University of Leicester

How to use Excel efficiently

University of Leicester

Why Excel?

- Spreadsheets are the most flexible form of software available. They may not be the *best* for everything but they can do everything, even if slightly less efficiently for some applications.
- There are costs in maintaining many different software packages, especially if they are used infrequently.
- Excel even has a free built-in programming language *VBA* (used to write *macros* and automate Excel).

University of Leicester

What we are not going to cover

- This module is not a crash course in Excel. It is assumed that you are familiar with the basics of spreadsheets and know how to:
  - Enter text, numbers and formulae and use these to carry out calculations
  - Basic operations such as copy/cut and paste, copy/fill across and down
  - Absolute and relative cell references: $ sign is used to make reference absolute, e.g. A2, $A2, A$2
- If not look them up.

University of Leicester

What are we going to cover?

- Instead, we are going to look at:
  - Ways to improve your use of Excel, so making it more efficient, and therefore more productive
  - Handling data in Excel, since this is the prime requirement for many organisations
  - Use Excel for optimisation
- This is not going to be a series of tutorials but a quick guide to things you ought to know, so you can look them up for yourselves, with assignments to test your understanding

University of Leicester

Spreadsheet design

University of Leicester

Overview

- Objectives
  - Better spreadsheet models
    - More reliable
    - More efficient
    - Easier to check, update & maintain
    - More likely to do the calculations intended
    - Less likely to fail an audit
  - Produced more quickly and easily

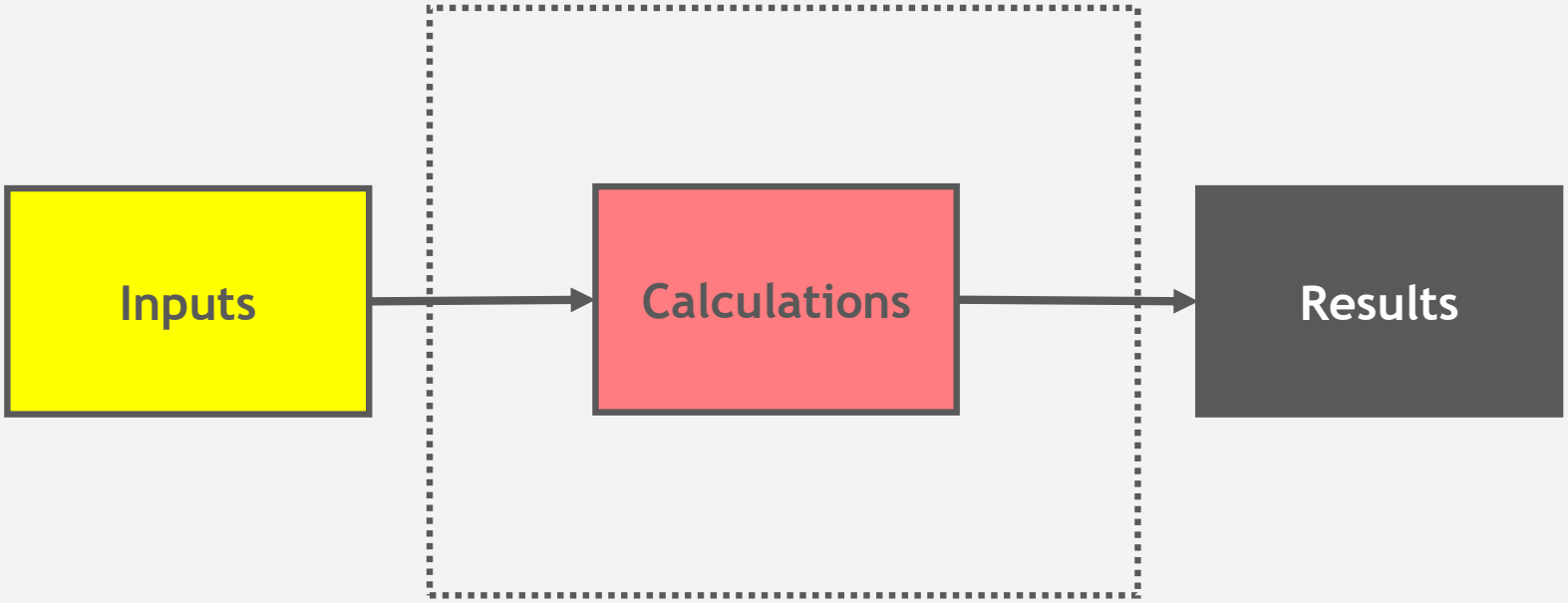
University of Leicester

Before you begin

- Plan ahead:
  - What do you want your spreadsheet to do?
  - Who will use it?
  - What future changes might you want to make?
  - How can you make the layout more efficient?
  - What are the key inputs and outputs?
- A bit of thought at this stage can save you lots of time later on.

University of Leicester

Components of a spreadsheet



University of Leicester

Spreadsheet Style Guide

Separate model functions

Use a clear, consistent layout

Label rows, columns, cells; name ranges

Check validity of inputs & results

Ensure correct use of absolute and relative references

Keep it simple!

Do not use constants in formulas

Only use formulas that can be copied across & down

Avoid hiding rows, columns, sheets

Avoid formulas that link to other workbooks

University of Leicester

When you have finished

- Document your spreadsheet
- Protect cells and worksheets that should not be changed (use a password if necessary)
- Highlight cells that users should change – and make sure they are visible when the spreadsheet is opened
- Give your spreadsheet names that are self-explanatory – add the date if necessary
- Make back-ups

University of Leicester

Examples

Refrigeration costs

Yellow cells are for entering parameters; blue cells are for entering rules on number of units operating

Maximum temperature rise	5	°C
Temperature rise	1.8	°C/h
Refrigeration rate	0.8	°C/h
Peak electricity rate	120	£/h
Off-peak electricity rate	40	£/h
Maximum number of units	3	
Initial temperature (at 06:30)	2.5	°C

Note: for convenience, temperatures are measured as °C above the minimum the pumps can operate at.

Total Cost	**£5,164**
Final temp (°C)	**0.06**

Time	Units on (num)	Temperature (°C)	Rate (£/h)	Cost (£)
6:30	1	2.50	120.00	£2.00
6:31	1	2.52	120.00	£2.00
6:32	1	2.53	120.00	£2.00
6:33	1	2.55	120.00	£2.00
6:34	1	2.57	120.00	£2.00

University of Leicester

The sand and gravel company

The decision matrix:

		C1	C2	C3	C4	C5		
	Q1						0	75
	Q2						0	150
	Q3						0	75
		0	0	0	0	0		
	Totals	100	60	40	75	25		300

The constraints are that each row and column in the decision matrix add to to the total given, and all are positive.

The cost matrix

	C1	C2	C3	C4	C5
Q1	3	2	3	4	1
Q2	4	1	2	4	2
Q3	1	0	5	3	2

The objective function, which we want to *minimise:*

Z = 0