Monday, April 2, 2018

Unit 1: Working with Text Files

Reading: *PCfB* Ch 1 (“Getting Set Up”)

- text files vs binary files
- filename extensions
- plain text *vs* rich text
- text editors
- invisible characters

Project	Milestone Exam
Text Files (10 points)	Exam 1 (10 points)

The project and exam for this unit are trivial “hello world” style exercises.

The goal is to help everyone (including the instructors) work through the process of submitting projects and taking milestone exams.

We recommend you try completing the project and taking the exam in this week’s lab (Apr 6).

Types of Data

The two basic kinds of computer files are **binary** files and **text** files

Good examples of binary files are music (MP3, etc) and images (JPEG, GIF, etc)

Image: Stuart Gregory / Getty Images

Each "pixel" is defined by a number

The file is just a sequence of numbers in binary notation

Text Files

The are also two basic kinds of text files

Rich Text, also called Documents

- the file contains detailed formatting information: size, color, font, etc
- a document may also have embedded images
- examples: Microsoft Word documents, Pages documents, …

Plain Text

- no formatting information
- the file is simply a sequence of characters
- old format: ASCII (127 different letters and symbols)
- modern format: Unicode (over 100,000 characters from all languages)

★ Plain text is used for:
scientific data (e.g. gene sequences)
programs (shell scripts, Python programs)

Filename Extensions

We can usually tell what kind of data is in a file by the “extension”

- file names typically end with a period followed by a few other characters
- when you double-click on a file the OS opens an application that works with that type of file
- not fool-proof: you can change the extension at any time

Extension	File Type	Typical Application
`.zip`	binary	compressed archive
`.doc`	binary	MS Word
`.pages`	binary	Pages (macOS)
`.pdf`	binary	“portable document format” for printers
`.xls`	binary	spreadsheet
`.csv`	text	plain text version of a spreadsheet file
`.tsv`	text	tab-separated values (similar to CSV)
`.html`	text	web browser
`.txt`	text	simple documents, e.g. mail messages
`.rtf`	binary	“rich text format”, old MS format still in use
`.fasta`	text	DNA and protein sequence filed
`.gbk`	text	Genbank “annotated sequence” file
`.db`	binary	SQL database
`.py`	text	Python program — you edit this type of file
`.pyc`	binary	compiled Python program — computer runs this

Text Editors

To work with a plain text file (e.g. a Python program or small data file) you will need a **text editor** application

There are lots of them, some free and open source, some are full-featured commercial products

PCfB recommends:

- for Windows 10: Notepad++
- for macOS: TextWrangler
- for Linux: gedit

If you want to explore, look for “program editors”

See: http://alternativeto.net for recommendations

★ Download and install a Text Editor application ASAP

Do Not Use a Document Editor when Programming

It's possible to use Microsoft Word to edit Python programs

- make sure you select the "save as plain text" option when you save the program

But we **strongly** recommend using a text editor instead

- you lose formatting information when a document is exported as plain text
- text editors have lots of useful features specifically for editing programs

★ Use the right tool for the job….

A Note for Mac Users

macOS comes with an application named Text Edit — don't use it

It was designed for Rich Text (.rtf) files

- any new documents you create will be .rtf by default
- comments above about MS Word apply to Text Edit, as well

Text Editing Demo

I created three documents, each containing the same text

- `rocks.pages` was created with Apple’s document formatting application
- `rocks.rtf` is a “rich text” file created with TextEdit
- `rocks.txt` is the plain text version

(1) The “get info” command in the Finder shows each file is a different size

- 161,962 bytes for the Pages doc
- 1,320 bytes for the RTF
- 961 bytes for plain text

(2) The preview gives a hint about what is in each file

Invisible Characters

Plain text files have special characters you should be familiar with

These characters are "invisible" — they appear to be blank space in the application window

- another term is "whitespace" or "whitespace characters"

Space

- the gaps between words are usually space characters
- there can be several spaces next to each other — they're easier to see if you use a fixed width ("monospace") font

Newline

- a newline character marks the end of a line in a document
- the application knows to display the next character at the start of the next line
- add a newline by clicking somewhere and hitting the the Enter (aka Return) key

CRLF

- older files created on a Windows system might have two characters at the end of each line: a "return" character followed by a "line feed"

Tab

- tabs are a relic left over from days when documents were created by typewriters
- typists could move metal tabs left and right, and hit the tab key to move the carriage to the next tab stop

In a text editing application the "tab stops" are preset at every 8th column

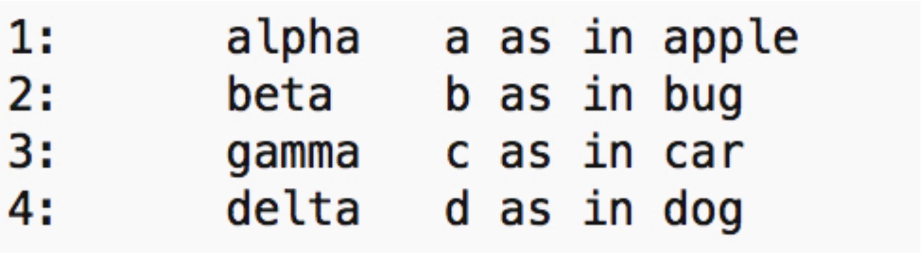
You can control the width using the applications Preferences panel (or sometimes as part of the document window)

Show Invisibles

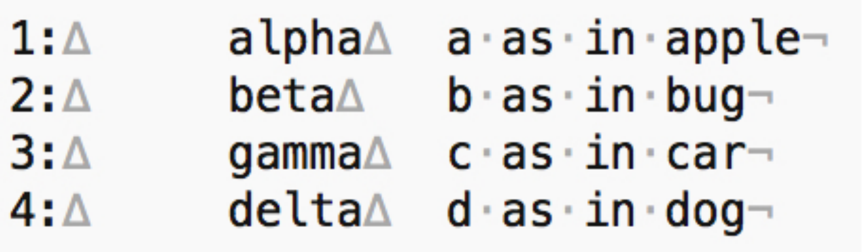
A very useful feature of a text editor app: **show invisibles**

When turned on, the app displays a special symbol wherever a whitespace character is located

In TextWrangler:



```
1:      alpha   a as in apple
2:      beta    b as in bug
3:      gamma   c as in car
4:      delta   d as in dog
```



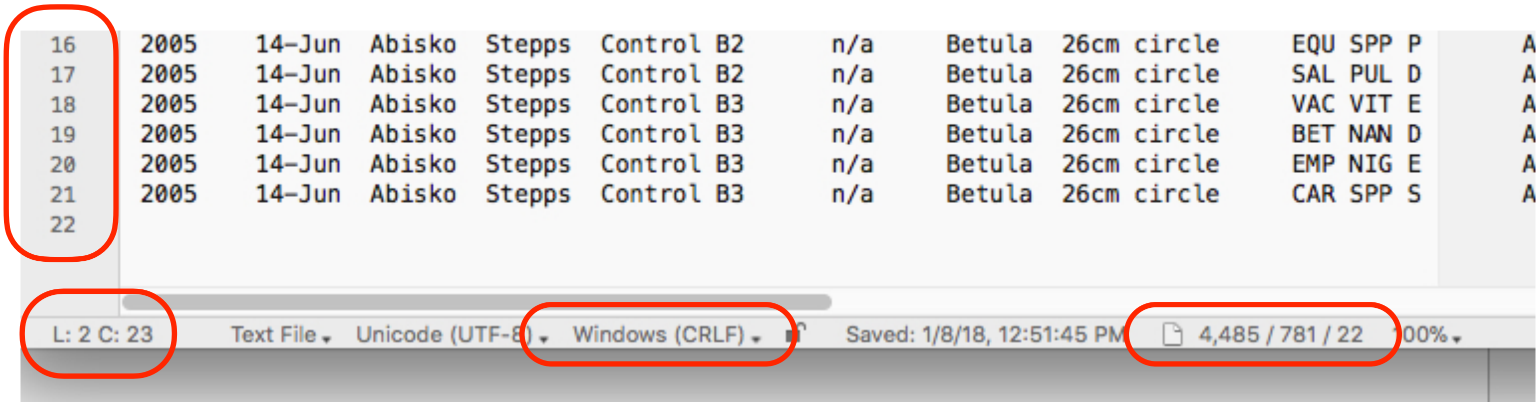
```
1:∆     alpha∆  a·as·in·apple¬
2:∆     beta∆   b·as·in·bug¬
3:∆     gamma∆  c·as·in·car¬
4:∆     delta∆  d·as·in·dog¬
```

Without “show invisibles” (top) we can’t tell if the break between words is from a tab or from multiple consecutive spaces

File Size and Other Information

A text editor also has features that help you learn basic information about a file

- turn on the “show line numbers” option to see a line number displayed before each line
- important: these line numbers are not part of the file — they are just displayed by the application



```
16  2005    14-Jun  Abisko  Stepps  Control B2      n/a     Betula  26cm circle     EQU SPP P       A
17  2005    14-Jun  Abisko  Stepps  Control B2      n/a     Betula  26cm circle     SAL PUL D       A
18  2005    14-Jun  Abisko  Stepps  Control B3      n/a     Betula  26cm circle     VAC VIT E       A
19  2005    14-Jun  Abisko  Stepps  Control B3      n/a     Betula  26cm circle     BET NAN D       A
20  2005    14-Jun  Abisko  Stepps  Control B3      n/a     Betula  26cm circle     EMP NIG E       A
21  2005    14-Jun  Abisko  Stepps  Control B3      n/a     Betula  26cm circle     CAR SPP S       A
22
```

★ This file is 21 lines long, not 22….

★ **Explore on your own:** Create a new blank file with your text editor, start adding characters (including spaces and tabs), learn about the features your application has that gives you information about the file

Search and Replace

The “search” operation is often called “find”

- it works the same way in a text editor as it does in a document formatter

A potential problem: how do you search for a tab? or put a tab character in the “replacement” box?

- if you hit the TAB key when using a GUI it usually means “move the focus to the next field in the display”
- Mac users: if you need to search for a tab (a) click in the search box and (b) hold the option key when hitting the TAB key
- Windows users: ??

Demo:
convert CSV to TSV
convert TSV to CSV

(open `shaver_pre.csv`, save a copy…)