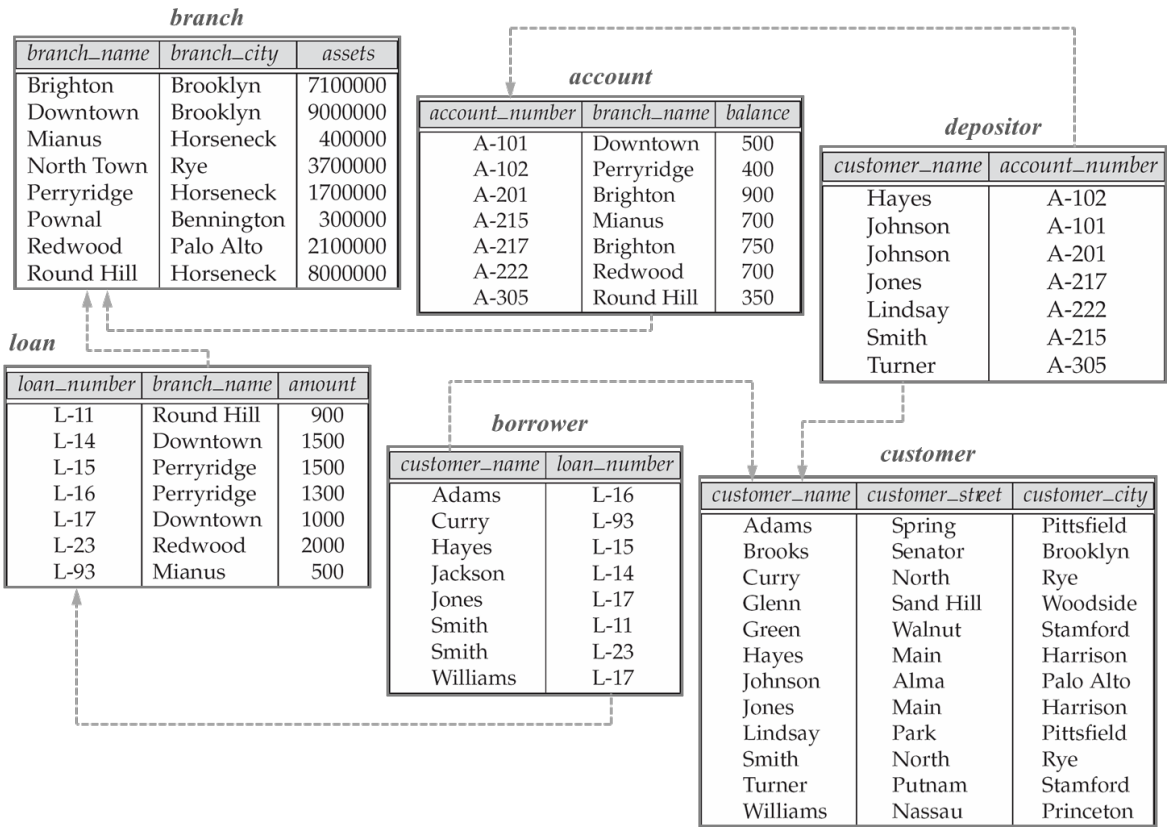
Lab Session 1: Introducing the Lab Environment

The file **bank.sql** that you should have downloaded with this lab guide contains a script for creating an example database with a set of tables as represented below:



branch

branch_name	branch_city	assets
Brighton	Brooklyn	7100000
Downtown	Brooklyn	9000000
Mianus	Horseneck	400000
North Town	Rye	3700000
Perryridge	Horseneck	1700000
Pownal	Bennington	300000
Redwood	Palo Alto	2100000
Round Hill	Horseneck	8000000

account

account_number	branch_name	balance
A-101	Downtown	500
A-102	Perryridge	400
A-201	Brighton	900
A-215	Mianus	700
A-217	Brighton	750
A-222	Redwood	700
A-305	Round Hill	350

depositor

customer_name	account_number
Hayes	A-102
Johnson	A-101
Johnson	A-201
Jones	A-217
Lindsay	A-222
Smith	A-215
Turner	A-305

loan

loan_number	branch_name	amount
L-11	Round Hill	900
L-14	Downtown	1500
L-15	Perryridge	1500
L-16	Perryridge	1300
L-17	Downtown	1000
L-23	Redwood	2000
L-93	Mianus	500

borrower

customer_name	loan_number
Adams	L-16
Curry	L-93
Hayes	L-15
Jackson	L-14
Jones	L-17
Smith	L-11
Smith	L-23
Williams	L-17

customer

customer_name	customer_street	customer_city
Adams	Spring	Pittsfield
Brooks	Senator	Brooklyn
Curry	North	Rye
Glenn	Sand Hill	Woodside
Green	Walnut	Stamford
Hayes	Main	Harrison
Johnson	Alma	Palo Alto
Jones	Main	Harrison
Lindsay	Park	Pittsfield
Smith	North	Rye
Turner	Putnam	Stamford
Williams	Nassau	Princeton

To create the database in the lab database management system, you will need to create the tables (with the SQL command CREATE TABLE) and then load the records to fill each table. For instance, the customer's table could be created with the following instruction:

```
CREATE TABLE customer
   ( customer_name      VARCHAR(255)      NOT NULL UNIQUE,
     customer_street    VARCHAR(255)      NOT NULL,
     customer_city      VARCHAR(255)      NOT NULL,
     PRIMARY KEY(customer_name) );
```

This instruction specifies the name of the table, the names of its three columns, the data type of each column and integrity constraints, such as declaring that no record field can be

NULL and that the customer name is the primary key of the table. The records are loaded into each table with INSERT instructions, such as:

```
INSERT INTO customer VALUES ('Jones' , 'Main' ,'Harrison');
```

The above instruction specifies the values in each column by the same order as they were defined upon table; its execution will add a new record to the customer table.

The script in file **bank.sql** includes SQL instructions to insert more records in the database than show in the figure above. We will be using these records to perform various tests. In future lab sessions we will also be using this database to demonstrate various concepts as they are introduced in the course.

You will have your own database at the server **db.ist.utl.pt,** which hosts the **MySQL** database management system.

To connect to that **MySQL** system you must have a password (which is not the same as the one you use to login to the Fénix academic system!).

You generate your new password by typing the command **mysql_reset** on cluster **sigma.tecnico.ulisboa.pt**

To execute the **mysql_reset** command, you must open an **SSH** session on cluster **sigma.tecnico.ulisboa.pt.** You could use **PuTTY** (on Windows) or **ssh** (on MacOS and Linux machines).

Once you have obtained your password[1], you can connect directly to the **MySQL** server at **db.tecnico.ulisboa.pt**, through the **mysql** command (or any another application, such as your own later in the course).

Let's now do it, one step at a time!

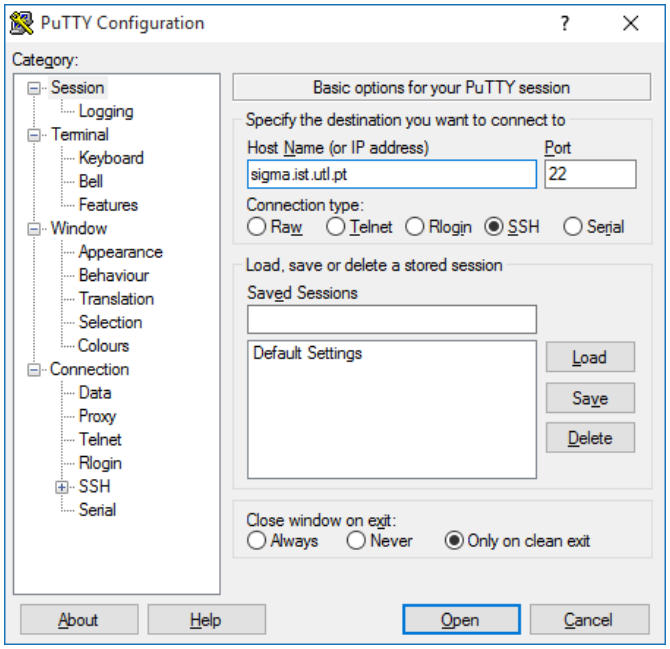
[1] If needed, you can generate a new password by invoking **mysql_reset** again any time later.

Obtaining the password to access your MySQL database at IST

1. Open the DSI self-service page:
   https://ciist.ist.utl.pt/servicos/self_service/index.php
   (use your Fénix credentials to authenticate.)

2. Activate the following services on the self-service page: **shell**, **web** and **cgi**.

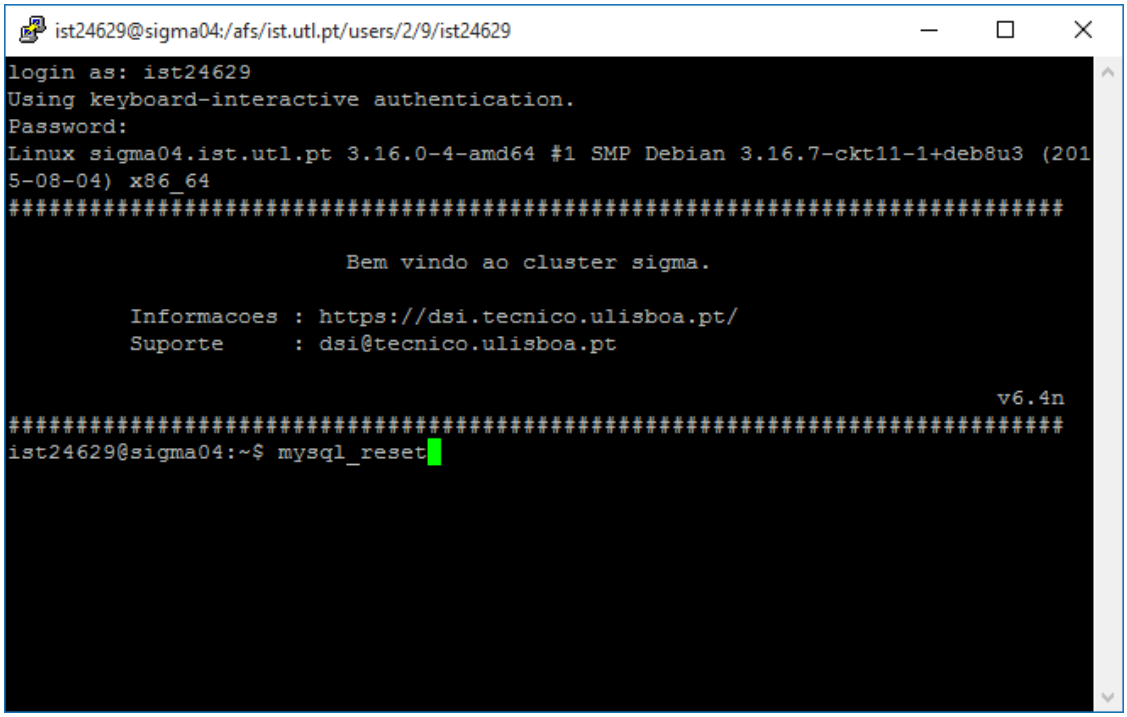
3. Open an SSL connection to the computing cluster at **sigma.tecnico.ulisboa.pt**
   - On Windows, try the **PuTTY** tool.
   - On MacOS or Linux, open a terminal window and use the **ssh** command**:**
     ssh sigma.tecnico.ulisboa.pt –l <your-identity>



4. Independently of the tool used to access sigma, you will have to authenticate with your Fenix credentials (username and password).

5. Once logged in on the cluster **sigma.tecnico.ulisboa.pt**, type the command:

   mysql_reset



   The output of this command has the format below, where istxxxxxx is your username and the password to the **MySQL** system is on the third line:

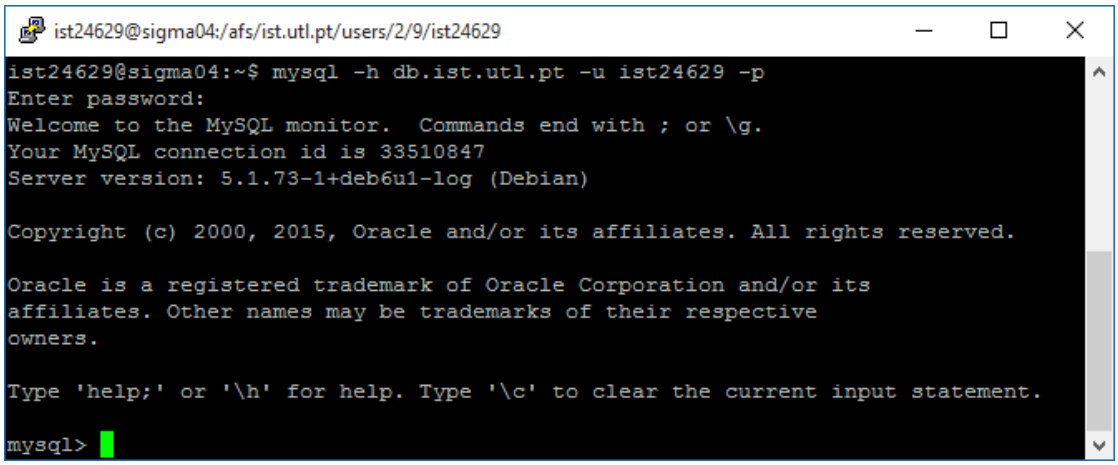
```
   login: istxxxxxx
database: istxxxxxx
password: xxxxxxxx

If you have a file ~/.my.cnf don't forget to update it.
```

IMPORTANT: save your password. You will be using it on the course throughout the semester. If you ever forget the password, run **mysql_reset** again to generate a new password.

Connecting to the database

6. At the command prompt in the cluster **sigma.tecnico.ulisboa.pt**, run the command below:
   mysql -h db.ist.utl.pt -u istxxxxxx -p
   - The parameter **-h db.tecnico.ulisboa.pt** specifies the MySQL server.
   - The parameter **-u istxxxxxx** specifies the username. Replace **istxxxxxx** by the login name obtained from **mysql_reset**.
   - The parameter **-p** requests the program to prompt your password. Give the password that you have just obtained from **mysql_reset**.



7. Once you have the **mysql>** prompt you are logged in to the **MySQL**. database server. At the **mysql>** prompt, run the command:
   SHOW DATABASES;
   The output is a list of the databases that are visible to you on the MySQL server.

   You should see two databases: **information_schema** e **istxxxxxx,** where **istxxxxxx** is your username.

8. Select your default database with the command, replacing **istxxxxxx** by your username:
   USE istxxxxxx

9. To list the tables created in your database, run the command:
   SHOW TABLES;

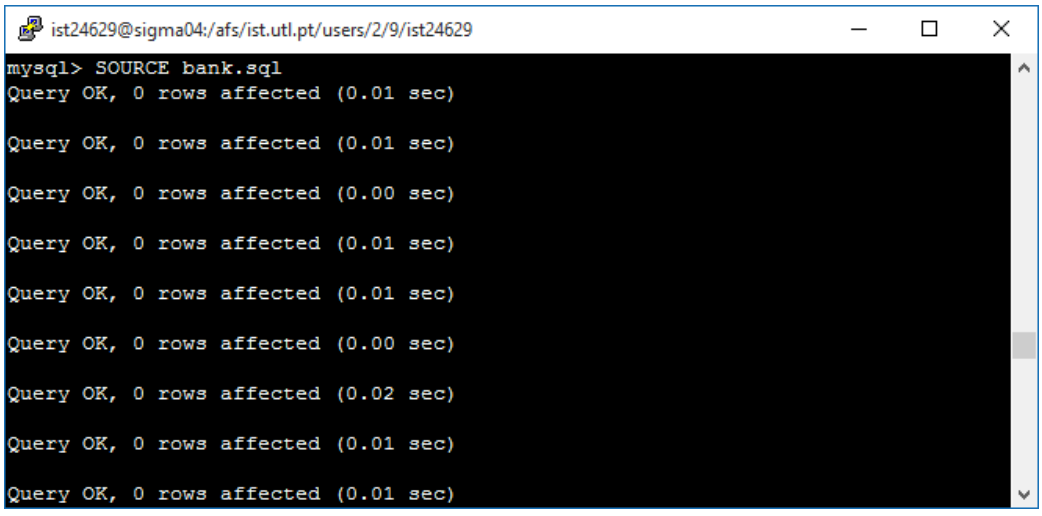
   At this point you should see none.

Creating the tables on the database

10. The file **bank.sql** has a set of SQL instructions to create an example database. Open the file with a text editor and inspect the content.

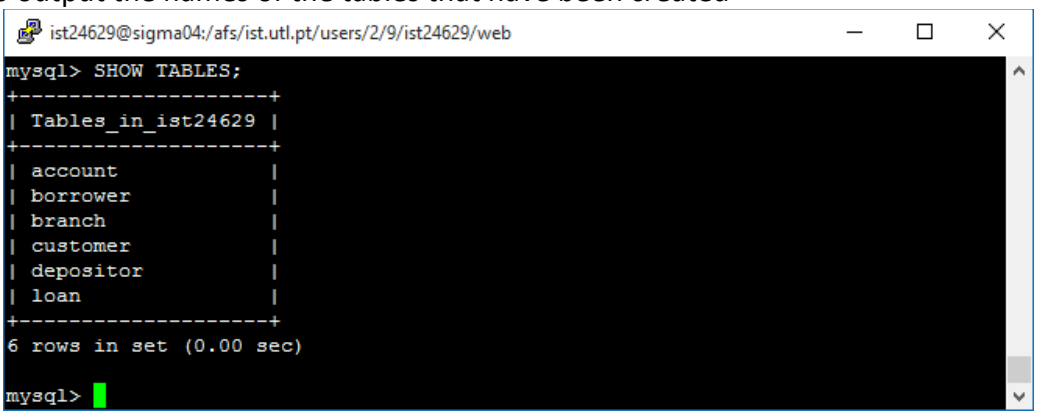
11. To create the tables of the example database you will need to transfer the file **bank.sql** to the cluster **sigma.tecnico.ulisboa.pt**.
    - In the lab desktop machines, your home directory at the cluster **sigma.tecnico.ulisboa.pt** is mapped to the home drive on your desktop. In that case, place the file in the *home directory* dragging it at the interactive user interface.
    - If you are using a Windows computer, use a tool like **FileZilla** or **WinSCP** to transfer the file **bank.sql**. You should authenticate at **sigma.ist.utl.pt** with your IST (Fénix) credentials. Drag the file to the *home directory*.
    - If using a Mac/Linux desktop, use a tool like **Cyberduck** or place the file in the home directory of the cluster **sigma.tecnico.ulisboa.pt** using the **scp** command on the shell:
      scp bank.sql istxxxxxx@sigma.tecnico.ulisboa.pt:

12. After transferring the file **bank.sql** to the cluster **sigma.tecnico.ulisboa.pt**, run the following command at **mysql>** prompt to create the tables in your selected database:
    SOURCE bank.sql

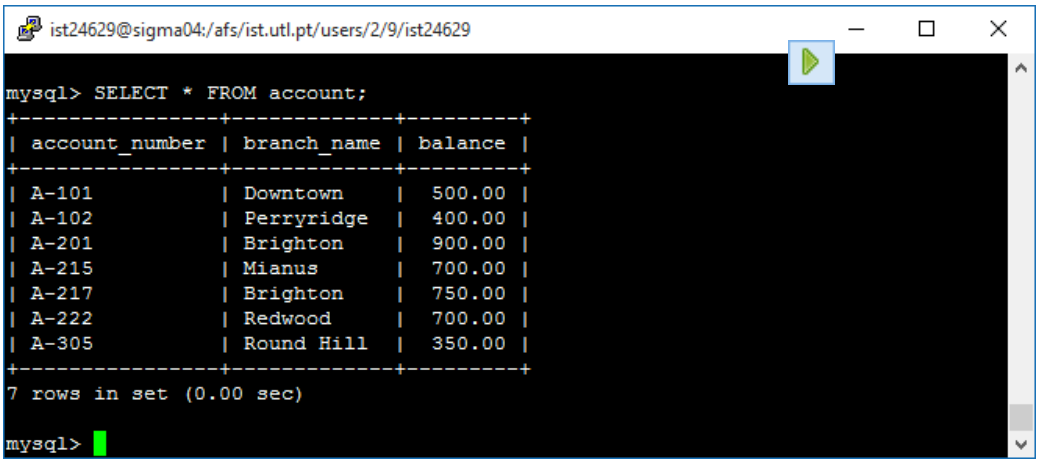


Querying the database

13. At the **mysql>** prompt run again the command:
    SHOW TABLES;
    To output the names of the tables that have been created



14. Write the SQL query below to see the contents of table **account**:
    SELECT * FROM account;



15. Now you can try other queries. For instance, to obtain the balance on 'A-101':
    SELECT balance
    FROM account
    WHERE account_number = 'A-101';

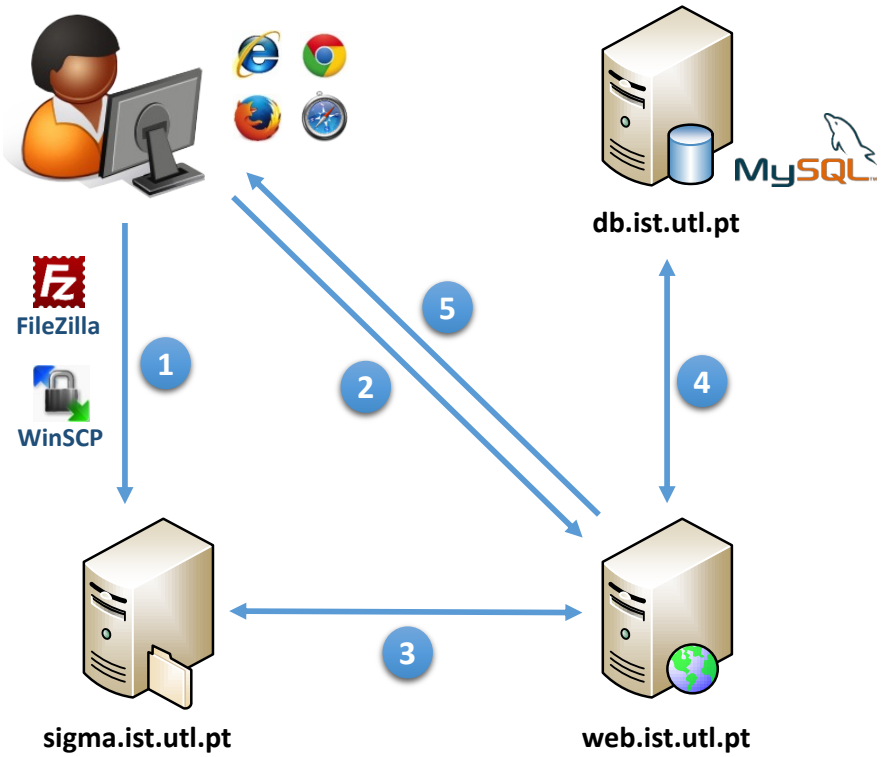
16. You can also obtain information about the structure of each table. Try:
    DESCRIBE customer;
    DESCRIBE account;
    Confirm now that the structure these tables matches their description on **bank.sql.** Try also other similar queries on the tables that you just created.

Simple PHP script to test the MySQL connection

The following steps will test the database connection using a **PHP** script, which will submit a query, identical to those you just run on the cluster **sigma.tecnico.ulisboa.pt** with the **mysql** program to obtain data that will be used to generate a **HTML** page to present the result to the end user.

You will have to transfer the script file to a Web server (more precisely, a folder shared at the cluster **sigma.tecnico.ulisboa.pt**), which will enable you to visualise the result of the execution of the script with any Web browser. The figure below illustrates this scenario:

1) Step 1: The PHP script is copied to the folder shared with the **web** server in the cluster **sigma.tecnico.ulisboa.pt**
2) When a user activates the URL that maps to the script from the Web browser (Step 2), the Web server (**web.tecnico.ulisboa.pt**) reads the script (Step 3), connects and sends the query to the database server (Step 4), and returns the result to the end (Step 5).

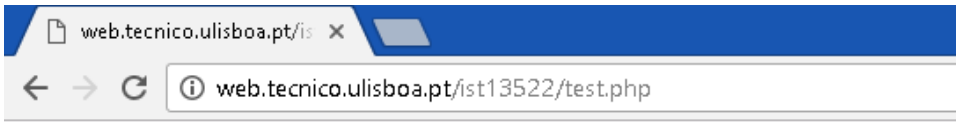


We will now develop one such scenario to test the environment to be used in this lab.

17. Open the file **test.php** in a text editor and change the username and password you see that with your username and password (from **mysql_reset**) in variables **$user** e **$password**. Remember to **save the file**.

18. You will have to transfer the file **test.php** to your **web** folder in the cluster **sigma.tecnico.ulisboa.pt**.
    - In the computing lab, your home directory on cluster **sigma.tecnico.ulisboa.pt** maybe already mapped in a *drive*. In that case, all you have to do is copy the file **test.php** to that *drive*. Place the script file in the **web** folder in your home directory.
    - If you are using your own computer, you will have to use **FileZilla WinSCP**.
    - On Mac/Linux shells, use the **scp** command, to transfer the file **test.php**.
      scp test.php istxxxxxx@sigma.tecnico.ulisboa.pt:web/
      In either case you need authenticate on **sigma.tecnico.ulisboa.pt** with your Fenix credentials.

19. On a Web browser (Firefox, Chrome, any) type the URL:
    http://web.tecnico.ulisboa.pt/istxxxxxx/test.php
    where **istxxxxxx** is replaced by your username. You should observe the following result:



Connected to MySQL database ist13522 on db.ist.utl.pt as user ist13522

Query: SELECT * FROM account;

7 records retrieved:

account_number	branch_name	balance
A-101	Downtown	500.00
A-102	Perryridge	400.00
A-201	Brighton	900.00
A-215	Mianus	700.00
A-217	Brighton	750.00
A-222	Redwood	700.00
A-305	Round Hill	350.00

Connection closed.

Test completed successfully. Now you know how to connect to your MySQL database.

Analysis of the PHP/HTML code

20. Open **test.php** again in a text editor and locate the calls to the following functions:
    - **$connection = new PDO**()
    - **$result = query**()
    - **$num = rowCount**()

    This is a typical sequence of calls to interact with a database server from a **PHP s**cript. What is the data type of variable **$result**?
    There is also a relevant statement at the end of **test.php**
    - **$connection = null;**

    What effect do you think that this statement will have?

    Observe the arguments being passed to each of the above functions.

21. In the file **test.php**, locate the query to the database:
    SELECT * FROM account;
    Compare the result of this query (it will show on the browser), with the result that obtained on the **mysql>** prompt.

22. Still on the code in **test.php**, observe how a HTML table is being assembled to generate the visualization of the query result using one of the typical ways of generating tables with database content:
    - **<table>** and **</table>** define the start/end of a HTML table
    - **<tr>** and **</tr>** define the start/end of row in the HTML table
    - **<td>** and **</td>** define the start/end of a cell (*division*) in the HTML table

23. In the Web browser, right-click on the page and observe its source code (on Firefox or Chrome, the option to inspect the code is “View page source”).

    Compare the **HTML** code of the page with the **PHP** code that generated that same page (**test.php**).