496
450 497
451 498
452 499
453 500
454 501
455 502
456 503
457 504
458 505
459 506
460 507
461 508
462 509
463 510
464 511
465 512
466 513
467 514
468 515
469 516
470 517
471 518
472 519
473 520
474 521
475 522
476 523
477 524
478 525
479 526
480 527
481 528
482 529
483 530
484 531
485 532
486 533
487 534
488 535
489 536
490 537
491 538
492 539
493 540
494 541
495 542
496 543
497 544
498 545
499 546
500 547
501 548
502 549
503 550
504 551
505 552
506 553
507 554
508 555
509 556
510 557
511 558
512 559
513 560
514 561
515 562
516 563
517 564
518 565
519 566
520 567
521 568
522 569
523 570
524 571
525 572
526 573
527 574
528 575
529 576
530 577
531 578
532 579
533 580
534 581
535 582
536 583
537 584
538 585
539 586
540 587
541 588
542 589
543 590
544 591
545 592
546 593
547 594
548 595
549 596
550 597
551 598
552 599
553 600
554 601
555 602
556 603
557 604
558 605
559 606
560 607
561 608
562 609
563 610
564 611
565 612
566 613
567 614
568 615
569 616
570 617
571 618
572 619
573 620
574 621
575 622
576 623
577 624
578 625
579 626
580 627
581 628
582 629
583 630
584 631
585 632
586 633
587 634
588 635
589 636
590 637
591 638
592 639
593 640
594 641
595 642
596 643
597 644
598 645
599 646
600 647
601 648
602 649
603 650
604 651
605 652
606 653
607 654
608 655
609 656
610 657
611 658
612 659
613 660
614 661
615 662
616 663
617 664
618 665
619 666
620 667
621 668
622 669
623 670
624 671
625 672
626 673
627 674
628 675
629 676
630 677
631 678
632 679
633 680
634 681
635 682
636 683
637 684
638 685
639 686
640 687
641 688
642 689
643 690
644 691
645 692
646 693
647 694
648 695
649 696
650 697
651 698
652 699
653 700
654 701
655 702
656 703
657 704
658 705
659 706
660 707
661 708
662 709
663 710
664 711
665 712
666 713
667 714
668 715
669 716
670 717
671 718
672 719
673 720
674 721
675 722
676 723
677 724
678 725
679 726
680 727
681 728
682 729
683 730
684 731
685 732
686 733
687 734
688 735
689 736
690 737
691 738
692 739
693 740
694 741
695 742
696 743
697 744
698 745
699 746
700 747
701 748
702 749
703 750
704 751
705 752
706 753
707 754
708 755
709 756
710 757
711 758
712 759
713 760
714 761
715 762
716 763
717 764
718 765
719 766
720 767
721 768
722 769
723 770
724 771
725 772
726 773
727 774
728 775
729 776
730 777
731 778
732 779
733 780
734 781
735 782
736 783
737 784
738 785
739 786
740 787
741 788
742 789
743 790
744 791
745 792
746 793
747 794
748 795
749 796
750 797
751 798
752 799
753 800
754 801
755 802
756 803
757 804
758 805
759 806
760 807
761 808
762 809
763 810
764 811
765 812
766 813
767 814
768 815
769 816
770 817
771 818
772 819
773 820
774 821
775 822
776 823
777 824
778 825
779 826
780 827
781 828
782 829
783 830
784 831
785 832
786 833
787 834
788 835
789 836
790 837
791 838
792 839
793 840
794 841
795 842
796 843
797 844
798 845
799 846
800 847
801 848
802 849
803 850
804 851
805 852
806 853
807 854
808 855
809 856
810 857
811 858
812 859
813 860
814 861
815 862
816 863
817 864
818 865
819 866
820 867
821 868
822 869
823 870
824 871
825 872
826 873
827 874
828 875
829 876
830 877
831 878
832 879
833 880
834 881
835 882
836 883
837 884
838 885
839 886
840 887
841 888
842 889
843 890
844 891
845 892
846 893
847 894
848 895
849 896
850 897
851 898
852 899
853 900
854 901
855 902
856 903
857 904
858 905
859 906
860 907
861 908
862 909
863 910
864 911
865 912
866 913
867 914
868 915
869 916
870 917
871 918
872 919
873 920
874 921
875 922
876 923
877 924
878 925
879 926
880 927
881 928
882 929
883 930
884 931
885 932
886 933
887 934
888 935
889 936
890 937
891 938
892 939
893 940
894 941
895 942
896 943
897 944
898 945
899 946
900 947
901 948
902 949
903 950
904 951
905 952
906 953
907 954
908 955
909 956
910 957
911 958
912 959
913 960
914 961
915 962
916 963
917 964
918 965
919 966
920 967
921 968
922 969
923 970
924 971
925 972
926 973
927 974
928 975
929 976
930 977
931 978
932 979
933 980
934 981
935 982
936 983
937 984
938 985
939 986
940 987
941 988
942 989
943 990
944 991
945 992
946 993
947 994
948 995
949 996
950 997
951 998
952 999
953 1000
954 1001
955 1002
956 1003
957 1004
958 1005
959 1006
960 1007
961 1008
962 1009
963 1010
964 1011
965 1012
966 1013
967 1014
968 1015
969 1016
970 1017
971 1018
972 1019
973 1020
974 1021
975 1022
976 1023
977 1024
978 1025
979 1026
980 1027
981 1028
982 1029
983 1030
984 1031
985 1032
986 1033
987 1034
988 1035
989 1036
990 1037
991 1038
992 1039
993 1040
994 1041
995 1042
996 1043
997 1044
998 1045
999 1046
1000 1047
1001 1048
1002 1049
1003 1050
1004 1051
1005 1052
1006 1053
1007 1054
1008 1055
1009 1056
1010 1057
1011 1058
1012 1059
1013 1060
1014 1061
1015 1062
1016 1063
1017 1064
1018 1065
1019 1066
1020 1067
1021 1068
1022 1069
1023 1070
1024 1071
1025 1072
1026 1073
1027 1074
1028 1075
1029 1076
1030 1077
1031 1078
1032 1079
1033 1080
1034 1081
1035 1082
1036 1083
1037 1084
1038 1085
1039 1086
1040 1087
1041 1088
1042 1089
1043 1090
1044 1091
1045 1092
1046 1093
1047 1094
1048 1095
1049 1096
1050 1097
1051 1098
1052 1099
1053 1100
1054 1101
1055 1102
1056 1103
1057 1104
1058 1105
1059 1106
1060 1107
1061 1108
1062 1109
1063 1110
1064 1111
1065 1112
1066 1113
1067 1114
1068 1115
1069 1116
1070 1117
1071 1118
1072 1119
1073 1120
1074 1121
1075 1122
1076 1123
1077 1124
1078 1125
1079 1126
1080 1127
1081 1128
1082 1129
1083 1130
1084 1131
1085 1132
1086 1133
1087 1134
1088 1135
1089 1136
1090 1137
1091 1138
1092 1139
1093 1140
1094 1141
1095 1142
1096 1143
1097 1144
1098 1145
1099 1146
1100 1147
1101 1148
1102 1149
1103 1150
1104 1151
1105 1152
1106 1153
1107 1154
1108 1155
1109 1156
1110 1157
1111 1158
1112 1159
1113 1160
1114 1161
1115 1162
1116 1163
1117 1164
1118 1165
1119 1166
1120 1167
1121 1168
1122 1169
1123 1170
1124 1171
1125 1172
1126 1173
1127 1174
1128 1175
1129 1176
1130 1177
1131 1178
1132 1179
1133 1180
1134 1181
1135 1182
1136 1183
1137 1184
1138 1185
1139 1186
1140 1187
1141 1188
1142 1189
1143 1190
1144 1191
1145 1192
1146 1193
1147 1194
1148 1195
1149 1196
1150 1197
1151 1198
1152 1199
1153 1200
1154 1201
1155 1202
1156 1203
1157 1204
1158 1205
1159 1206
1160 1207
1161 1208
1162 1209
1163 1210
1164 1211
1165 1212
1166 1213
1167 1214
1168 1215
1169 1216
1170 1217
1171 1218
1172 1219
1173 1220
1174 1221
1175 1222
1176 1223
1177 1224
1178 1225
1179 1226
1180 1227
1181 1228
1182 1229
1183 1230
1184 1231
1185 1232
1186 1233
1187 1234
1188 1235
1189 1236
1190 1237
1191 1238
1192 1239
1193 1240
1194 1241
1195 1242
1196 1243
1197 1244
1198 1245
1199 1246
1200 1247
1201 1248
1202 1249
1203 1250
1204 1251
1205 1252
1206 1253
1207 1254
1208 1255
1209 1256
1210 1257
1211 1258
1212 1259
1213 1260
1214 1261
1215 1262
1216 1263
1217 1264
1218 1265
1219 1266
1220 1267
1221 1268
1222 1269
1223 1270
1224 1271
1225 1272
1226 1273
1227 1274
1228 1275
1229 1276
1230 1277
1231 1278
1232 1279
1233 1280
1234 1281
1235 1282
1236 1283
1237 1284
1238 1285
1239 1286
1240 1287
1241 1288
1242 1289
1243 1290
1244 1291
1245 1292
1246 1293
1247 1294
1248 1295
1249 1296
1250 1297
1251 1298
1252 1299
1253 1300
1254 1301
1255 1302
1256 1303
1257 1304
1258 1305
1259 1306
1260 1307
1261 1308
1262 1309
1263 1310
1264 1311
1265 1312
1266 1313
1267 1314
1268 1315
1269 1316
1270 1317
1271 1318
1272 1319
1273 1320
1274 1321
1275 1322
1276 1323
1277 1324
1278 1325
1279 1326
1280 1327
1281 1328
1282 1329
1283 1330
1284 1331
1285 1332
1286 1333
1287 1334
1288 1335
1289 1336
1290 1337
1291 1338
1292 1339
1293 1340
1294 1341
1295 1342
1296 1343
1297 1344
1298 1345
1299 1346
1300 1347
1301 1348
1302 1349
1303 1350
1304 1351
1305 1352
1306 1353
1307 1354
1308 1355
1309 1356
1310 1357
1311 1358
1312 1359
1313 1360
1314 1361
1315 1362
1316 1363
1317 1364
1318 1365
1319 1366
1320 1367
1321 1368
1322 1369
1323 1370
1324 1371
1325 1372
1326 1373
1327 1374
1328 1375
1329 1376
1330 1377
1331 1378
1332 1379
1333 1380
1334 1381
1335 1382
1336 1383
1337 1384
1338 1385
1339 1386
1340 1387
1341 1388
1342 1389
1343 1390
1344 1391
1345 1392
1346 1393
1347 1394
1348 1395
1349 1396
1350 1397
1351 1398
1352 1399
1353 1400
1354 1401
1355 1402
1356 1403
1357 1404
1358 1405
1359 1406
1360 1407
1361 1408
1362 1409
1363 1410
1364 1411
1365 1412
1366 1413
1367 1414
1368 1415
1369 1416
1370 1417
1371 1418
1372 1419
1373 1420
1374 1421
1375 1422
1376 1423
1377 1424
1378 1425

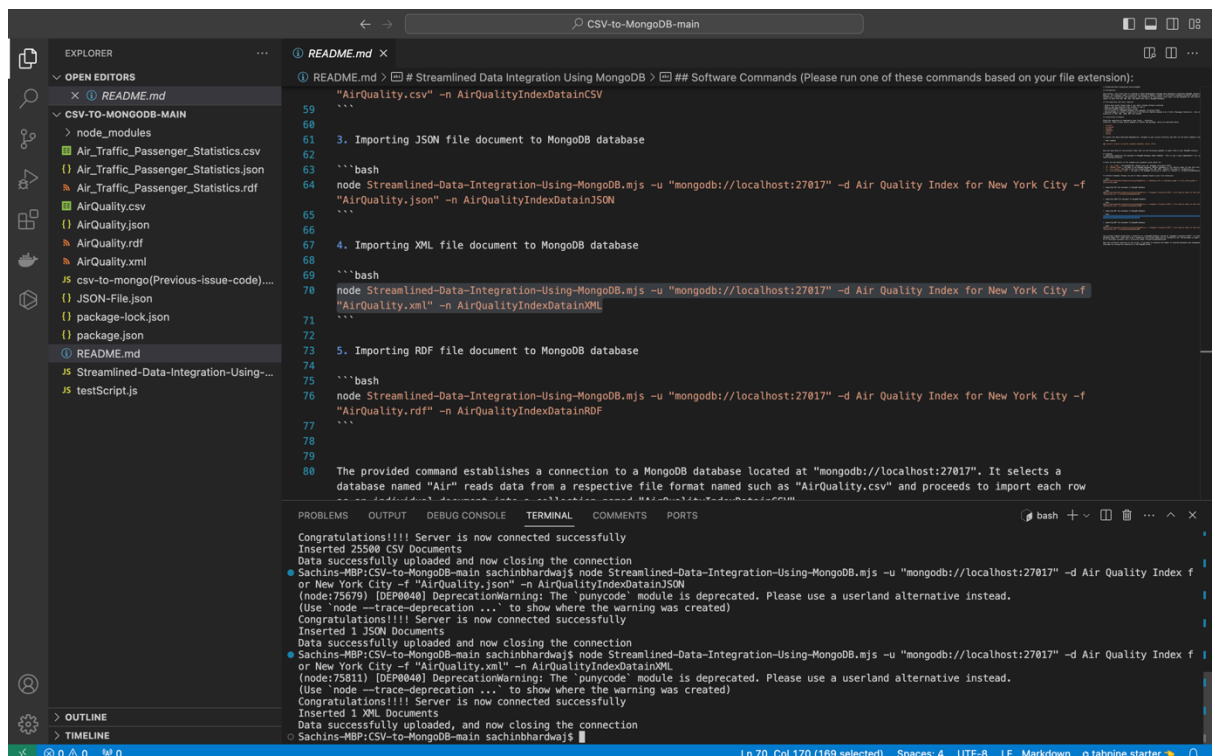
```



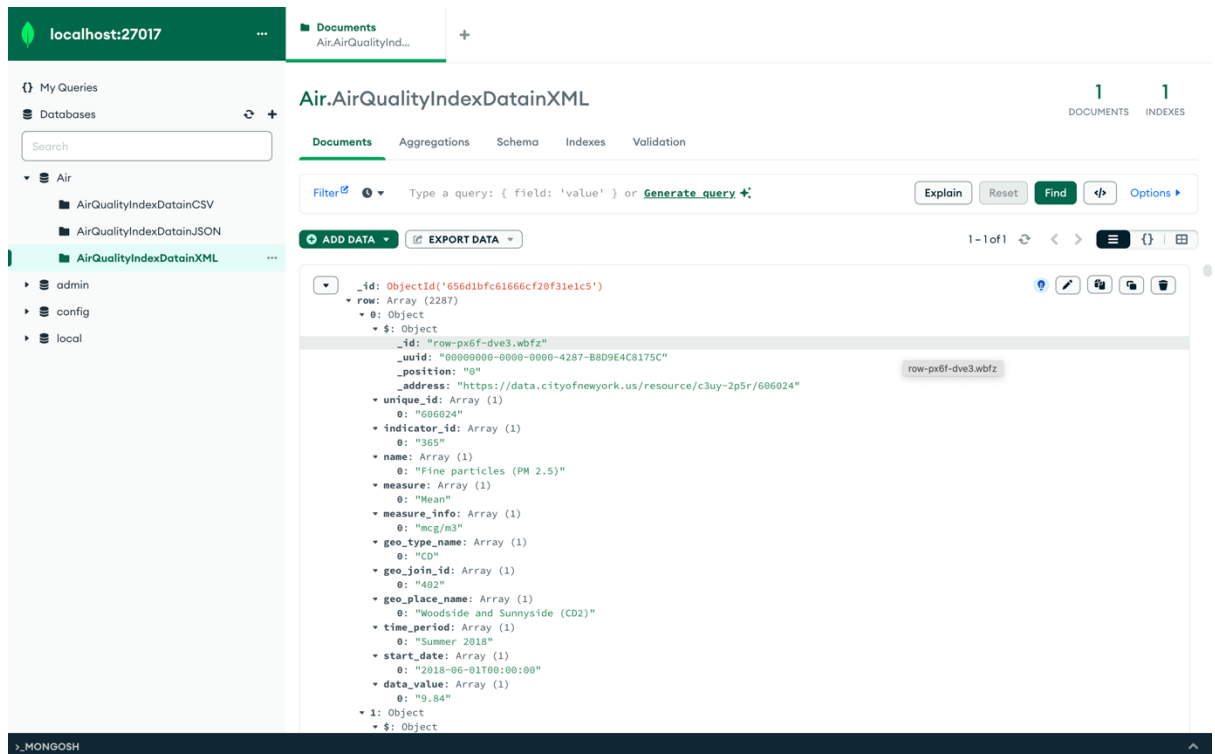
### III. Importing XML file document to MongoDB database:

#### Command:

```
node Streamlined-Data-Integration-Using-MongoDB.mjs -u
"mongodb://localhost:27017" -d Air Quality Index for New York City -f
"AirQuality.json" -n AirQualityIndexDatainJSON
```



## MongoDB



## IV. Importing RDF file document to MongoDB database

### Command:

```
node Streamlined-Data-Integration-Using-MongoDB.mjs -u
"mongodb://localhost:27017" -d Air Quality Index for New York City -f
"AirQuality.json" -n AirQualityIndexDatainJSON
```

The image shows a VS Code editor with a project named 'CSV-to-MongoDB-main'. The Explorer sidebar shows files like 'README.md', 'node\_modules', 'Air\_Traffic\_Passenger\_Statistics.csv', 'Air\_Traffic\_Passenger\_Statistics.rdf', 'AirQuality.csv', 'AirQuality.json', 'AirQuality.xml', 'AirQuality.rdf', 'csv-to-mongo(Previous-issue-code)...', 'package-lock.json', 'package.json', 'README.md', 'Streamlined-Data-Integration-Using-MongoDB.mjs', and 'testScript.js'. The README.md file is open, showing instructions for importing CSV and RDF data into MongoDB. The terminal window shows the execution of a Node.js script that successfully imports data into a MongoDB database.

```

1  README.md > # Streamlined Data Integration Using MongoDB > ## Software Commands (Please run one of these commands based on your file extension):
2
3
4  4. Importing XML file document to MongoDB database
5
6  ```bash
7  node Streamlined-Data-Integration-Using-MongoDB.mjs -u "mongodb://localhost:27017" -d Air Quality Index for New York City -f
8  "AirQuality.xml" -n AirQualityIndexDatainXML
9  ```
10
11 5. Importing RDF file document to MongoDB database
12
13  ```bash
14  node Streamlined-Data-Integration-Using-MongoDB.mjs -u "mongodb://localhost:27017" -d Air Quality Index for New York City -f
15  "AirQuality.rdf" -n AirQualityIndexDatainRDF
16  ```
17
18  The provided command establishes a connection to a MongoDB database located at "mongodb://localhost:27017". It selects a
19  database named "Air" reads data from a respective file format named such as "AirQuality.csv" and proceeds to import each row
20  as an individual document into a collection named "AirQualityIndexDatainCSV"
21
22  Upon the successful execution of the script, it proceeds to showcase the number of inserted documents and subsequently
23  concludes by closing the connection to the MongoDB server.
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

```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL COMMENTS PORTS

```

bash
Congratulations!!!! Server is now connected successfully
Inserted 1 JSON Documents
Data successfully uploaded and now closing the connection
Sachins-MBP:CSV-to-MongoDB-main sachinbhardwaj$ node Streamlined-Data-Integration-Using-MongoDB.mjs -u "mongodb://localhost:27017" -d Air Quality Index f
or New York City -f "AirQuality.xml" -n AirQualityIndexDatainXML
(node:75811) [DEP0040] DeprecationWarning: The 'punycode' module is deprecated. Please use a userland alternative instead.
(Use 'node --trace-deprecation ...' to show where the warning was created)
Congratulations!!!! Server is now connected successfully
Inserted 1 XML Documents
Data successfully uploaded, and now closing the connection
Sachins-MBP:CSV-to-MongoDB-main sachinbhardwaj$ node Streamlined-Data-Integration-Using-MongoDB.mjs -u "mongodb://localhost:27017" -d Air Quality Index f
or New York City -f "AirQuality.rdf" -n AirQualityIndexDatainRDF
(node:75989) [DEP0040] DeprecationWarning: The 'punycode' module is deprecated. Please use a userland alternative instead.
(Use 'node --trace-deprecation ...' to show where the warning was created)
Congratulations!!!! Server is now connected successfully
Inserted 6500 RDF Documents
Data successfully uploaded, and now closing the connection
Sachins-MBP:CSV-to-MongoDB-main sachinbhardwaj$

```

## MongoDB

The image shows the MongoDB Compass interface. The left sidebar shows the 'Air' database with collections 'AirQualityIndexDatainCSV', 'AirQualityIndexDatainJS...', 'AirQualityIndexData...', and 'AirQualityIndexDatainXML'. The main panel shows the 'Air.AirQualityIndexDatainRDF' collection with 6.5k documents and 1 index. The 'Documents' tab is active, showing a list of documents. Each document has a unique '\_id' and a 'rowID' field. The status bar at the bottom indicates 'MongoDB'.

_id	rowID
ObjectId('656d1c456bce99d5669df47f')	"https://data.cityofnewyork.us/resource/c3uy-2p5r/172653"
ObjectId('656d1c456bce99d5669df480')	"https://data.cityofnewyork.us/resource/c3uy-2p5r/172653"
ObjectId('656d1c456bce99d5669df481')	"https://data.cityofnewyork.us/resource/c3uy-2p5r/172653"
ObjectId('656d1c456bce99d5669df482')	"https://data.cityofnewyork.us/resource/c3uy-2p5r/172653"
ObjectId('656d1c456bce99d5669df483')	"https://data.cityofnewyork.us/resource/c3uy-2p5r/172653"
ObjectId('656d1c456bce99d5669df484')	"https://data.cityofnewyork.us/resource/c3uy-2p5r/172653"
ObjectId('656d1c456bce99d5669df485')	"https://data.cityofnewyork.us/resource/c3uy-2p5r/172653"
ObjectId('656d1c456bce99d5669df486')	"https://data.cityofnewyork.us/resource/c3uy-2p5r/172653"

In this situation, there appears to be an issue with the row\_id, although the RDF document was successfully uploaded to MongoDB. This case will be considered as a failed scenario.