



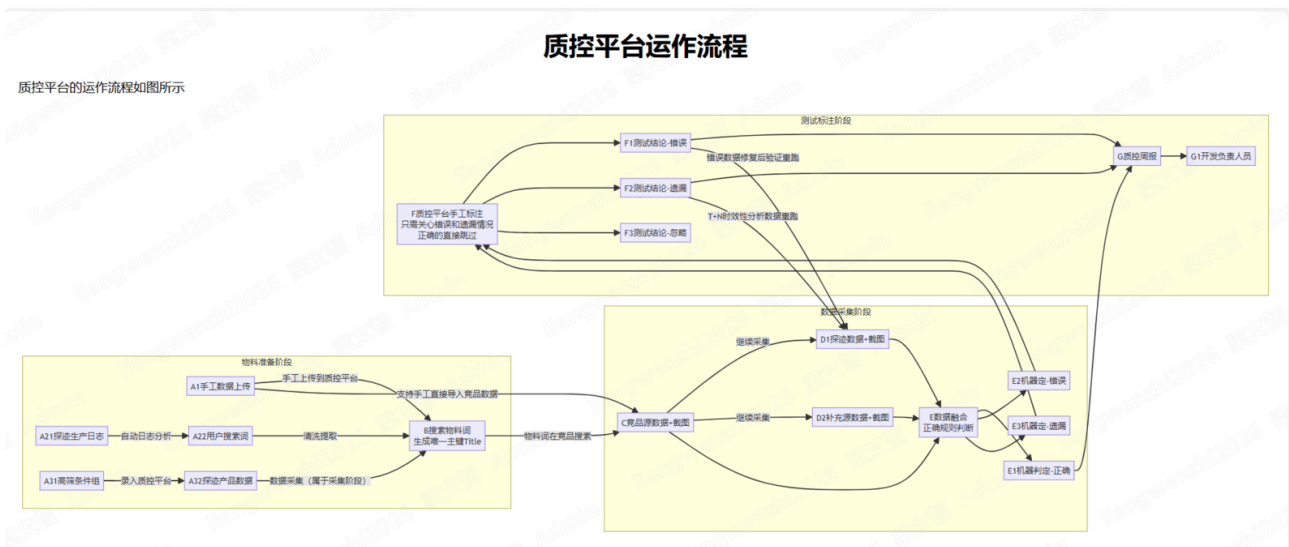
AI Spider 质控平台

点此进入质控平台,外网不可访问

质控平台原理和使用说明

https://alidocs.dingtalk.com/i/nodes/m9bN7RYPWdlyezMbFD1QPdGoWZd1wyK0?doc_type=wiki_doc&utm_medium=main_vertical&utm_scene=team_space&utm_source=search

质控平台运作流程



一 物料准备阶段 (用于生成 spider 库中的数据种子)

1 A1 手工数据上传-->竞品源数据 + 截图

应用场景：不需要去竞品网站爬取数据，竞品数据已经保存在文件中

具体如下：

1 如果搜索词不是简单类型的 如 excel 表格或者其他，和传入的这些数据不用去网站采集数据而是所有数据都在文件中则选择物料文件上传和修改 data-quality-monitor 文件夹里面的 quality 里面的 view.py 文件中的 def upload_excel 函数 在里面根据传入的文件构造 spider 库中的数据，状态码设置为 0.

- [selenium爬虫资源集群](#)
- [\[用户数据\]LQJ的数据图](#)
- [\[用户数据\]LQJ的数据二图](#)
- [\[物料\]物料文件上传](#)
- [\[物料\]名单配置上传](#)
- [\[工具\]OSS文件上传](#)
- [\[统计\]运行情况和统计数据](#)
- [\[标注\]工商数据](#)
- [\[标注\]工商企查查数据](#)
- [\[标注\]行政处罚数据](#)

```

def upload_excel(request):
    fangnan
    def create_table(data):
        table_html = '<table border="1">'
        table_html += '<tr><th>Index</th><th>Name</th><th>Message</th></tr>'
        for item in data:
            table_html += f'<tr><td>{item["index"]}</td><td>{item["name"]}</td><td>{item["msg"]}</td></tr>'
        table_html += '</table>'
        return table_html

    fangnan + 1
    def make_tungee_title(policy_name, tungee_data):
        """
        文件上传时定义唯一主键
        :param policy_name:
        :param tungee_data:
        :return:
        """

    fangnan
    def title_kuohao_clear(title):
        title = title.replace("(", "(")
        title = title.replace(")", ")")
        return title

    if policy_name == '限制高消费':
        return title_kuohao_clear(tungee_data['company_name'] + "+" + policy_name + "+" + tungee_data['content'])
    if policy_name == '行政处罚':
        return title_kuohao_clear(tungee_data['company_name'] + "+" + policy_name + "+" + tungee_data['content'])
    if policy_name == '失信被执行人':
        return title_kuohao_clear(tungee_data['company_name'] + "+" + policy_name + "+" + tungee_data['content'])
    if policy_name == '被执行人':
        return title_kuohao_clear(tungee_data['company_name'] + "+" + policy_name + "+" + tungee_data['content'])
    if policy_name == '严重违法':
        return title_kuohao_clear(

```

2 A21 探迹生产日志-->竞品源数据 + 截图

应用场景：需要去网站爬取数据，只有公司名等不知道详细数据

具体：

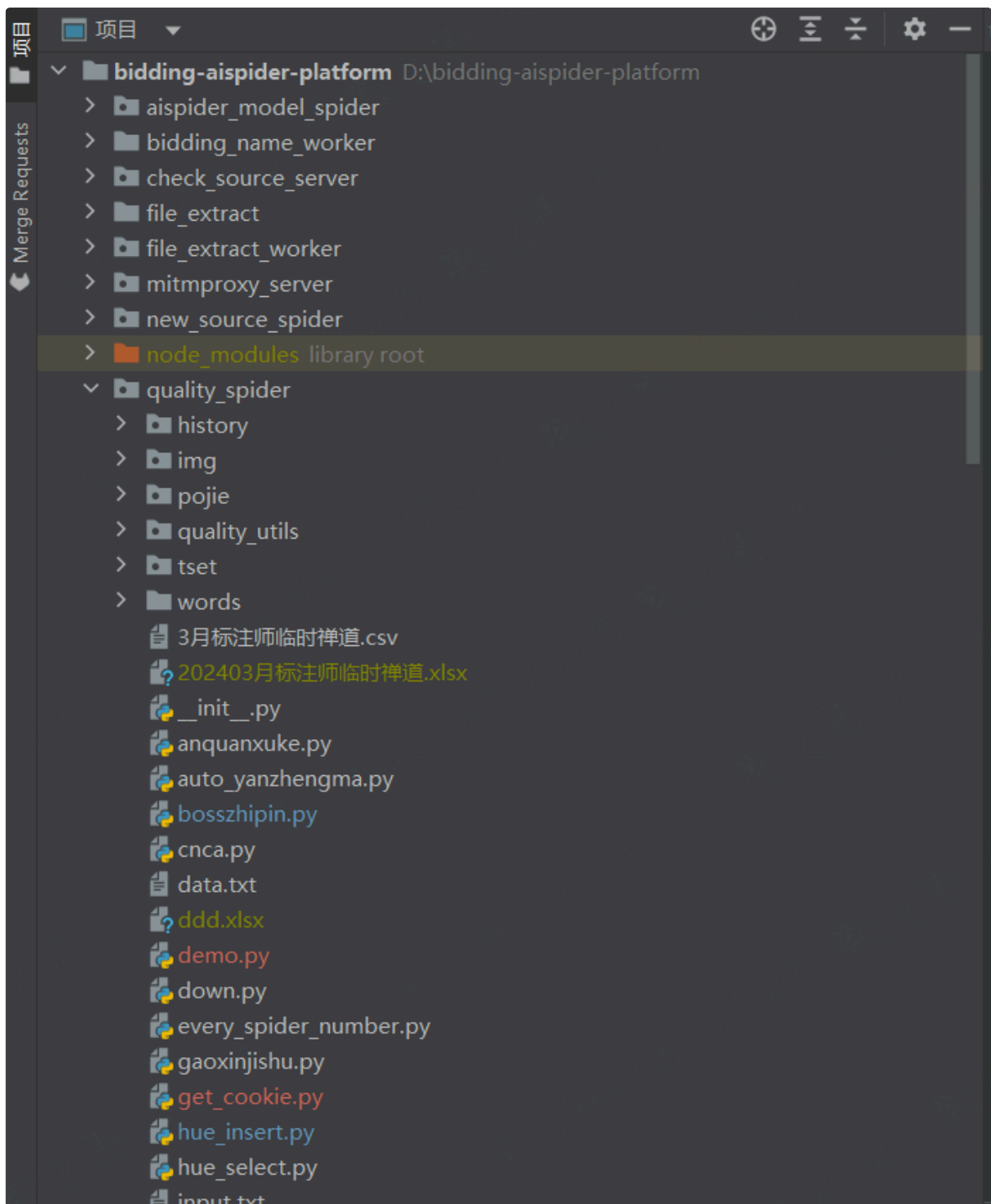
1: 如果搜索词比较简便如（公司名+XXX+XXX）和需要该搜索词去某某网站上去搜索，可以往 Template configs 添加新的种子源 再到质控平台欢迎页在名单配置上传就可以上传搜索词了

QUALITY	
Accuracy datas	+ Add
Button settings	+ Add
Company lists	+ Add
Hands datas	+ Add
Hands2 datas	+ Add
Merge datas	+ Add
Spider datas	+ Add
Spider search datas	+ Add
Template configs	+ Add
Tungee datas	+ Add
Tungee search datas	+ Add

- [selenium爬虫资源集群](#)
- [\[用户数据\]LQJ的数据图](#) 
- [\[用户数据\]LQJ的数据二图](#) 
- [\[物料\]物料文件上传](#)    
- [\[物料\]名单配置上传](#) 
- [\[工具\]OSS文件上传](#)
- [\[统计\]运行情况和统计数据](#)
- [\[标注\]工商数据](#) 
- [\[标注\]工商企查查数据](#) 
- [\[标注\]行政处罚数据](#) 
- [\[标注\]二手车过户数据](#) 

2 编写 spider 爬虫脚本

1 在 bidding-aispider-platform 中的 quality_spider 中添加新的.py 爬虫脚本，根据搜索词去网站搜索数据，构建数据，然后存储成下面 data 的这种数据结构 title 为匹配探迹数据的主键（重要） source 为网站的名称 type 为该爬虫爬取数据的类型 data 为爬取的数据 url 为爬取网站的 url screenshot_url 为截图的 url week_id 为数据存储时的时间 存储到 spider 库中，状态码更新为 1。



```
title = tree.xpath("./h1/@title")[0]
# 截图
screenshot_url = self.take_screenshot_and_upload_to_oss(self.
data = {
    "title": company_name + "+" + title,
    "source": "boss直聘",
    "type": "招聘",
    "data": json.dumps(obj: {
        "company_name": company_name,
        "work_name": title,
        "lrDate_time": lrDate_time,
        "updated_time": updated_time,
    }, separators=(',', ':'), ensure_ascii=False),
    "url": url,
    "screenshot_url": screenshot_url,
    "week_id": get_week_id(),
}
self.insert_data_into_quality_spiderdata(data)
datas.append(data)
```

3 A31 高筛条件组（咨询方楠）

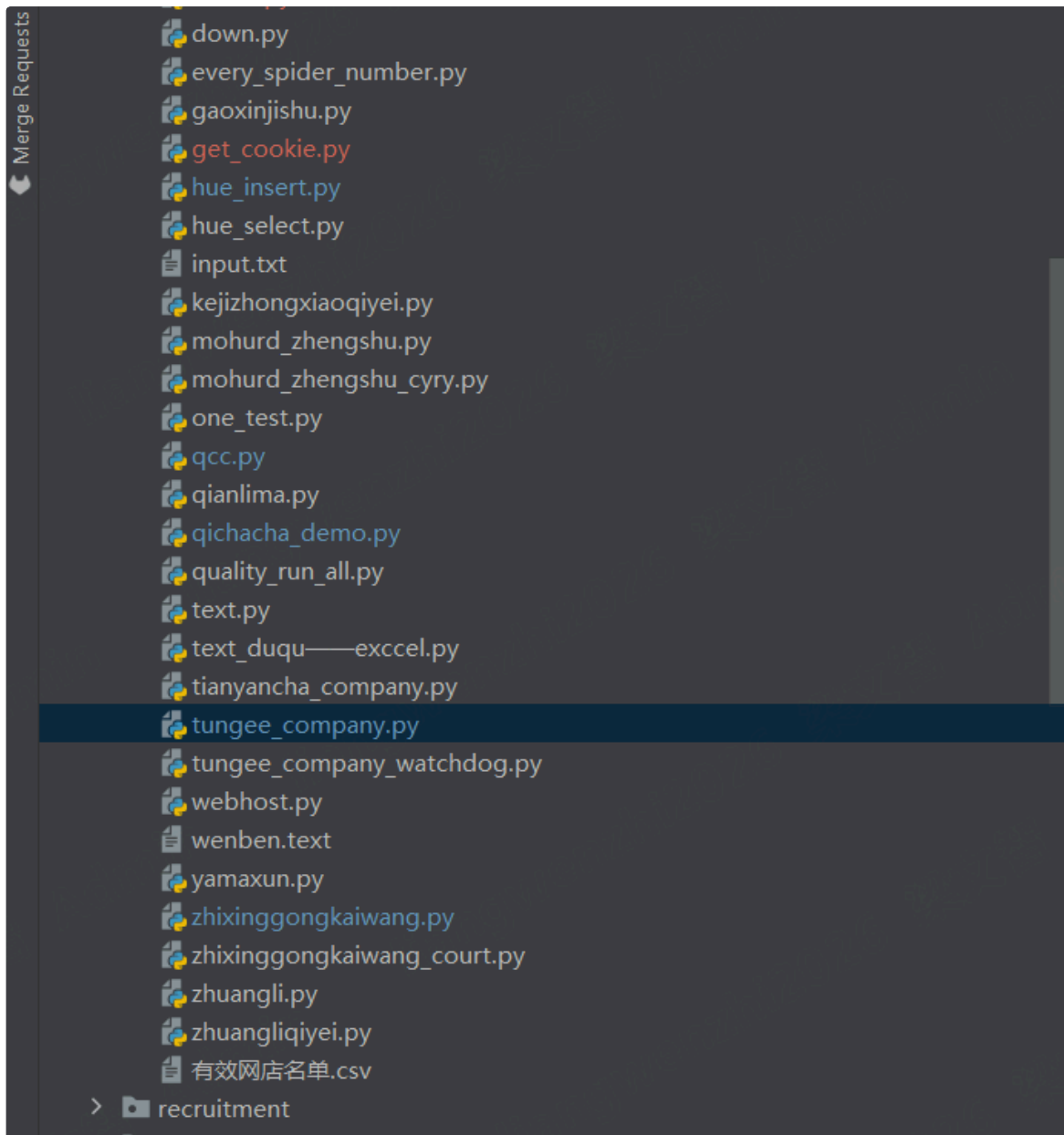
二 数据采集阶段

1 D1 探迹数据 + 截图

应用场景：只有探迹和竞品数据比对

具体：

1 编写 tungee_company.py 数据采集脚本，在已有的框架下编写脚本，首先从 spider 库中取出数据并根据数据的公司名或者其他在探迹对应的版本下去搜索获取想要的 data_msg title 为匹配 spider 库的主键 spider_id 为从 spider 库中数据的 id source 为探迹 type 为数据的类型（如 限制高消费）data 为在探迹的数据，状态码更新为 2.



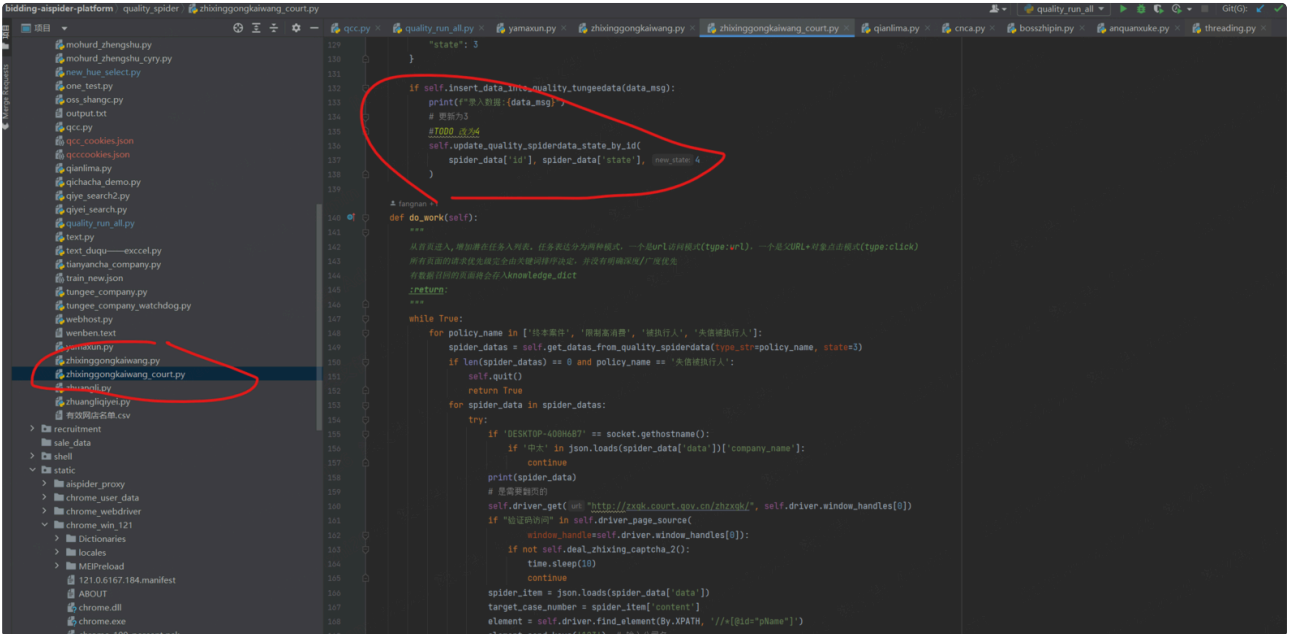
```
value = t.xpath('//td/text()')
datas[key[0]] = value[0]
data_msg = {
    "title": self.make_tungee_title(policy_name, datas),
    "spider_id": word['id'],
    "source": "探迹",
    "type": policy_name,
    "data": json.dumps(datas, separators=(',', ':'),
                       ensure_ascii=False),
    "url": self.driver_current_url(window_handle=window_handle),
    "screenshot_url": self.take_screenshot_and_upload_to_oss(
        window_handle=window_handle),
    "week_id": word['week_id'],
}
```

2 D2 补充源数据 + 截图

应用场景：存在除了探迹和竞品的第三方网站（限制高消费）

具体：

如需要到第三方平台进行比较的数据，如（限制高消费）状态码设置为 4

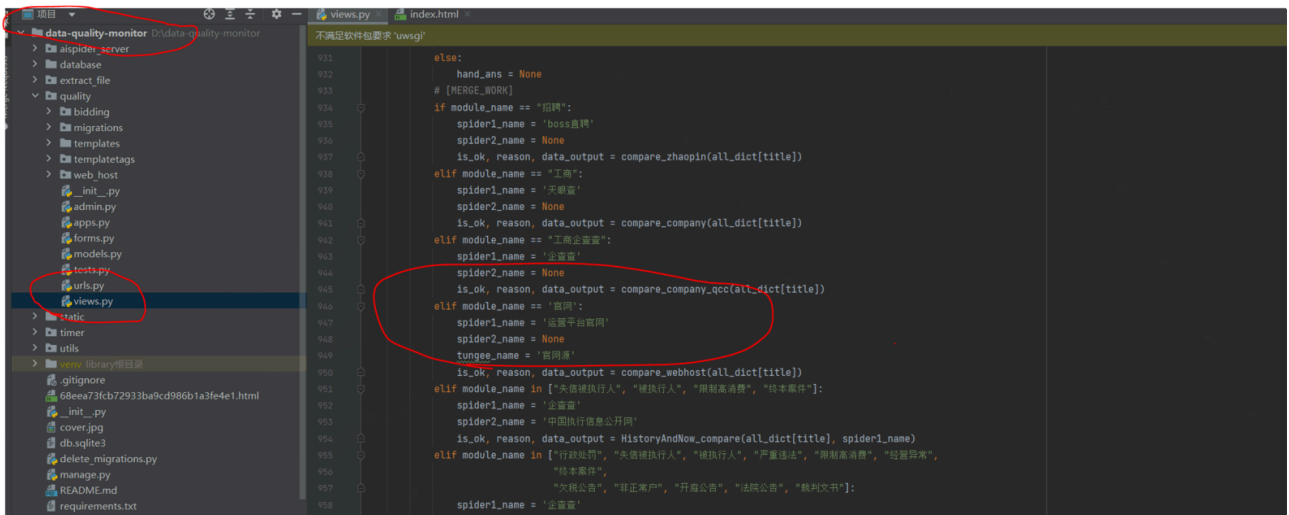


```
217     "state": 3
218 }
219
220 if self.insert_data_into_quality_tungeedata(data_msg):
221     print("插入数据 -> {data_msg}")
222     # 更新力3
223     # TODO: 这里
224     self.update_quality_spiderdata_state_by_id(
225         spider_data['id'], spider_data['state'], new_state=4
226     )
227
228
229
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
```

3 E 数据融合正确规则判断

具体：

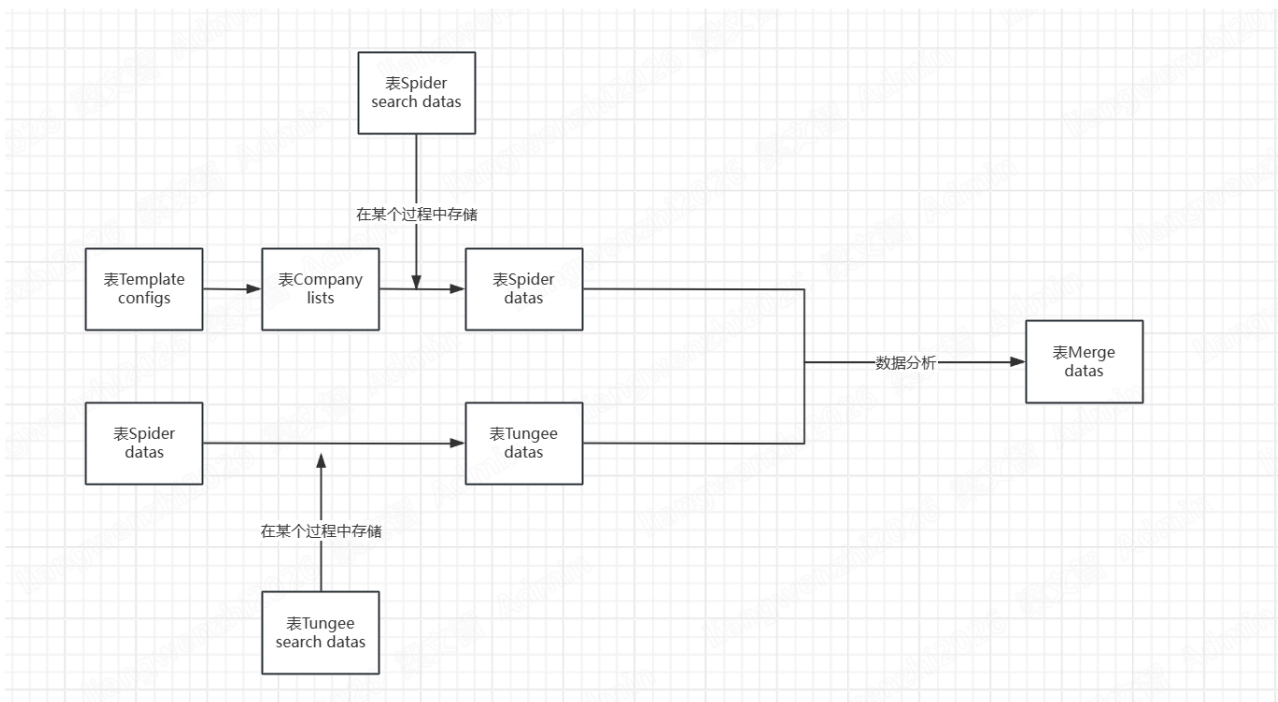
最后，在 data-quality-monitor 这个文件的 views 然后在标红的代码块附近添加新源的新函数，构建比对规则。



```
931     else:
932         hand_ans = None
933         # [MERGE_WORK]
934         if module_name == "招聘":
935             spider1_name = "boss直聘"
936             spider2_name = None
937             is_ok, reason, data_output = compare_zhaopin(all_dict[title])
938         elif module_name == "工商":
939             spider1_name = "天眼查"
940             spider2_name = None
941             is_ok, reason, data_output = compare_company(all_dict[title])
942         elif module_name == "工商企业查询":
943             spider1_name = "企查查"
944             spider2_name = None
945             is_ok, reason, data_output = compare_company_qcc(all_dict[title])
946         elif module_name == "官网":
947             spider1_name = "运营平台官网"
948             spider2_name = None
949             tungee_name = "官网源"
950             is_ok, reason, data_output = compare_webhost(all_dict[title])
951         elif module_name in ["失信被执行", "被执行人", "限制消费令", "终本案件"]:
952             spider1_name = "企查查"
953             spider2_name = "中国执行信息公开网"
954             is_ok, reason, data_output = HistoryAndNow_compare(all_dict[title], spider1_name)
955         elif module_name in ["行政处罚", "失信被执行", "失信被执行人", "严重违法", "经营异常", "终本案件", "法院公告", "非正常户", "开庭公告", "开庭公告", "裁判文书"]:
956             spider1_name = "企查查"
```

三 数据库表功能解释：

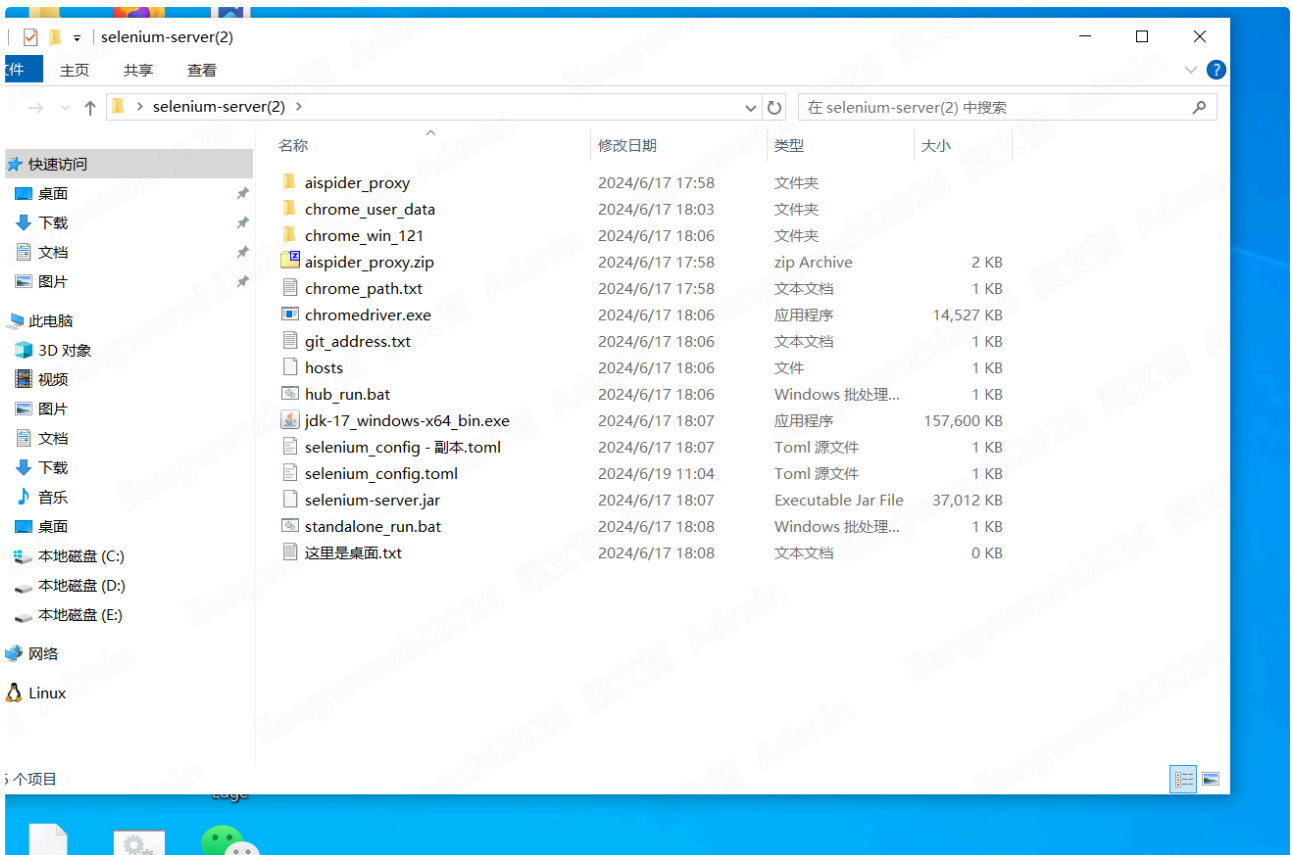
前面是从业务角度进行梳理的，但往往直接操作表效率更高



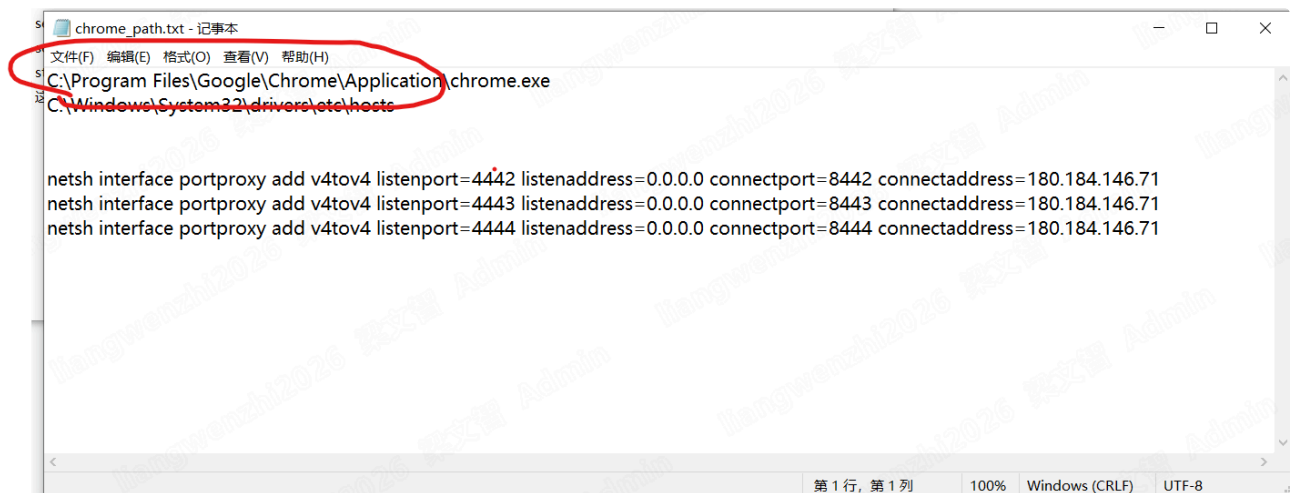
质控平台的大体工作流程是根据种子爬取数据存储到 spiderdatas 状态码设置为 0，在采集过程中把每一个数据采集的最后一步的截图存储到 spider search datas,然后从 spiderdatas 表里面取出数据在探迹产品端进行数据采集，将采集的数据存储到 Tungeedatas 里面，状态码更新为 2，在采集过程中把每一个数据采集的最后一步的截图存储到 Tungee search datas 里面，然后根据一定的比对方式，把结果存储到 Merge datas 里面

四 selenium 集群启动流程

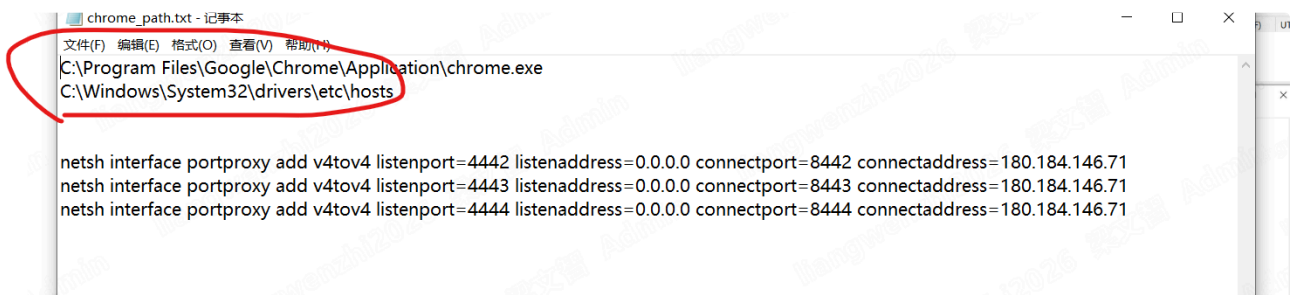
1 首先，需要检查每一个机器是否是 121 版本，和是否下载了 jdk_17-windows-x64_bin.exe 和是否把 selenium-server 这个文件夹放在桌面上。(只限在同一个私域里面运行，如探迹公司的内部网络)



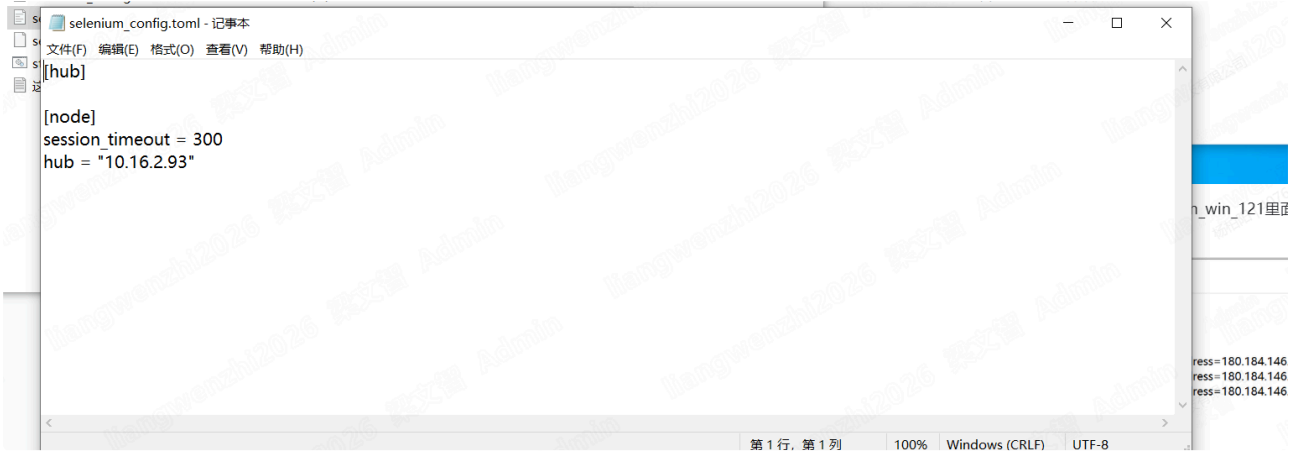
怎么 · 检查是否是 121 版本呢，打开以下这个路径把里面所有文件都删除掉，把上面 chorm_win_121 里面的所有文件复制到删除的文件夹里面



防止浏览器自动更新 找到以下路径把 hosts 文件替换成文件夹下的 hosts 文件



2 配置 selenium_config.toml 文件



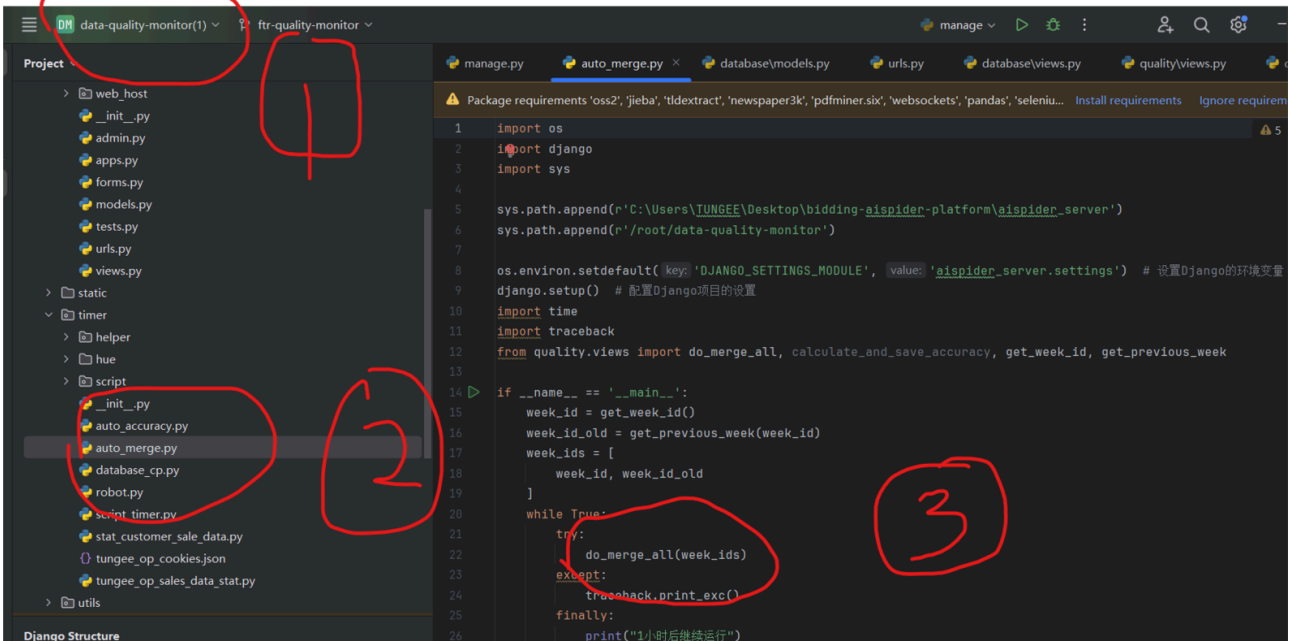
把所有主机的这个文件的 hub 改为运行程序的主机的 ip (运行程序的主机改为 http://+ 主机 ip+:4444)

3 最后，启动子节点运行 standalone_run.bat 主节点运行 hub_run.bat

五：某个维度的调试方法：

如图所示：

找到 data-quality-monitor 文件下的 auto_merge.py 里面的 d_merge_all()的函数：导航到这个函数里面



如果你想调式某个维度，可以在下图里面进行修改，然后就可以调试了

```
manage.py auto_merge.py database\models.py urls.py database\views.py quality\views.py
95 def deal_merge(week_id, module_name):
1056     traceback.print_exc()
1057
1058
4 usages
1059 def do_merge_all(week_ids=None):
1060     if not week_ids:
1061         week_ids = [
1062             "202401", "202402", "202403", "202404", "202405", "202406", "202407", "202408", "202409",
1063             "202410", "202411", "202412", "202413", "202414", "202415", "202416", "202417", "202418",
1064         ]
1065         week_ids.sort(reverse=True)
1066     for week_id in week_ids:
1067         print(f"正在处理:{week_id}")
1068         unique_types = SpiderData.objects.filter(week_id=week_id).values_list('type', flat=True).distinct()
1069         for module_name in unique_types:
1070             print(f"正在处理:{week_id}->{module_name}")
1071             if socket.gethostname() in ['DESKTOP-400H687', 'liangwenzhi2026']:
1072                 # if module_name not in ["生产许可证"] or week_id != "202420":
1073                 if module_name not in ["新限制高消费"]:
1074                     continue
1075             print(week_id, module_name)
1076             deal_merge(week_id, module_name)
1077
1078
5 usages
1079 def calculate_and_save_accuracy(week_ids):
```

官网的启动在 在 bidding-aispider-platform 中的 quality_spider 中

```
tungee_company.py
tungee_company_watchdog.py
tungee_company_wuliu.py
webhost.py
wenben.txt
yamaxun.py
zhixinggongkaiwang.py
```

admin 后台看任务状态可以在 180.184.146.71:8000/admin 账号密码为 liqiuju666 147258369Asd