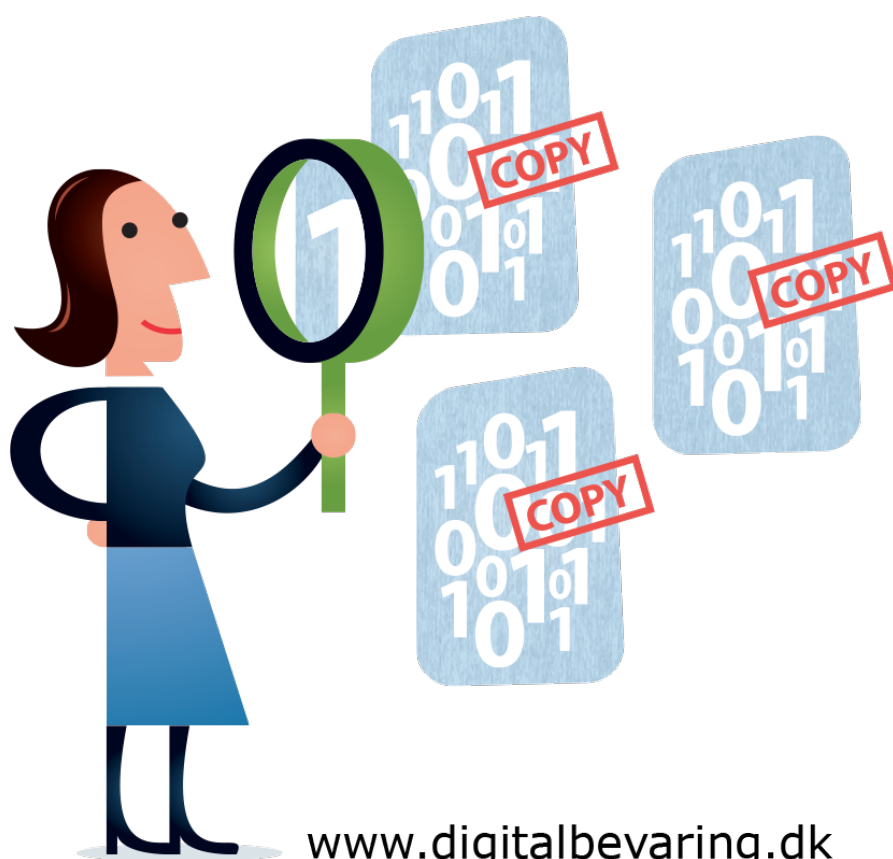


Monitoring the Health of your Digital Collections Exercise booklet

Bodleian Libraries
23 May 2018
Version 1



www.digitalbevaring.dk

Exercise 1: Using Fixity 1.0

Program: Fixity 1.0 by AVP <https://www.weareavp.com/products/fixity/>

Part 1a. Set up a project in Fixity 1.0

1. Set up a New Project (Under File) named *My_Collection*
2. Set it to run **Weekly** at **12:00**
3. **Untick** the box that says "Email only upon warning or failure"
4. Add the directories `~/Desktop/test-files` and in `~/MyDocuments/demo-files`.
5. Under Preferences, set the checksum algorithm to **MD5**.
6. Set the notification email to your own Bodleian email.
7. Once finished, select **Run Now** (Under File).

Once you have run it, you should get an email. Check your email via the web browser to confirm you have received it. The email should provide information about the fixity check and have a more detailed report attached.

1. Confirm you have selected the right checksum algorithm, MD5. **YES / NO**
2. How many files did it check? _____

Part 1b. Run the report a second time.

Select Run Now again and then check your email for a second report in order to answer the following questions:

1. How many files did it confirm? _____

Important: When you are done be sure to delete the project or you will continue to receive emails from it!

Part 2. Checking Fixity 1.0 email reports

1. Go to www.gmail.com to check the email reports from Fixity 1.0.
2. Log in as:
Username: *insert gmail email*
Password: *insert password*

Be sure to have a look through the emails and the attached reports in order to answer the following questions:

Note: Open the attachment as a Google Spreadsheet if prompted, or else it may not display correctly.

1. What do the reports tell you? _____

2. What is missing in the reports? _____

3. Would you keep all of the emails and reports from Fixity 1.0? Why or why not? _____

4. What information do you think should be retained from these reports? Would the email or the report be more important to you? Where would you keep it?

5. How often would you run these reports? _____

a. How often would you run them if your backups were refreshed and overwritten every 14 days?

b. How often would you run them if you kept an archival backup permanently?

Exercise 2: Using the Command Line

This will allow you to see checksum generation in action and see the outputs from it. It will also show you the command line interface and allow you to do some basic work in it.

Program: Mobaxterm <https://mobaxterm.mobatek.net/>

Scripting language: Bash

File directory: ~/Desktop/test-files

Part 1. Run the following code:

```
sha256sum ~/Desktop/test-files/Word-file-with-image.docx
```

6. What did it return? _____
7. Copy the checksum to Notepad, so you can view it later.

Hints:

- If you type only part of the directory word (e.g. Word- for Word-file-with-image.docx) and then press TAB, Mobaxterm will autocomplete (this saves time typing long folders and file names).
- Press the up arrow on your keyboard to bring back previously use commands and then scroll with the left and right arrows to make adjustments to the written commands as necessary.

Part 2. Run the following code:

```
sha256sum ~/Desktop/test-files/Word-file-with-image.docx >>  
~/Desktop/test-files/checksum.txt
```

Note: you will not be able to use TAB autocomplete for checksum.txt because the file does not exist yet.

This will do the same command as part 1, but the calculated checksum will be sent to a **newly created text file** rather than printed on the terminal screen for you to see.

Now run this code to view the file *checksum.txt* on the terminal screen to make sure a checksum has been recorded.

```
less ~/Desktop/test-files/checksum.txt
```

Does the output of the file appear to match the output from part 1? **YES / NO**

Note: hit Q when you are done to quit and return to the command line interface.

Part 3. Run the `sha256sum` command again, only select your own files from the `~/Desktop/test-files/` directory.

Example:

```
sha256sum ~/Desktop/test-files/Roachmotel.pdf >>
~/Desktop/test-files/checksum.txt
```

Run the `sha256sum` command over **at least four (4) files**.

Hints:

- Remember you have now created the `checksum.txt` file so you can use TAB to autocomplete it
- Make sure your code has `>>` in it and not `>` or it will overwrite your text document, not append it.
- If you want to see the files available to you type: `ls ~/Desktop/test-files` – it will return you a list of all the files – remember use TAB to help autofill in the file names
- If you want to check your document in the terminal to make sure all of the checksums are recorded, type: `less ~/Desktop/test-files/checksum.txt`

Part 4. Verifying the checksums

Run the following code:

```
sha256sum -c ~/Desktop/test-files/checksum.txt
```

This checks the file you named in the command, finds the file listed, generates new checksums and checks them against the file. It reports on what it finds in the terminal window.

Did they all verify OK? **YES / NO**

Part 5. Verifying the checksums with a twist

1. Make a change to the document `~/Desktop/test-files/Word-file-with-image.docx` – any change you want.
2. Run the code again:

```
sha256sum -c ~/Desktop/test-files/checksum.txt
```

Did the check run differently? What change did you see? _____

Part 6. (optional) Run the *sha256sum* command over multiple files

First, navigate to the correct directory:

```
cd Desktop/test-files
```

Now type in the following commands:

```
FILES=`find ./`  
for f in $FILES; do sha256sum $f >> sha256.txt 2>/dev/null; done
```

If you want to make sure the code was successful, type the following command:

```
less sha256.txt
```

You should see a lot of checksums and file paths listed – if not try your code again, you have typed something incorrectly.

Hint: in the first line of code the ``` symbol is not an apostrophe, but is actually called a backtick and it is a different symbol. Most Windows keyboards have them in the upper left of the keyboard next to the 1 key.

Now run a *sha256sum* check over *sha256.txt* and check that everything returns okay. If you have time left, edit some of the files and run more checks to see that failures it returns.

Exercise 3. Monitoring file formats

Program: DROID <http://www.nationalarchives.gov.uk/information-management/manage-information/preserving-digital-records/droid/>

Folder of test files: ~/Desktop/test-files

Part 1. Run the report and save it as a CSV

1. Create a profile for the folder ~/Desktop/test-files (Click +)
2. Run a DROID report (Click Start)
3. Export it and save to the Desktop as *droid.csv*
4. Open the CSV and have a look through the report – see the fields provided

Part 2. Create a pivot table in Excel to help answer these questions:

1. How many different file formats do you have? _____
2. What is the most common file format? _____
3. The second most common file format? _____
4. How many different versions of Adobe PDF are there? _____
5. How many are unidentified files are there? _____
6. How many file types were missing an extension? What file types are they?

Part 3. File formats at risk

Think about the Library of Congress' factors for deciding the sustainability of a file format:

- Disclosure
- Adoption
- Transparency
- Self-documentation
- External dependencies
- Impact of patents
- Technical protection mechanisms

Library of Congress Sustainability factors:

<https://www.loc.gov/preservation/digital/formats/sustain/sustain.shtml>

1. Using these principles, are there any file formats that might be at risk? _____
2. List the file formats you think might most be at risk based on these factors.
(You can do additional research, including looking up the PUID in the PRONOM technical registry: <https://www.nationalarchives.gov.uk/PRONOM>):

3. What are some potential digital preservation solutions for these file formats? _____

Part 4. (Optional) Data analysis with Qlik Sense

Sign up for an account on Qlik Sense Cloud Hub (choose the free option): www.qlik.com

Note: you will need access to your work email to activate the account

Click on 'new app' and select 'create app' – give the app a title that means something to you.

Load the *droid.csv* into the application and try generating some charts to familiarise yourself with how Qlik Sense works.

Data analysis in Qlik Sense can run from very easy to more complex, but once you learn it data analysis becomes fairly quick.