T his exercise builds on last week's experience with Linux and MySQL to build a non-trivial database that we'll use in the future.

The purpose of exercise 2 is to practice/revise using SQL, become familiar with using phpMyAdmin and building MySQL database tables.

Task o: Finish exercise 1

Ensure you know how to:

- 1. Start SUSE Linux and <u>http://studentnet</u>
- 2. Edit/create a PHP document in Linux.
- 3. Use a browser to view <u>http://localhost</u> and web pages inside there.

#### Task 1: Import some data manually into a MySQL database

- 1) Open Firefox & load the "phpMyAdmin" page to manipulate your StudentNet database.
- Using phpMyAdmin, create an InnoDB table called country. It has 15 columns and you
  must decide the most appropriate column type to use from either <u>this week's lecture</u>
  <u>slides</u> or <u>the MySQL manual</u>.
  - i) The columns are as follows (in this order):

1) Code	• Always 3 letters & also the table's primary key.
2) Name	At most 52 letters.
3) Continent	<ul> <li>Selected from the set 'Asia', 'Europe', 'North America', 'Africa', 'Oceania', 'Antarctica', 'South America'.</li> </ul>
4) Region	At most 26 letters.
5) SurfaceArea	<ul> <li>Floating point number with max 10 digits and 2 decimal placess.</li> </ul>
6) IndepYear	• An integer year value, such as 1999.
7) Population	An integer with max 11 digits
8) LifeExpectancy	• A floating point number with max 3 digits and 1 decimal place.
9) GNP	• A floating point number with max 10 digits and 2
10) GNPOld	decimal places.
11) LocalName	• At most 4E lattors
12) GovernmentForm	• At most 45 letters.
13) HeadOfState	At most 60 letters.
14) Capital	<ul> <li>An 11 digit integer. (Becomes a <i>foreign key</i> in Task 3).</li> </ul>
15) Code2	2 characters

- Instead of point-and-click in phpMyAdmin you could build the necessary "CREATE" SQL command in the textbox or using an editor (then copy/paste).
- ii) Make sure you verify the table structure in phpMyAdmin or with MySQL's **DESCRIBE country;**

- 3) Import data into **country** from this <u>country.csv file</u> using phpMyAdmin's "Import" tab.
  - Glance through the data CSV files are common formats although they lack "meta data" that explains what the columns *mean*.
- 4) Check it!
  - If you enter "SELECT COUNT(\*) FROM country;" you should get 239.
  - "SELECT SUM(GNP) FROM country;" the total GNP is 29354907.9 (SUM is an aggregate function & there are many in MySQL.)

*Congratulations!* You have just imported one table from the Finnish world countries database <grin> (MySQL AB used to use it as their example database...)

- 5) Try some queries in the phpMyAdmin SQL tab:
  - i) List the names of the 18 countries that became independent in 1991.
  - ii) List the 5 countries in Southern Europe with population less than 1 000 000.
  - iii) What's the *continent* that contains the *country* with the maximum life expectancy? How about the minimum?
  - iv) What's the *continent* with the best *average* life expectancy?You could use an "aggregate function" with a GROUP BY clause in MySQL look in the manual?
    - Kudos to the first person to post the right answers to the StudySpace discussion forum <grin>

### Task 2: Download and install phpMyAdmin on Linux

PHPMyAdmin is an open source web-based administration tool for PHP and MySQL and you can find it at <u>http://www.phpmyadmin.net</u>.

- 1) Download a copy of the *stable* distribution from <u>http://www.phpmyadmin.net</u>.
  - Once the download starts you should be prompted to open or save as usual for Firefox – in KDE-based Linux the "File Roller" application makes handling compressed files easy.

篇

1949

- ii) Extract the archive file into a subdirectory inside the "public\_html" directory in your home directory as follows:
  - *E.g.* click the "Extract" button.
  - Browse to "Home" then "public\_html".

Extract in folder: Dublic html

- Click "Extract" to do the job.
- Next, to make remembering the URL easier, on the desktop, browse to "Home" then find "public\_html" and rename the "phpMyAdmin..." folder to just "phpMyAdmin" or even "pma"!
- Before using phpMyAdmin yourself you really should familiarise yourself with the phpMyAdmin instructions.
- Configure PHPMyAdmin to use your database on StudentNet (*e.g.* see online doc's):
- Type angres . Folder **hingk** 0001 Fold E Felloa Extract in Isider: 🛅 ma \$ Fles Actions · All files · Overgets extraing the Do not eghact older files O Elec-Ques destruction folder after extract C Heb Extract M Cancat 63 B 1 + Eliku13043 public ht Places ¥ C Search mhu13043 Desktop File Syste Network Servers ma<sup>e</sup> selected icontaining 97 items)
  - i) It's simple to create "config.inc.php" by making a copy of the "config.sample.inc.php" file.
    - You can do this by browsing to the "pma" folder from your desktop's "home" folder and then double-clicking on the sample file, choosing to open it in "gedit" and saving it as "config.inc.php"

Considered and a reaching	4	Hamman and Anna Anna Anna Anna Anna Anna Ann	- II A
In Pa fee beers Door	Deserve St	*	
0	8 9		
il contradiction +			
c) physical and a second s	And and a state of the local diversion of the	1	1
	A BORT PLAN		
As its bo wat that set	S.m.	a checking and photo	
/* 134 contio estault			*7
// use: sugandtab as-4	Strategilter	DetellyAerm3.73.49	55
And Anno and Annotation and A	Character Code	g Career Loode 6/07-86 (*	
* phpMyAdmin defmalt	1 desertes	alar hillers ED.	pour
* enerig.inc.php			
7 ATI disections are		# Cent Gase	
	-		
100			
tour prosyders out.			
	i siste	the state of the s	
		- 14 L	CHI NS

- ii) Add the <u>appropriate information</u> to the configuration file. A minimum working set is as follows:
  - You have no "root" access to StudentNet's MySQL so leave 'controluser' and 'controlpass' empty.
  - \$cfg['blowfish\_secret'] should be some random text in the quotes. (It's a random seed used with Blowfish encryption.)
  - \$ \$cfg['Servers'][\$i]['host'] should be 'studentnet'
  - **\$cfg['Servers'][\$i]['auth\_type']** offers a choice of 3 options:
    - (1) '**cookie**' gives a login page and *requires* a Blowfish passphrase for key generation (an earlier setting any text will do.)
    - (2) 'http' will use HTTP Basic Authentication to prompt you for the user name & password when you use PHPMyAdmin.
    - (3) '**config**' means you must permanently add the user/password for ftemysql to the PHP script, as follows
      - \$cfg['Servers'][\$i]['user'] and \$cfg['Servers'][\$i]['password']
      - Which do you think is the most sensible option for
        - (i) Standalone PCs?
        - (ii) Public servers like StudentNet?
- iii) To verify that it is working, open the relevant URL on the PC, e.g.

http://localhost/~linux/phpMy Admin/ You should get a framed page that allows you to select the **country** table you created earlier...

4) If you are not in "single database" mode, select your database in the left frame and wait for the righthand frame to reload so it looks something like this:





i) Browse the **country** table, either by clicking on the 🔲 icon or selecting the table name in the left-hand frame.

ii) The table's browse page looks something like this:

Server: 😭 localhost 🕨 Database: 🛔	🗊 world 🕨 Table: 🏢	country		
Structure Browse	🐺 SQL 🔎 Search	📑 İnsert 📑 Exp	ort 🕺 Operations	Empty 🔀 Drop
Showing rows 0 - 29 (239 total, Que	ery took 0.0049 sec)			
SQL-query: SELECT * FROM 'country' LIMIT 0, 30				
[Edit] [Explain SQL] [Create PHF	P Code] [Refresh]			
Show: 30 row in horizontal	w(s) starting from record #	# 30 aders after 100 cells	> >> Page n	umber: 1 🗸
Sort by key: None	Go			
$\leftarrow \top \rightarrow$ Code Name C	Continent Region	SurfaceArea IndepYear	Population LifeExpectancy	GNP GNPOR
🗌 🎤 🗙 AFG Afghanistan A	sia Southern and Central Asia	652090.00 1919	22720000 45.9	5976.00 0.

• *E.g.* you would click the pencil **2** to edit a particular row.

Now let's setup the other two tables in the world database...

### Task 3: Create and populate the **city** table

1) Go back to your database view and create a table called **city** with 5 fields

Name:	Number of fields:
	Go

2) Its fields are

1) ID	• An integer field with at most 11 digits that is also the primary key.
2) Name	35 letters
3) CountryCode	• 3 letters, a foreign key that references country's code field.
4) District	• 20 letters.
5) Population	• An integer field with at most 11 digits.

Make sure you find the relevant drop-downs and buttons to assign each field's properties in phpMyAdmin...

◆ By using InnoDB tables we can arrange for the foreign key to be correctly associated with the **country** table ☺

3) On the subsequent page you should get an HTML table showing the **city** table:

		107		,			0					
	Field	Туре	Attributes	Null	Default	Extra			Act	ion		
	<u>ID</u>						1	$\mathbf{x}$	1	1	U	T
	Name						1	$\mathbf{x}$	1	1	U	T
	CountryCode						1	$\mathbf{x}$	1	P	U	T
	District						1	$\mathbf{x}$	1	1	U	T
	Population						1	$\mathbf{x}$	1	1	:U	T

- 4) Repeat the import process (from earlier) using this city.csv file.
  - i) What you should end up with is a table with 4079 rows whose "browse" page starts something like this:

Server: Blocalhost > Database	e: 📠 world 🕨	Table: 🏢 city			
Structure Browse	📲 SQL 🔊	Search 📑 🕯 Ir	nsert Export	% Operations	📅 Empty 🛛 🐹 Drop
Showing rows 0 - 29 (4079 tot	al, Query took	0.0019 sec)			
SQL-query: SELECT * FROM 'city' LIMIT 0 , 30					
[Edit] [Explain SQL] [Creat	e PHP Code] [F	Refresh]			
Show : 30	row(s) starting	from record # 3	30 s after 100 cell	s >>>>	Page number: 1
Sort by key: None	V Go				
$\leftarrow   \rightarrow   D \qquad \text{Name}$	CountryCode	District	Population		
I Kabul	AFG	Kabol	1780000		
🗌 🖉 🗙 2 Qandahar	AFG	Qandahar	237500		

#### Task 4: Create and populate the countrylanguage table

1) Final table! This time we want to create **countrylanguage** with these fields:

1) CountryCode	• 3 letters, a foreign key that references country's code field that is also part of the composite primary key.
2) Language	• 30 letters that is the other part of the composite primary key.
3) IsOfficial	<ul> <li>Boolean, simulated by a field that takes one of two values: "T" or "F" only.</li> </ul>
4) Percentage	<ul> <li>A floating point number representing a percentage with 3 digits and 1 decimal place.</li> </ul>

2) However there is a quick and easy way to do this in one step ;-) just import this "SQL dump" backup file: http://dtaffaet.kingstep.ac.uk/@ku12042/MabDB/ov/weak02/countrylanguage.cgl

<u>http://staffnet.kingston.ac.uk/~ku13043/WebDB/ex/week02/countrylanguage.sql</u> (You should browse through the file to see the structure.)

- 3) Finally, ensure the referential integrity constraints are in place:
  - i) There's a nice simple command to make any table InnoDB type, e.g.:
    - ALTER TABLE countrylanguage TYPE=InnoDB;

(There's a straightforward way in the phpMyAdmin GUI too ...)

- ii) If you have not already done-so, set-up the appropriate primary keys.
  - E.g. from the "Structure" view for country highlight the key and press the button.
     Server: fremysql + @ Detabase: ku12043 + @ Table: country "InnoD8 free: 11264 kB"
     Browse ☆ Structure ☆ SQL > Search ≩-Insert ☆ Export ☆ Operations ☆ Empty ☆ Drop
     Field Type Attributes Null Default Extra Action
     Code char(3)
     No
     N
- iii) Look at the foreign keys which table "owns" the key and which table "uses" a copy of the key? The parent should be allowed to DELETE or UPDATE a key field and have that deletion *cascade* into the child table, *not* the other way around and **not** in both directions either! (Circular constraints will prevent all UPDATE/INSERT/DELETE operations.)
- iv) From the structure tab in the relevant tables, open the "Relation view" and link the keys as appropriate.
  - If you don't know what the "ON DELETE" and "ON UPDATE" options mean, <u>look</u> them up on mysql.com
- v) Test it ;-> (maybe do "Task 5" 1<sup>st</sup>?!) Delete a row in a "parent" table and see if the deletion cascades as it should.

### Task 5: Backup your precious database tables in one file

Thankfully now that the data exist in MySQL it's easy to reduce the database creation and population steps to one easy task by exporting the table data:

Thursday,

- 1) Open phpMyAdmin and locate the "export" tab from *your* database :
  - i) Select all of your tables
  - Select "Add DROP TABLE" and "Add IF NOT EXISTS" so the backup is easy to restore.

	152. sphere G ANS curves convent the basile (in spits lines) Distribute expert in a basiles (in spits lines) Distribute expert in a basilese	
NE. NE. LafeA Microsoft Event 2008	C Sectors C Aut 2007 Mala II Aut 2017 Solid II Aut Auto, Scientific II Colors and auto, Scientific Veloc II Colors and and Suit Investelli Instances Mid Insciences II Colors (Automotive Automotive II Colors (Automotive Automotive)	
Otvier Hit Excel Otvier Hit Excel	Image: Strategy of the second to a second	
	Equit spectrum -	

- "Extended inserts" shortens it slightly.
- iii) Choose a file format (if not, you get a big SQL textarea that you can copy/paste to a file if necessary ...)
- iv) Click "Go" and save the file on your H: drive.
- 2) Examine the *DBname*.sql file (open the ZIP and unpack if necessary)
  - i) Open it in an editor ("gedit" or similar) & examine the structure, it's educational...
  - ii) Make sure it contains data and CREATE definitions for all the tables you exported.
    - To restore everything you could now (provided the textarea is big enough!) copy/paste the file's contents into either SQL text area or upload the commands as-is.

#### <huge sigh of relief>

That's it for today ... if you have time you could continue to explore <u>phpMyAdmin</u>, the <u>PHP</u> <u>manual</u> or the <u>MySQL documentation</u>.