

# CSV and spreadsheets

# Spreadsheets

How do we get data out of spreadsheets? (Like Excel etc.)

- ▶ Export as text
  - ▶ Write our own parser
- ▶ Export as XML
  - ▶ XML parsers in Python
- ▶ Third-party libraries
  - ▶ xlrd
  - ▶ pyExcelerator
  - ▶ Windows COM components
- ▶ Export as “X-separated values”
  - ▶ CSV: comma-separated
  - ▶ TSV: tab-separated

# Format

- ▶ Fields delimited by commas
- ▶ Records end with end-of-line
- ▶ Double quotes escape special characters
- ▶ Flanking whitespace is trimmed
- ▶ First line *may be* field names
- ▶ Several flavours:

```
Year,Make,Model,Notes,Price  
1997,Ford,E350,"ac, abs, moon",3000.00  
1999,Chevy,"Venture Edition",,4900.00  
1996,Jeep,Grand Cherokee,"MUST SELL!  
air, sun roof, loaded",4799.00
```

## Simple example

Return each row as a list:

```
import csv
```

```
in_file = open ('seqdata.csv')
```

```
reader = csv.reader (in_file)
```

```
for row in reader:
```

```
    print row
```

```
['Year', 'Make', 'Model', 'Notes', 'Price']
```

```
['1997', 'Ford', 'E350', 'ac, abs, moon', '3000.00']
```

```
...
```

## More complex example

Print items from each a list:

```
import csv

in_file = open ('seqdata.tab')
reader = csv.reader (in_file, delimiter='\t')

for year, make, model, notes, price in reader:
    print "Record: %s %s %s" % (year, model, price)
```

*Record Year, Model Price*

*Record 1997 Ford 3000.00*

*...*

# Reading API

▶ *import csv*

▶ **csv.reader:**

```
__init__ (self,  
         file_handle,  
         delimiter='',  
         skipinitialspace=False  
        )
```

▶ *reader is iterable*

▶ *No other methods*

## Writing example

Write rows one by one to file:

```
import csv  
  
out_file = open ('dataout.csv', 'w')  
writer = csv.writer (out_file)  
data = ['Year', 'Make', 'Model', 'Notes', 'Price']  
writer.writerow (data)
```

## Writing example 2

Writer takes same options as reader:

```
import csv

out_file = open('dataout.csv', 'w')
writer = csv.writer(out_file, delimiter='\t')
data = [
    ['Year', 'Make', 'Model', 'Notes', 'Price'],
    ['1997', 'Ford', 'E350', 'ac, abs, moon', '3000.00'],
    ['1999', 'Chevy', "Venture Edition", , 4900.00],
]
writer.writerows(data)
```

## Writing fields as dictionaries

`csv.DictWriter (file_handle, field_list, restval ...)` write to file with columns labelled from list. `restval` is what to write if a value isn't provided.

`DictWriter.writerow (dict)` write dictionary contents as row

`DictWriter.writerows (dict_list)` write all dictionaries as rows

```
out_file = open ('dat.csv', 'w')
dwriter = csv.DictWriter (out_file, ['id', 'title', 'data'], restval='')
data = {'id': 'X47', 'title': 'Apis cyt6', 'data': 'ACGT'}
dwriter.writerow (data)
```

## Reading dictionaries

`csv.DictReader (file_handle, field_list, restval ...)` read from file with columns labelled from list. If fields are not provided, they are taken from first line. `restval` is what to write if a value isn't provided.

`DictReader.__iter__ (dict)` reader is iterable and will return row

```
in_file = open ('dat.csv', 'r')
dreader = csv.DictReader (in_file, ['id', 'title', 'data'], restval='')
for row_dict in dreader:
    print row_dict
```

# Gotchas

- ▶ Doesn't support Unicode yet
- ▶ Everything is a string
- ▶ Detect dialects with `csv.Sniffer`
- ▶ Write your own dialect
- ▶ Close file handles

# Resources

- ▶ [Wikipedia on CSV](#)
- ▶ [Howto: the CSV format](#)
- ▶ [Effbot on CSV](#)
- ▶ [Python documentation on the CSV module](#)