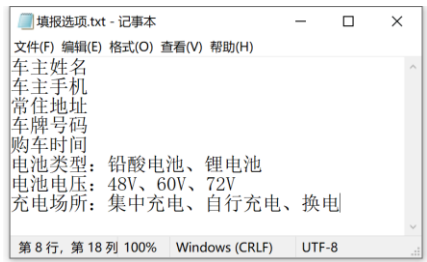


# 11. 文件 课后作业

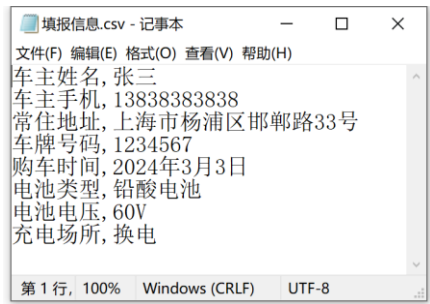
## 1) 信息填报 （30 分）

近年来，电动自行车引发的火灾急剧增加，给人民群众生命财产安全造成严重威胁，国务院安委办召开视频会议，部署全面启动电动自行车安全隐患全链条整治行动。上海在全市范围内开展电动自行车基础数据填报工作。具体的填报选项保存在“**填报选项.txt**”文件中，如下：



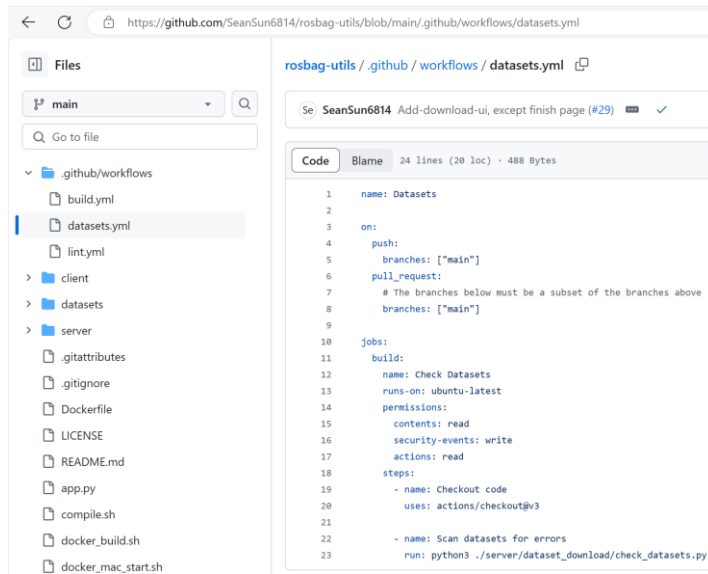
编写代码，从文件中读取填报选项，然后让用户填写信息，并将填报信息保存到“**填报信息.csv**”文件中。

请输入车主姓名：张三  
请输入车主手机：13838383838  
请输入常住地址：上海市杨浦区邯郸路33号  
请输入车牌号码：1234567  
请输入购车时间：2024年3月3日  
请选择 电池类型 ：  
1 -- 铅酸电池  
2 -- 锂电池  
输入选择数字：1  
请选择 电池电压 ：  
1 -- 48V  
2 -- 60V  
3 -- 72V  
输入选择数字：2  
请选择 充电场所 ：  
1 -- 集中充电  
2 -- 自行充电  
3 -- 换电  
输入选择数字：3



## 2) 配置读取 (20 分)

yaml 文件十分常用，下图是在 github 中一个网页中显示的 datasets.yml 文件。



请编写代码，读取 datasets.yml 中的内容，并以 json 输出效果如下：

```
{
  "name": "Datasets",
  "true": {
    "push": {
      "branches": [
        "main"
      ]
    },
    "pull_request": {
      "branches": [
        "main"
      ]
    }
  },
  "jobs": {
    "build": {
      "name": "Check Datasets",
      "runs-on": "ubuntu-latest",
      "permissions": {
        "contents": "read",
        "security-events": "write",
        "actions": "read"
      },
      "steps": [
        {
          "name": "Checkout code",
          "uses": "actions/checkout@v3"
        },
        {
          "name": "Scan datasets for errors",
          "run": "python3 ./server/dataset_download/check_datasets.py"
        }
      ]
    }
  }
}
```

【注】上图红色部分为 true，是因为在 yaml 中 on 为关键字，其含义为 true。然后，输出最后一行中的“run”变量：

需要运行： `python3 ./server/dataset_download/check_datasets.py`





















3) 批量更名 （50 分）

编写代码，运行程序时，首先在该程序子目录下生成一个子目录：**temp1**。这个子目录用来存放后续的文件。

然后，编写代码创建 10 个文本文件（见下图左侧），文件名分别为：1.txt, 2.txt, …, 10.txt，每个文件中的内容分别为数字 1~10，即，1.txt 中内容为：**1**，2.txt 中的内容为：**2**，以此类推。

最后，用循环语句，遍历 temp1 子目录下的所有文件，并将 1.txt 修改为 001.txt，2.txt 修改为 002.txt，以此类推，最后的效果见下图右侧。

运行效果：

temp1	> temp1
名称	名称
 1.txt	 001.txt
 2.txt	 002.txt
 3.txt	 003.txt
 4.txt	 004.txt
 5.txt	 005.txt
 6.txt	 006.txt
 7.txt	 007.txt
 8.txt	 008.txt
 9.txt	 009.txt
 10.txt	 010.txt