

ADS Lab 04 - Scripting

Authors: Pablo Urizar and Vincent Peer

Date: April 09, 2023

The lab5 session is available with all our scripts and directories.

username: lab5

password: fBAW3g&jnH8&

The other session lab6 was used our tests.

Task 2: Create thumbnails

Show dimensions script

```
#!/bin/bash
#
# Description : Prints the name of each picture in the lab04_raw_files directory
# followed by its dimensions. Picture extension should be .png or .jpg. Dimensions
# are width and height.
#
# Authors : Pablo Urizar, Vincent Peer
#
# Date : 09.04.2023

cd lab04_raw_files

for file in *.png *.jpg
do
    echo -e $(identify -format '%f width: %w, height: %h' $file)
done
```

Rename pictures script

```
#!/bin/bash
#
# Description : Renames each picture with png or jpg extention. The resulting name
# is the name followed by the width and the height of this picture and each new
# picture is placed in a new directory named renamed_pictures.
#
# Authors : Pablo Urizar, Vincent Peer
#
# Date : 09.04.2023

cd lab04_raw_files
```

```
# Removes the directory and creates a new one
rm -rf renamed_pictures
mkdir renamed_pictures

# Add dimensions in the picture's name
for picture in *.png *.jpg
do
    picture_ext=${picture##*.}
    name=${picture%[.]*}
    dimension=$(identify -format '%w_%h' $picture)
    filename=${name}_${dimension}.${picture_ext}

    echo "new name : $filename"
    cp $picture renamed_pictures/${filename}
done
```

To prevent any accumulation in the name when running several time this script, it will remove the renamed_pictures directory, rename each picture and save it in the new renamed_pictures directory placed in lab04_raw_files.

Make thumbnails script

```
#!/bin/bash
#
# Description : Converts png and jpg pictures into thumbnails. Converts a pdf file
# into png picture with the first page of the file.
#
# Authors : Pablo Urizar, Vincent Peer
#
# Date : 09.04.2023

path=lab04_raw_files
cd $path

for file in *.png *.jpg *.pdf
do
    ext=${file##*.}
    name=${file%.*}

    if [ "${name##*_}" = "thumb" ]
    then
        echo "$name is already a thumbnail"
    elif [ "$ext" = "pdf" ]
    then
        convert -geometry 300 ${file}[0] ${name}_thumb.png
    else
        convert -geometry 300 $file ${name}_thumb.${ext}
    fi
done
```

To prevent from making thumbnails with thumbnails due to several executions of this script, we add some tests to check if the name contains the "thumb" keyword. In this case, we print that this file is already a thumbnail and do nothing more. Then we manage wheather it is a picture or a pdf file.

Task 3: Generate HTML file

```
#!/bin/bash
#
# Description : Generates a html file based on the given template.
# This script adds thumbnails of pictures and pdf and allows to click on them
# to see the full version.
#
#Authors : Pablo Urizar, Vincent Peer
#
# Date : 09.04.2023

path=raw_files
html_page_rel_path=../page.html

# Writes the beginning of the template into page.html
cat /home/lab5/public_html/lab04_template/template_begin.html > page.html

cd $path

# Adds to the html file the pictures thumbnails
for file in *.png *.jpg
do
    ext=${file##*.}
    name=${file%.*}

    echo "<div class=\"col-md-6 col-xs-12\">" \
        "<a href=\"${path}/${file}\"><img class=\"vignette\" \" \
        \"src=\"raw_files/thumbnails/${name}_thumb.${ext}\" /></a>" \
        "</div>" >> ${html_page_rel_path}
done

# Content and title for pdf files
echo "</div></div>" >> ${html_page_rel_path}
echo "<div class=\"row\" style=\"margin-top: 40px;\">" \
    "<div class=\"col-md-10 col-md-pull-3 col-md-offset-4 article__content\">"
\
    "<div>" \
    "<div><h2>Téléchargez nos brochures</h2></div>" \
    "</div>" \
    "<div class=\"row\">" >> ${html_page_rel_path}

# adds to the html file the pdf thumbmails
for file in *.pdf
do
    name=${file%.*}
```

```

        echo "<div class=\"col-md-6 col-xs-12\">" \
            "<a href=\"${path}/${file}\">" \
            "<img class=\"vignette\" \
src=\"raw_files/thumbnails/${name}_thumb.png\" />" \
            "</a>" \
            "</div>" >> ${html_page_rel_path}

done

echo "</div></div></div>" >> ${html_page_rel_path}

# Writes the end of the template to the html page
cat ../lab04_template/template_end.html >> ${html_page_rel_path}

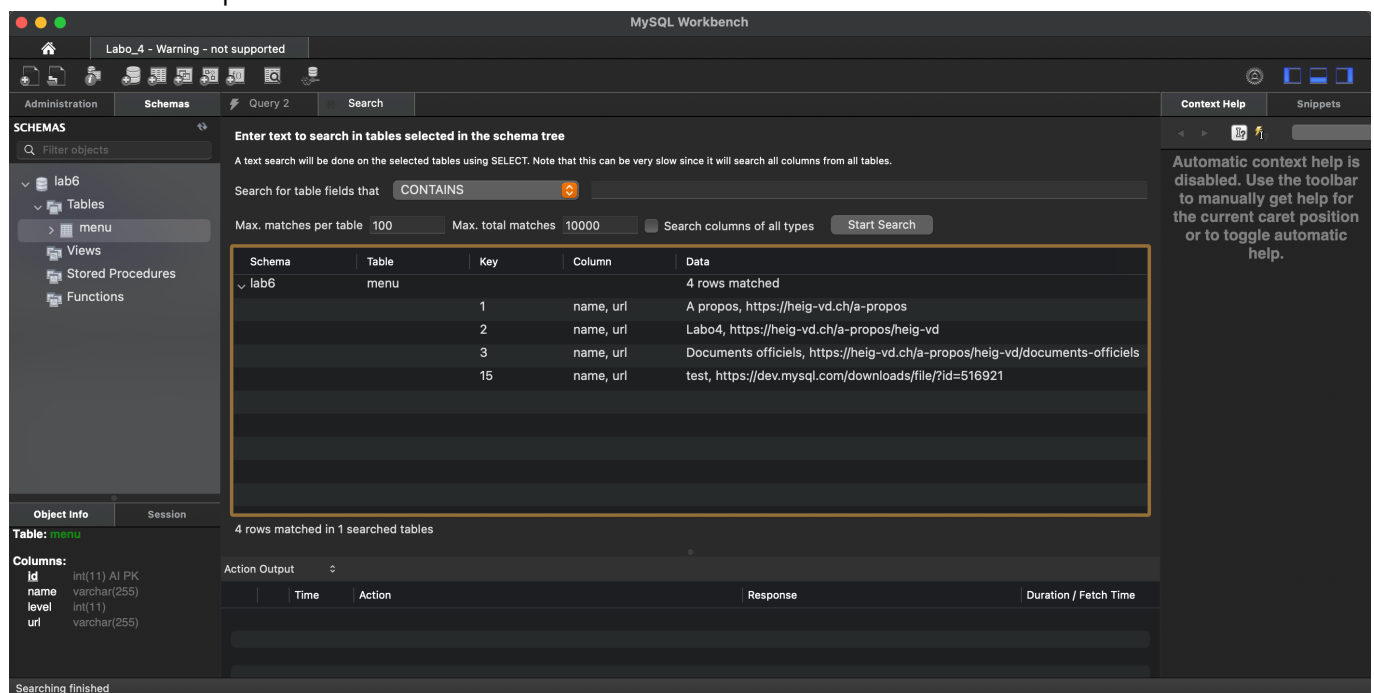
```

Task 4: Use SSH Tunneling

SSH command :

```
ssh ads.iict.ch -L 5454:localhost:3306
```

-L allows to bind port from remote to local.



The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' panel shows the 'lab6' schema selected. The 'Tables' list under 'lab6' shows 'menu' selected. The 'Object Info' panel shows the 'menu' table structure with columns: id (int(11) AI PK), name (varchar(255)), level (int(11)), and url (varchar(255)).

The main panel shows a search for 'menu' in the 'lab6' schema. The search results show 4 rows matched in the 'menu' table. The results are displayed in a table with columns: Schema, Table, Key, Column, and Data.

Schema	Table	Key	Column	Data
lab6	menu			4 rows matched
		1	name, url	A propos, https://heig-vd.ch/a-propos
		2	name, url	Labo4, https://heig-vd.ch/a-propos/heig-vd
		3	name, url	Documents officiels, https://heig-vd.ch/a-propos/heig-vd/documents-officiels
		15	name, url	test, https://dev.mysql.com/downloads/file?id=516921

The 'Action Output' panel at the bottom shows the search results in a table with columns: Time, Action, Response, and Duration / Fetch Time.

We can see some content of the index.php, we added a test with key 15.