Module ICT3715

INFORMATION AND COMMUNICATION TECHNOLOGY PROJECT

STUDENT NUMBER (Student completes)									
5	8	0	3	4	0	9	9		

IDENTITY NUMBER (Student completes)												
9	6	0	4	5	5	4	4	0	8	7		

No handwritten assignments will be accepted.

INSTRUCTIONS:

Complete this Front Page (page 1)

Complete the Plagiarism Pledge (page 2). Your Assessment (assignment) will not be assessed without this.

After you have completed the front page with your information, the plagiarism pledge, and Assessment (Assignment) 1 with Section A to C, save the document as a PDF document.

You must save your Assessment (Assignment) 1 as a PDF document, or it will not be assessed.

Keep a copy of the original should there be problem with the upload.

PLAGIARISM PLEDGE BY THE STUDENT

1. I have read Unisa’s plagiarism policy.
2. I understand Unisa’s plagiarism policy.
3. I agree to abide by Unisa’s plagiarism policy.
4. I have read the direct copying, plagiarism, and “patch-writing” document.
5. I understand what direct copying, plagiarism, and “patch-writing” is.
6. I undertake to avoid copying directly, plagiarism and patch writing.
7. All academic work, written or otherwise, that I submit is expected to be the result of my own skill and labour.
8. I understand that, if I am guilty of the infringement of breach of copyright/plagiarism or unethical practice, I will be subject to the applicable disciplinary code as determined by Unisa.
9. The marker has the right to refuse to assess the assignment and the system if plagiarism is detected.
10. [Here you can add your references that you have used e.g., information taken from the Internet]

Student name and Surname: RESSICK KEAMOGETSOE MASHISHI

Student number: 58034099

R.K MASHISHI 06/05/2024

Student signature: Date:

Table of Contents

1 Assessment 1

1.1 Section A [4]

1.1.1 Programming Languages (3)

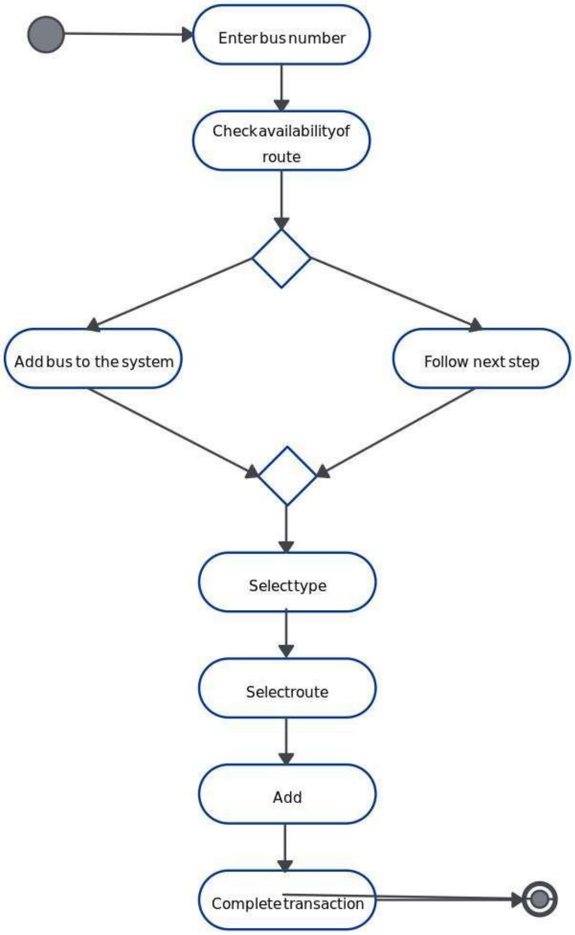
- Programming Languages are; html and CSS with a client side language JAVASCRIPT to make a visual appealing web form, and the information submitted in the placeholders needs to know where to go. I will need a web server with an SQL database and PHP programming language to process the record from the web form and save it in my database.

1.1.2 Database (1)

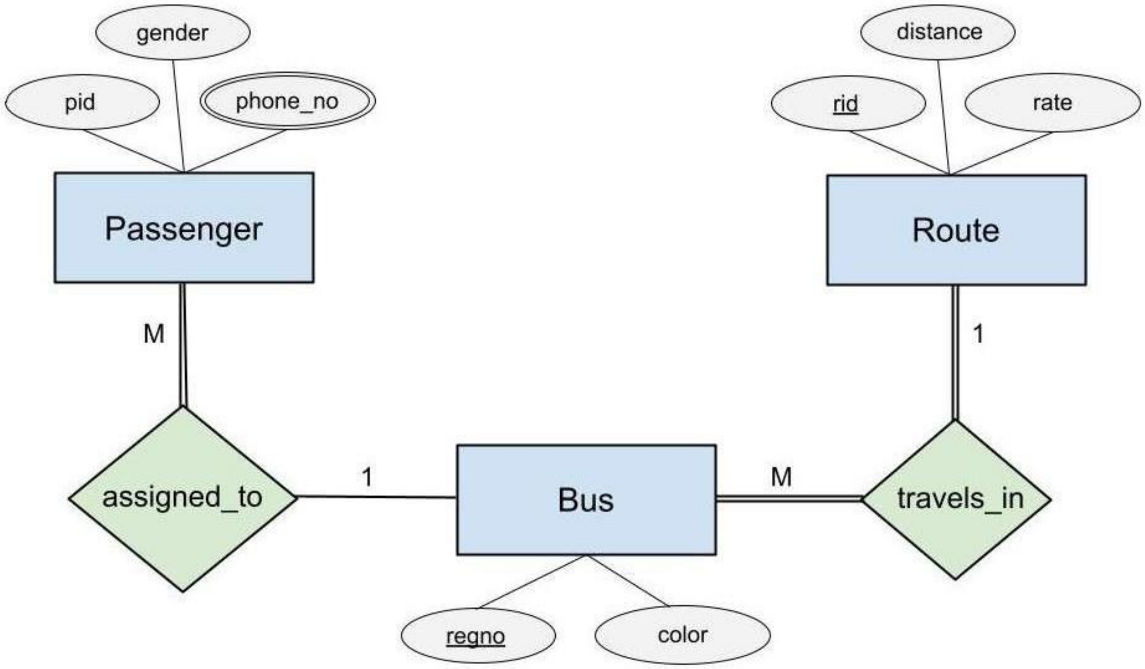
- I am going to use MYSQL for database management system, because a database housed in a DBMS, which provides the functionality to create and work with a database.

1.2 Section B [28]

1.2.1 Activity Diagram (16)



1.2.2 ERD Diagram (12)



1.3 Section C (Backup and Recovery for the Database and Programming code) [4]

1.3.1 Backup and Recovery Software for the Database (2)

- XAMPP includes phpMyAdmin, an open source, browser-based tool for managing MYSQL database servers.
- SQL Backup Master. Back up your SQL server database to any number of popular cloud storage services.

1.3.2 Backup and Recovery process for the Programming code and your Portfolio (assessments) (2)

1. BACK-UP PROCESS(using XAMPP)

- Browse to http://localhost/phpMyAdmin or http://127.0.0.1/phpMyAdmin. If required, log in using your database access credentials. On a fresh XAMPP installation without any changes, you can log in as root with a blank password.
- Select the database to be backed-up from the list in the left navigation pane.

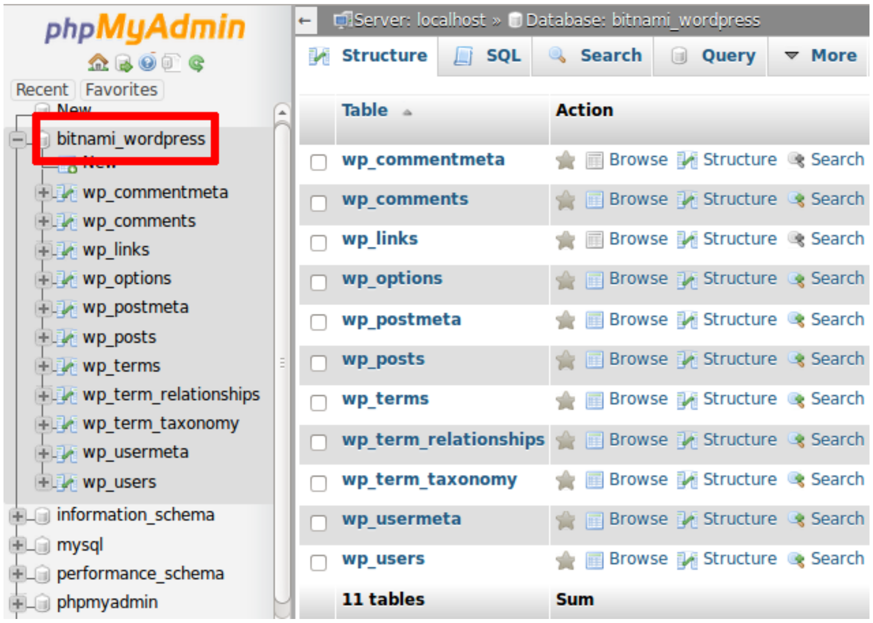
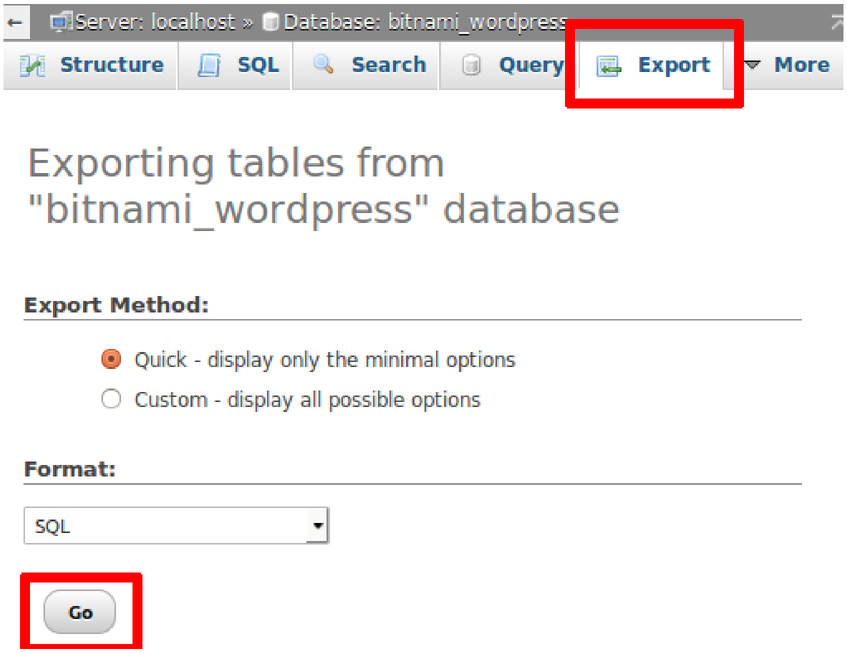


Table	Action
wp_commentmeta	Browse Structure Search
wp_comments	Browse Structure Search
wp_links	Browse Structure Search
wp_options	Browse Structure Search
wp_postmeta	Browse Structure Search
wp_posts	Browse Structure Search
wp_terms	Browse Structure Search
wp_term_relationships	Browse Structure Search
wp_term_taxonomy	Browse Structure Search
wp_usermeta	Browse Structure Search
wp_users	Browse Structure Search
11 tables	Sum

- Select the "Export" command in the top navigation bar.
- On the resulting page, select "Quick" as the export method and "SQL" as the output format. Click "Go" to proceed.



Exporting tables from "bitnami_wordpress" database

Export Method:

● Quick - display only the minimal options

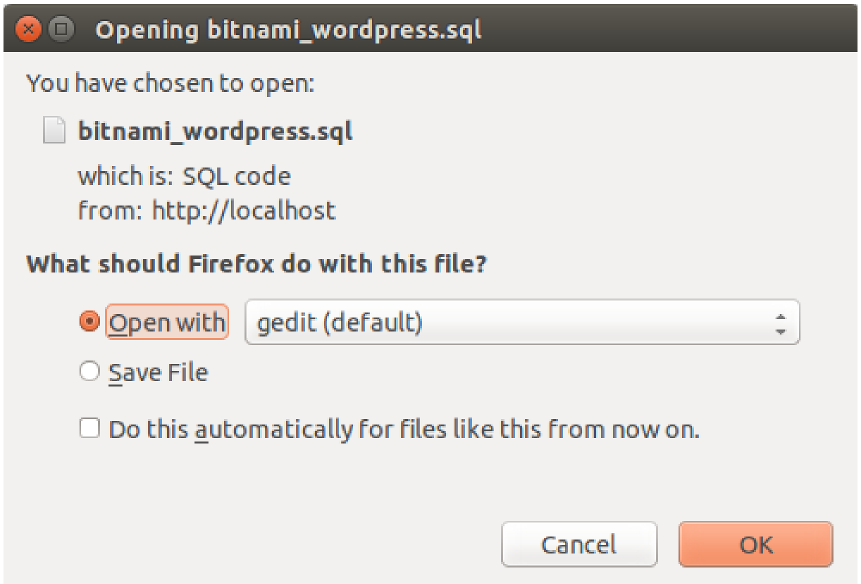
○ Custom - display all possible options

Format:

SQL

Go

- phpMyAdmin will export the database to a text file as a series of SQL statements. Once done, the browser will prompt you to download it to the desktop. This text file is your backup, so store it safely!



Opening bitnami_wordpress.sql

You have chosen to open:

bitnami_wordpress.sql

which is: SQL code
from: http://localhost

What should Firefox do with this file?

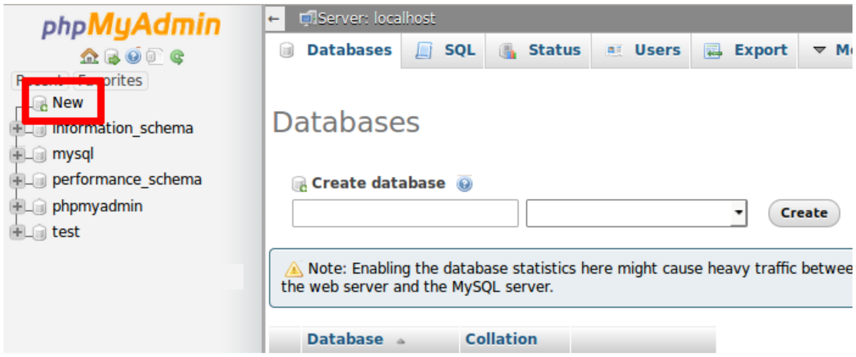
● Open with gedit (default)

○ Save File

□ Do this automatically for files like this from now on.

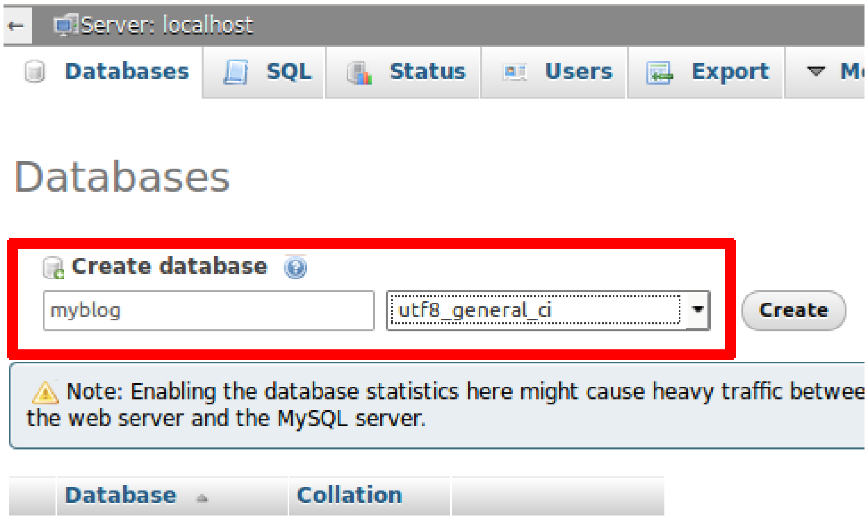
2. RECOVERY-PROCESS

- Browse to http://localhost/phpMyAdmin or http://127.0.0.1/phpMyAdmin. If required, log in using your database access credentials.
- Select the "New" command in the left navigation pane.



Note: Enabling the database statistics here might cause heavy traffic betwee the web server and the MySQL server.

- On the resulting page, enter a name for the new database (in this case, myblog). Select the collation "utf8_general_ci". Click "Create" to create the database.

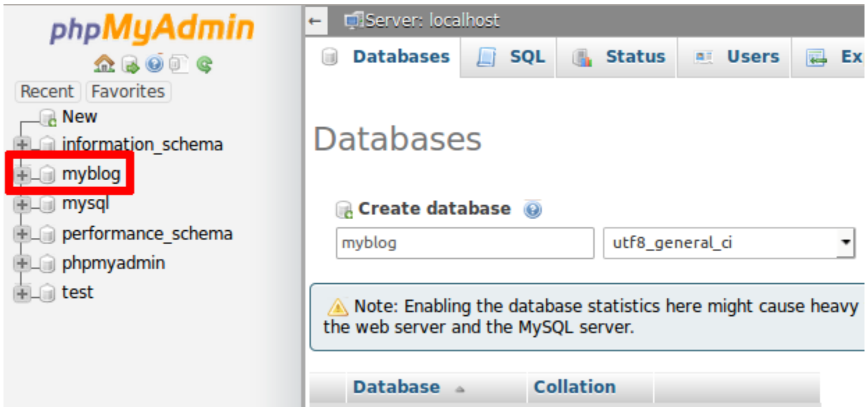


myblog

utf8_general_ci

Note: Enabling the database statistics here might cause heavy traffic betwee the web server and the MySQL server.

- Once the database is been created, it will appear in the left navigation pane.

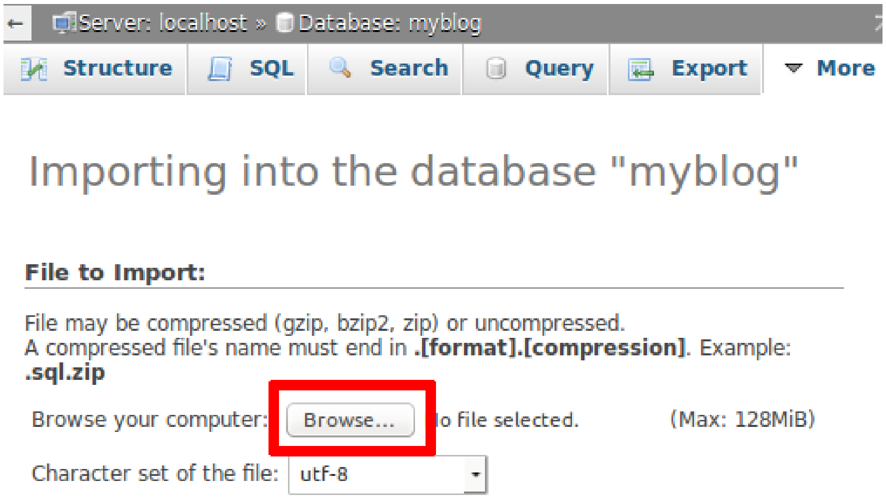


myblog

utf8_general_ci

Note: Enabling the database statistics here might cause heavy the web server and the MySQL server.

- Select the new database from the left navigation pane. In the resulting page, select the "Import" command in the top navigation bar.
- Click the "Browse..."button and select the backup file created earlier. Click "Go" to proceed.



Importing into the database "myblog"

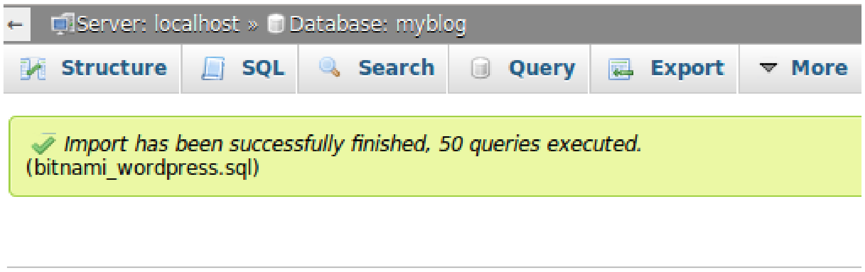
File to Import:

File may be compressed (gzip, bzip2, zip) or uncompressed.
A compressed file's name must end in **.[format].[compression]**. Example: **.sql.zip**

Browse your computer: Browse... lo file selected. (Max: 128MiB)

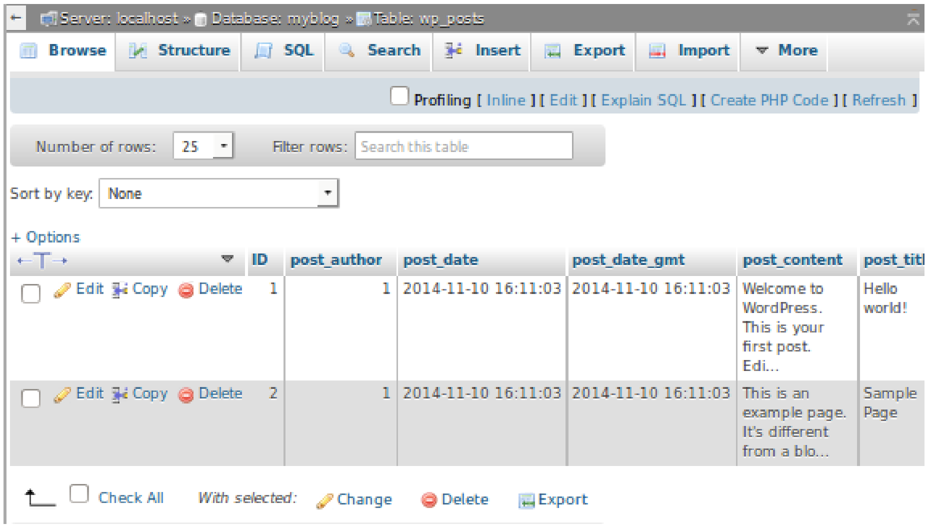
Character set of the file: utf-8

- phpMyAdmin will import the data from the backup file. Once complete, you will see a screen indicating the status of the import.



Import has been successfully finished, 50 queries executed. (bitnami_wordpress.sql)

- If you browse the contents of the database, you should now see your original data.


Server: localhost » Database: myblog » Table: wp_posts
Browse
Structure
SQL
Search
Insert
Export
Import
More
Profiling [Inline] [Edit] [Explain SQL] [Create PHP Code] [Refresh]
+ Options
Check All
With selected:
Change
Delete
Export

Number of rows: 25 Filter rows: Search this table

Sort by key: None

	ID	post_author	post_date	post_date_gmt	post_content	post_titl
□ Edit Copy Delete	1	1	2014-11-10 16:11:03	2014-11-10 16:11:03	Welcome to WordPress. This is your first post. Edi...	Hello world!
□ Edit Copy Delete	2	1	2014-11-10 16:11:03	2014-11-10 16:11:03	This is an example page. It's different from a blo...	Sample Page

USING COMMAND-LINE TOOLS

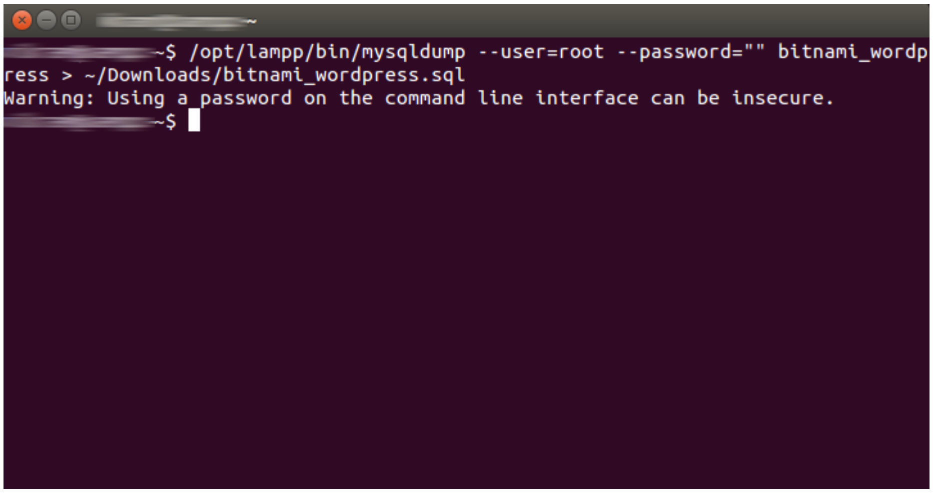
To back-up a MySQL database from the command line, follow these steps:

- Ensure that the MySQL server is running.

- Open a new Linux terminal.

- Use the command below to export the contents of the selected database. In this example,we're backing up the WordPress database, which is named bitnami_wordpress, to a file named bitnami wordpress.sql. This text file is your backup, so store it safely!

```
/opt/lampp/bin/mysqldump --user=root --password="" bitnami_wordpress > bitnami_wordpress.sql
```



```
[redacted]~$ /opt/lampp/bin/mysqldump --user=root --password="" bitnami_wordp
ress > ~/Downloads/bitnami_wordpress.sql
Warning: Using a password on the command line interface can be insecure.
[redacted]~$ 
```

At a later point, you may wish to restore the database. To restore the data to a fresh MySQL database from the command line, follow these steps:

- Ensure that the MySQL server is running.
- Open a new Linux terminal.
- Use the mysql client to create a new, empty database to hold your data. In this example, the new database is named myblog.

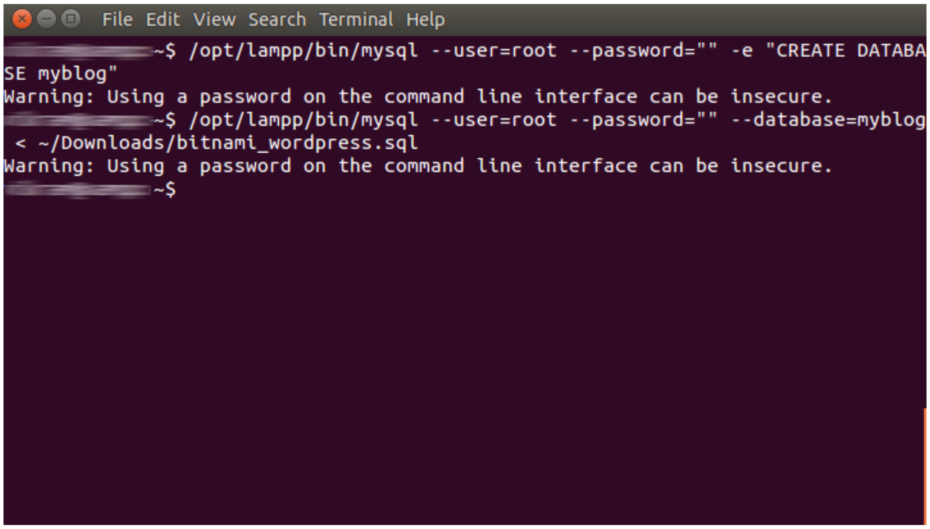
  /opt/lampp/bin/mysql --user=root --password="" -e "CREATE DATABASE myblog"

  Remember to use the correct database access credentials in the command. On a fresh XAMPP installation without any changes, you can usually log in as root with a blank password.

- Use the mysql client to import the contents of the backup file into the new database.

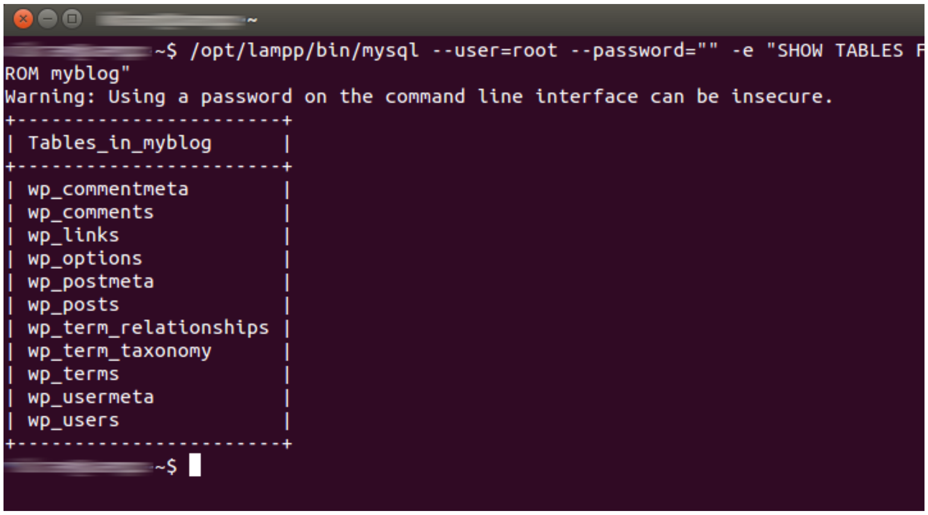
  /opt/lampp/bin/mysql --user=root --password="" --database=myblog < bitnami_wordpress.sql

  The command-line client will now import the data from the backup file.



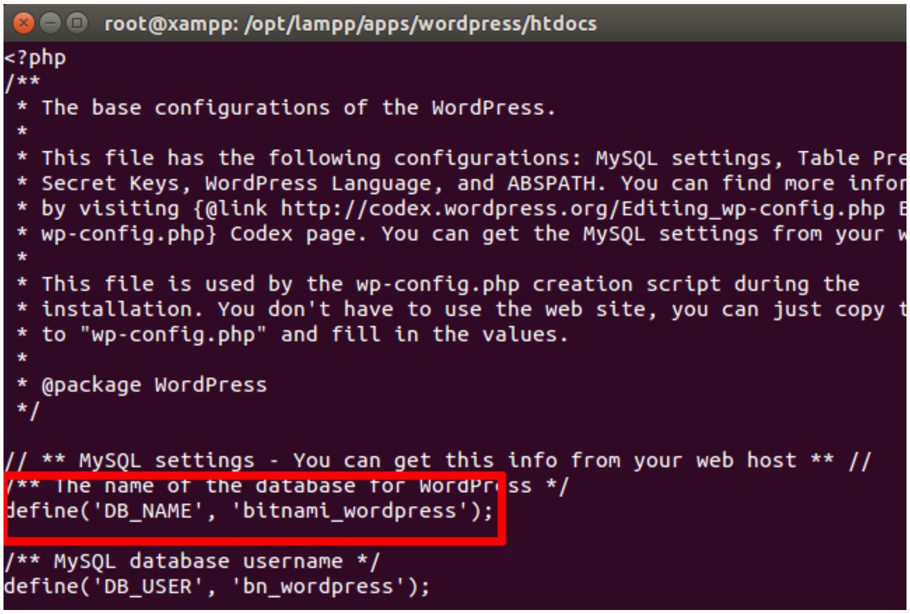
```
[redacted]~$ /opt/lampp/bin/mysql --user=root --password="" -e "CREATE DATABA
SE myblog"
Warning: Using a password on the command line interface can be insecure.
[redacted]~$ /opt/lampp/bin/mysql --user=root --password="" --database=myblog
 < ~/Downloads/bitnami_wordpress.sql
Warning: Using a password on the command line interface can be insecure.
[redacted]~$
```

If you browse the contents of the database, you should now see your original data.



```
[redacted]~$ /opt/lampp/bin/mysql --user=root --password="" -e "SHOW TABLES F
ROM myblog"
Warning: Using a password on the command line interface can be insecure.
+-----------------------+
| Tables_in_myblog      |
+-----------------------+
| wp_commentmeta        |
| wp_comments           |
| wp_links              |
| wp_options            |
| wp_postmeta           |
| wp_posts              |
| wp_term_relationships |
| wp_term_taxonomy      |
| wp_terms              |
| wp_usermeta           |
| wp_users              |
+-----------------------+
[redacted]~$ 
```

You should now update your application's configuration and point it to your newly-created database. For example, with the Bitnami WordPress module, you will need to edit the wp-config.php file in the apps/wordpress/htdocs/ subdirectory of your XAMPP installation directory. Within this file, you will need to update the DB_NAME constant to use the new database name.



```
<?php
/**
 * The base configurations of the WordPress.
 *
 * This file has the following configurations: MySQL settings, Table Pre
 * Secret Keys, WordPress Language, and ABSPATH. You can find more infor
 * by visiting {@link http://codex.wordpress.org/Editing_wp-config.php E
 * wp-config.php} Codex page. You can get the MySQL settings from your w
 *
 * This file is used by the wp-config.php creation script during the
 * installation. You don't have to use the web site, you can just copy t
 * to "wp-config.php" and fill in the values.
 *
 * @package WordPress
 */

// ** MySQL settings - You can get this info from your web host ** //
/** The name of the database for WordPress */
define('DB_NAME', 'bitnami_wordpress');

/** MySQL database username */
define('DB_USER', 'bn_wordpress');
```