STATEMENT SCRAPER PROJECT SUMMARY

Converting .PDF Bank Statements Into .CSV

The Problems

A client was having an issue with a financial institution which did not allow easy export of transaction information. This faced them with the daunting task of manually copying the dozens of transaction items from each of the several hundreds of .PDF statements.

The Solution

Instead of doing this by hand (and anticipating this to be a recurring future request) I wrote a Python script that when provided a folder full of PDF statements will generate a single .CSV table of all the transaction line items from all said statements. The .CSV collected the following columns/fields:

```
AccountNumber
StatementDate
TransactionDate
TransactionType
TransactionQuantity
TransactionName
TransactionDescription
TransactionAmount
TransactionDebitOrCredit
```

While this isn't fully normalized there was much more to be gained by keeping this as a simpler, single file than attempting anything relational.

Challenges & Risks

The Python script was built largely around the pdfminer library which worked like a charm. Unfortunately, the same could not be said of the statements provided by the financial institution: pdfminer only found text as LTChar objects and numerous iterations were required to sufficiently wrinkle out edge case issues. Reliance on regular expressions made the script itself run relatively slowly; for larger batch jobs optimization would likely be required, but for the several hundred documents involved with this project it was of a lower priority such that it was better to just quickly build something that worked.