

# Deliverable 2 Implementation Notes

This second deliverable focuses on getting the core structure of the spreadsheet working with a minimal, clean design. The goal was to keep things simple while leaving room for later extensions like formula parsing and dependency tracking.

## Overview

I implemented three main domain classes:

### **Cell**

A small object that stores the raw content of a cell as a string. At this point, I'm not evaluating values or distinguishing between numbers, text, or formulas. The cell just holds whatever the user or file provides.

### **CellAddress**

A tiny value object that keeps the row/column position of each cell and provides `equals` / `hashCode` so the map in `Spreadsheet` can index cells consistently.

### **Spreadsheet**

The spreadsheet manages a collection of `Cell` objects. For storage, I chose a `Map<CellAddress, Cell>` so I only keep the cells that have been explicitly assigned and avoid having to maintain large arrays filled mostly with empty entries. This keeps the implementation lightweight and efficient for the current assignment. In the future, this data structure may need to be adapted to support evaluations while keeping performance good, but for now it works well.

The spreadsheet also exposes simple methods to:

- Set the content of a cell
- Get the content of a cell
- Query the current number of rows and columns based on the populated cells

# File I/O

To load and save spreadsheets, I added basic S2V-style operations:

- Load: Read each row, split by ; , and create cells for every entry present in the file.
- Save: Write out the spreadsheet row by row using the same separator rules.

No parsing or evaluation is done; the data is stored exactly as it appears in the file so that round-tripping works for now.

## Code

The full implementation is here:

<https://github.com/henryblu/SOFTENG/tree/main/TheSpreadsheet>

## Directory Structure

So far, the project is organized as follows:

```
TheSpreadsheet/
├── src/
│   └── (Java source files)
├── data/
│   └── (sample S2V input/output files)
└── deliverables.md
```