```python
__author__ = 'Luzaofa'
__date__ = '2019/8/14 14:13'

import time
import json
import requests
import multiprocessing as mp

from Helper.ProxyHelper import proxy
from Helper.DBHelper import DB_Helper
from Helper.RedisHelper import RedisHelper
from Helper.UserAgentHelper import UserAgent

redis_helper = RedisHelper()


class SpiderHelper(object):
    '''数据爬取封装'''

    def __init__(self, IP=False):
        self.IP = IP    # 控制是否添加代理 IP
        self.db_helper = DB_Helper()

    def data_mp(self, func, pros):
        '''进程池'''
        pool = mp.Pool(processes=4)
        for pro in pros:
            pool.apply_async(func, args=(json.loads(pro)['url'],))
        pool.close()
        pool.join()

    def get_response(self, url):
        '''获取网页数据'''
        headers = {"User-Agent": UserAgent().random()}    # 随机伪装浏览器
        try:
            if self.IP:
                host, port = proxy()
                if host and port:
                    print('启动代理:', host, port)
                    proxies = {"http": 'http://{}:{}'.format(host, port)}    # 代理 IP
                    response = requests.get(url=url, headers=headers, proxies=proxies).text
                else:
                    return None
            else:
                print('未启动代理')
                response = requests.get(url=url, headers=headers)
            response.encoding = 'utf-8'
            data = response.text
            print('数据采集成功')
            return data
        except Exception as e:
            print(e)
            return None

    def get_content(self, response):
        '''解析数据'''
        print('数据解析成功')
        return response

    def write_2_db(self, result):
        '''写入数据库'''
        sql = 'XXX'
        data = self.db_helper.query(sql)
        if data:
            print('数据入库成功')
            return result
        return False

    def main(self, url):
        '''逻辑入口'''
        response = self.get_response(url)
        if response:
            results = self.get_content(response)
            if self.write_2_db(results):
                print('执行完毕!')
                return True
            print('数据库插入失败')
        print('解析出错')
        return False

    def mp_main(self):
        '''多进程入口'''
        urls = redis_helper.get_beach('url', beach_num=10)    #  批量获取
        num = len(urls)
        if num > 1:
            print('该批次获取到：{0}个任务'.format(num))
            self.data_mp(self.main, urls)    #  多进程
        else:
```

```
                        time.sleep(60 * 1)
                        print('未获取到任务，一分钟后重新获取数据')


def add_pros():
        '''添加采集 URL'''
        URL = 'http://www.baidu.com{0}'
        for page in range(100):
                url = URL.format('')
                # url = URL.format(page)
                item = json.dumps({'url': url})
                redis_helper.put_set_time(item, 'url', ex_time=1200)


def add_IP():
        '''定时向 redis 添加 IP'''
        host, port = proxy()
        item = json.dumps({'host': host, 'port': port})
        print('正在添加：', item)
        redis_helper.put_set_time(item, 'ip', ex_time=120)


if __name__ == '__main__':
        add_pros()   #  添加任务（测试）
        spider = SpiderHelper(IP=False)
        while True:
                spider.mp_main()
```