上上个周末，Maybe，记不清了，也许是上上上个周末，总之那会儿我在听英语，但那其实就是催眠曲，反正很快就靠着椅子睡着了。而后很自然地做起了美梦，梦里却为那台办公笔记本电脑犯着愁，为啥愁？磁盘空间又满了，是完完整整地满了，1B都插不进去的那种。为了释放出可怜的磁盘空间，哪怕一点点，这1年里，可谓想了不少招数，这次似乎真的无计可施了。那该怎么办？一觉被吓醒。

眯着眼睛想了想，也许是有些文件很大，或许可以找到它们，删除一些来释放一些空间。

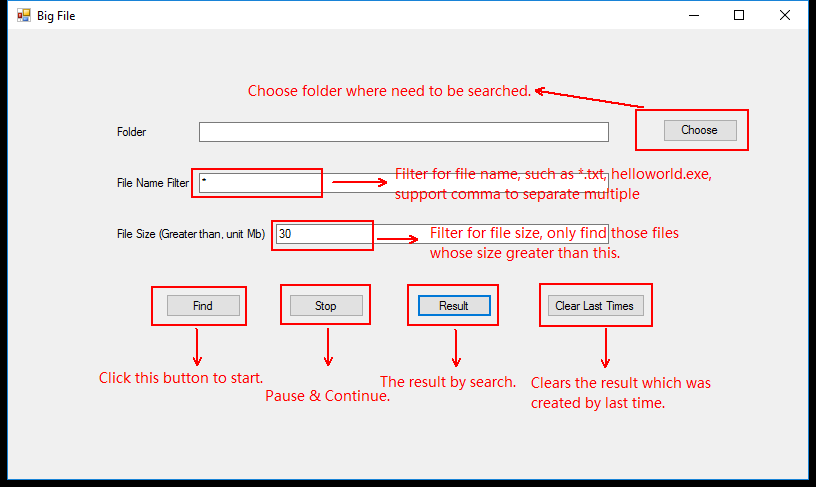
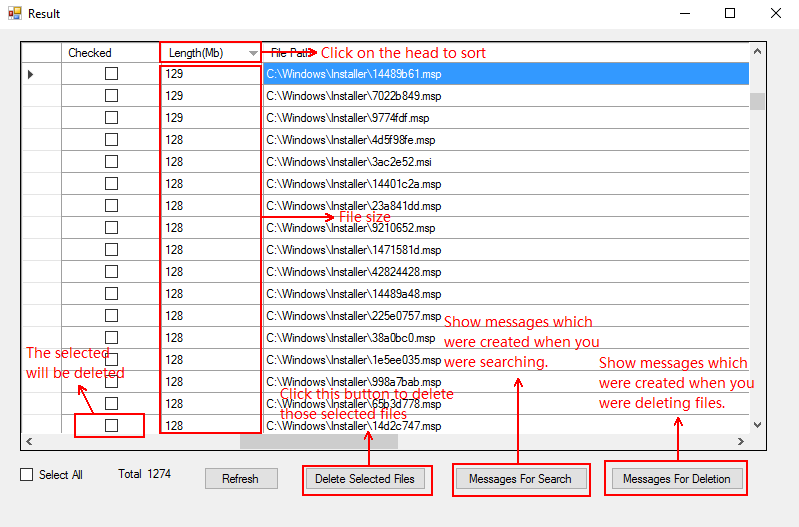
于是乎坐起来，趁着阳光还不错，便写了这个工具。它主要提供以下功能：

1. 输入字符串去搜索符合文件名规则的文件
2. 输入文件大小去搜索大于该Size的文件
3. 查找的结果会被保存在SQLite数据库中
4. 可以暂停查找，也可以继续
5. 对查找结果可以再进行排序，方便二次筛选
6. 选中某些查找结果进行删除操作
7. 如果在查找和删除的过程中发生了异常，比如资源被占用而无法删除，可以通过Messages窗口查看

之前发布的工具很少去写文字来表达设计上的优势和缺点，以后可能会多写一些设计思路。这个工具在设计时，主要考虑了以下问题：

1. 由于针对文件系统的操作通常是比较消耗计算机资源的，所以考虑了暂停功能，这样当需要优先在其它应用下操作时，可以先暂停
2. 有的时候我们会查找出大量的文件需要去甄选，可能一时半会儿难以做完，所以考虑了将查找的结果保存到SQLite数据库中，这样下次可以直接打开上次的查找结果继续甄选
3. 为了面向更多的操作系统，核心的代码都基于.Net Standard编写，这意味着可以方便地运行在多种操作系统上（当前仅表示层使用WinForm实现）
4. 考虑只是个简单的工具，Layer的划分比较简单随意，但在设计时独立了Business Layer，也解除了对DA的依赖，为未来可能的独立化该组件提供捷径。
5. 考虑查找的过滤条件未来可能会继续增加，抽象了过滤条件组合，能够很方便添加新的过滤规则

现在来实际的展示一下该工具：

1. 工具主界面  
   
   1. 点击Choose按钮选择需要被搜索的文件夹，比如C:/
   2. 键入File Name Filter来过滤需要被搜索的文件名，比如\*代表匹配所有文件
   3. 键入File Size仅搜索大于等于该Size的文件
   4. 点击Find按钮开始搜索
   5. 点击Stop按钮可以暂停搜索（Stop会变为Continue），点击Continue按钮会继续搜索
   6. 点击Result按钮可以打开窗口显示查找的结果
   7. 点击Clear last time按钮可以清除上次查找的结果，以进行新一轮搜索
2. Result窗口  
   
   1. 如果需要可以点击每列的标题进行排序
   2. 选中需要删除的文件
   3. 点击Delete Selected Files按钮删除被选中的文件
   4. 点击Messages For Search按钮可以打开窗口显示搜索时产生的异常信息
   5. 点击Messages For Deletion按钮可以打开窗口显示删除时产生的异常信息，比如某个文件无法删除