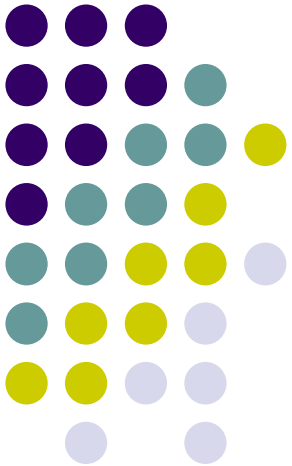
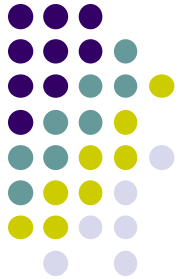
Data bases

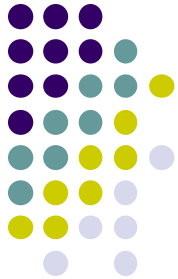


Software para data-bases



- Oracle
- Microsoft SQL Server (muy potente) y Microsoft Access (simple)
- PostgreSQL (potente/complejo libre open-source)
- SQLite (transportabile, ligero free open-source)
- MySQL (simple free open-source)
  - Muchos servidores usan "LAMP" (Linux, Apache, MySQL, y PHP)
  - Wikipedia esta gestionado con PHP y MySQL
  - Nos centraremos en MySQL en este curso

Ejemplo de una base de datos (world.sql)



code	name	continent	independence_year	population	gnp	head_of_state	...
AFG	Afghanistan	Asia	1919	22720000	5976.0	Mohammad Omar	...
NLD	Netherlands	Europe	1581	15864000	371362.0	Beatrix	...
...	...	...	...	...	...	...	...

countries

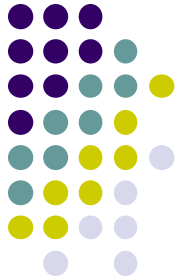
id	name	**country_code**	district	population
3793	New York	**USA**	New York	8008278
1	Los Angeles	**USA**	California	3694820
...	...	**...**	...	...

cities

country_code	language	official	percentage
AFG	Pashto	T	52.4
NLD	Dutch	T	95.6
...	...	...	...

languages

Repaso rápido SQL



- Ejemplos de sql query:

```
SELECT name FROM cities WHERE id = 17;
INSERT INTO countries VALUES ('SLD',
   'ENG', 'T', 100.0);
SHOW DATABASES;
USE database;
SHOW TABLES;
```

Select

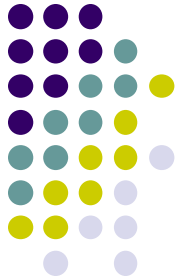
- SELECT column(s) FROM table;

SELECT name, code FROM countries;

name	code
China	CHN
United States	USA
Indonesia	IND
Brazil	BRA
Pakistan	PAK

SELECT * FROM *table*; recupera todas las columnas

Distinct



- SELECT *DISTINCT column(s)* FROM *table*;

SELECT language
FROM languages;

language
Dutch
English
English
Papiamento
Spanish
Spanish
Spanish

SELECT DISTINCT language
FROM languages;

language
Dutch
English
Papiamento
Spanish
...

Where

- SELECT column(s) FROM table WHERE condition(s);

```
SELECT name, population FROM cities WHERE country_code =
    "FSM";
```

name	population
Weno	22000
Palikir	8600

condiciones
BETWEEN min AND max
LIKE pattern
IN (value, value, ..., value)

```
SELECT code, name, gnp FROM countries WHERE gnp >
    2000000;
```

code	name	gnp
JPN	Japan	3787042.00
DEU	Germany	2133367.00
USA	United States	8510700.00

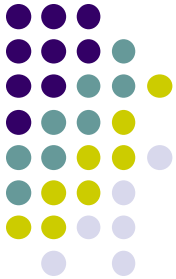
Like

- WHERE column LIKE pattern
- LIKE 'text%' busca por un texto que empieza con un patrón dado.
- LIKE '%text' busca por un texto que acaba con un patrón dado.
- LIKE '%text%' busca por un texto que contiene exactamente el patrón.

```
SELECT code, name, population FROM countries WHERE name LIKE
    'United%';
```

code	name	population
ARE	United Arab Emirates	2441000
GBR	United Kingdom	59623400
USA	United States	278357000
UMI	United States Minor Outlying Islands	0

Order by



- ORDER BY column(s)

```
SELECT code, name, population FROM countries
WHERE name LIKE 'United%' ORDER BY population;
```

code	name	population
UMI	United States Minor Outlying Islands	0
ARE	United Arab Emirates	2441000
GBR	United Kingdom	59623400
USA	United States	27835700

Limit

LIMIT number

```
SELECT name FROM cities WHERE name LIKE 'K%' LIMIT 5;
```

name
Kabul
Khulna
Kingston upon Hull
Koudougou
Kafr al-Dawwar

- Se puede usar para obtener los primeros N elementos de una category (combinando `ORDER BY` y `LIMIT`)
- Permite visualizar los primeros elementos cuando se hace un test

Query multi-tabla

- **Llave primaria**: una columna que garantiza che cada columna es unica (e.g. Lisa Simpson tiene el 888)
- **Llave foranea**: una columna de la tabla A que almacena un valore de *primary key* de la tabla B
  - (ej. Valores de notas con el student_id de 888 son de Lisa)

students

id	name	email
123	Bart	bart@fox.com
456	Milhouse	milhouse@fox.com
888	Lisa	lisa@fox.com
404	Ralph	ralph@fox.com

courses

id	name	teacher_id
10001	Computer Science 142	1234
10002	Computer Science 143	5678
10003	Computer Science 190M	9012
10004	Informatics 100	1234

grades

student_id	course_id	grade
123	10001	B-
123	10002	C
456	10001	B+
888	10002	A+
888	10003	A+
404	10004	D+

teachers

id	name
1234	Krabappel
5678	Hoover
9012	Stepp

Query multi-tabla

- Cuando el dataset es grande y se distribuye en múltiples tablas, necesitamos queries que respondan a cuestiones mas complejas como:
- Que cursos ha tomado Bart obteniendo un B- o mas?
- Que cursos han sido elegidos por Bart y Lisa?
- Quien son los profesores que ha tenido Bart?
- A cuantos estudiantes en total ha ensenado Ms. Krabappel y cual son sus nombres?
- Para hacer eso necesitamos juntar (**join**) datos da diferente tablas en nuestra query SQL

students

id	name	email
123	Bart	bart@fox.com
456	Milhouse	milhouse@fox.com
888	Lisa	lisa@fox.com
404	Ralph	ralph@fox.com

courses

id	name	teacher_id
10001	Computer Science 142	1234
10002	Computer Science 143	5678
10003	Computer Science 190M	9012
10004	Informatics 100	1234

grades

student_id	course_id	grade
123	10001	B-
123	10002	C
456	10001	B+
888	10002	A+
888	10003	A+

teachers

id	name
1234	Krabappel
5678	Hoover
9012	Stepp

Juntar usando ON

```
SELECT column(s)
FROM table1
JOIN table2 ON condition(s)
...
JOIN tableN ON condition(s);
```

Ejemplo:

```
SELECT *
FROM students
JOIN grades ON id = student_id;
```

- join: combina informaciones provenientes de dos o mas tablas que satisfacen unas cuantas condiciones
- La condición ON especifica cual dado de cada será elegido
- A menudo las líneas son conectadas a sus columnas clave (id)

Ejemplo

QUERY= Que cursos ha tomado Bart?

```
SELECT *
FROM students
JOIN grades ON id = student_id;
```

id	name	email	student_id	course_id	grade
123	Bart	bart@fox.com	123	10001	B-
123	Bart	bart@fox.com	123	10002	C
404	Ralph	ralph@fox.com	404	10004	D+
456	Milhouse	milhouse@fox.com	456	10001	B+
888	Lisa	lisa@fox.com	888	10002	A+
888	Lisa	lisa@fox.com	888	10003	A+

- table.column puede ser utilizado para distinguir los nombres de columnas:

```
SELECT *
FROM students
JOIN grades ON students.id = grades.student_id;
```

Filtrar resultados

```
SELECT name, course_id, grade
FROM students
JOIN grades ON id = student_id;
```

name	course_id	grade
Bart	10001	B-
Bart	10002	C
Ralph	10004	D+
Milhouse	10001	B+
Lisa	10002	A+
Lisa	10003	A+

Filtrar resultados

```
SELECT name, course_id, grade
FROM students
JOIN grades ON id = student_id
WHERE name = 'Bart';
```

name	course_id	grade
Bart	10001	B-
Bart	10002	C

- FROM / JOIN juntan las tablas correctas, y WHERE filtra los resultados
- Cual es el rol de ON y cual es el rol de WHERE?
  - ON relaciona directamente columnas de las tablas juntadas
  - WHERE set limita ulteriormente a un particular valor (123, 'Bart')

Errores comunes

```
SELECT name, id, course_id, grade
FROM students
JOIN grades ON id = 123
WHERE id = student_id;
```

name	id	course_id	grade
Bart	123	10001	B-
Bart	123	10002	C

- La query arriba produce el mismo resultado que antes, pero el estilo es incorrecto Porque?
- La condicion JOIN ON es incorrecta. En este caso no relaciona valores de notas con notas de estudiantes
  - Se relacionan cuando pertenecen a un estudiante con el mismo ID.
  - El filtraje de un ID especifico deberia ser hecho con la condicion WHERE, y no JOIN ON.

Definir nombres a tablas

```
SELECT s.name, g.*
FROM students s
JOIN grades g ON s.id = g.student_id
WHERE g.grade <= 'C';
```

name	student_id	course_id	grade
Bart	123	10001	B-
Bart	123	10002	C
Milhouse	456	10001	B+
Lisa	888	10002	A+
Lisa	888	10003	A+

- Podemos definir nombres a tablas, como si fuera una variable
- Para especificar todas las columnas de una tablas, podemos escribir *table*.*
- (grade ordena alfabeticamente, por lo cual se visualizan las notas C superior al umbral <= C)

Juntar mas que una tabla

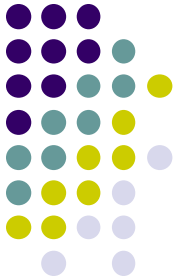
```
SELECT c.name
FROM courses c
JOIN grades g ON g.course_id = c.id
JOIN students bart ON g.student_id = bart.id
WHERE bart.name = 'Bart' AND g.grade <= 'B-';
```

Name

Computer Science 142

- Se han juntado mas que dos tablas
- La query representa:
  - El nombre de todos los cursos en los cuales Bart ha obtenido B- o mas.

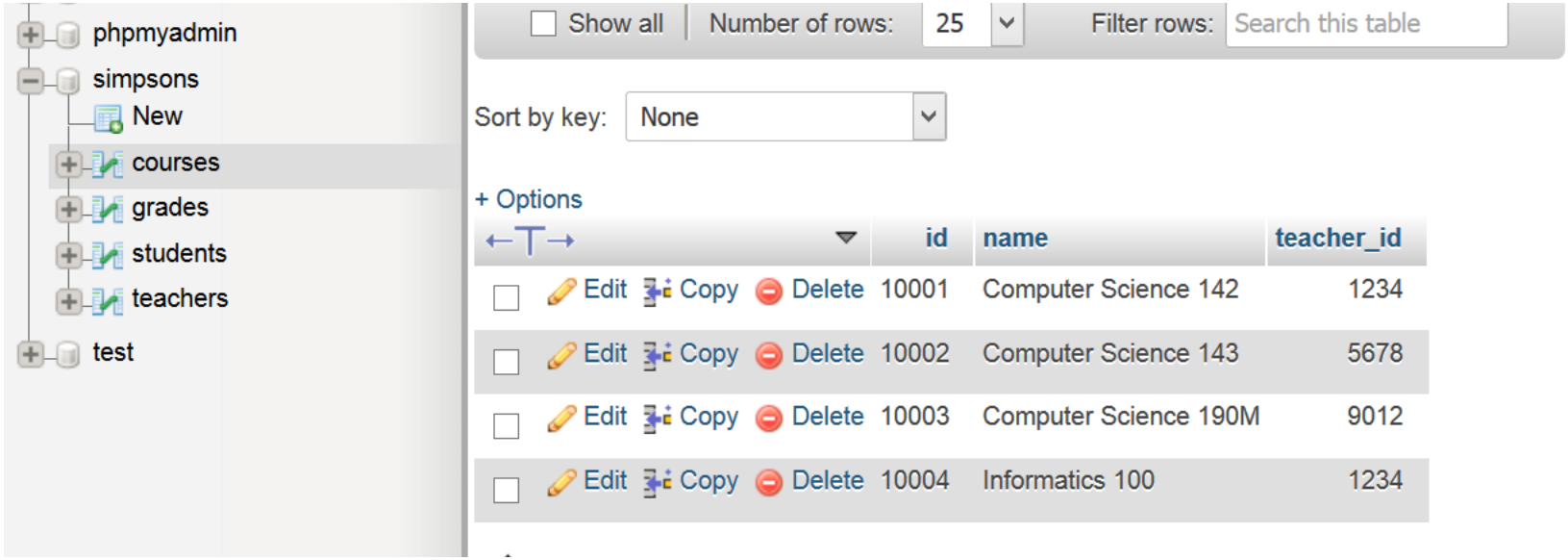
Consejos para escribir un query



- Para identificar la sintaxis del query podemos seguir estos pasos:
  - Cual tabla(s) contiene los datos buscados? (FROM)
  - Desde que columna necesito extraer los resultados? (SELECT)
  - Como están conectadas las tablas (JOIN) e como tengo que filtrar los valores (WHERE)?
- Testear en un data set pequeño (imdb_small, simpson, world).
- Confirmar que funciona en un data set mas grande (imdb).
- Intentar la query inicialmente en un MySQL console (PhpMyAdmin).
- Escribir el código PHP que le corresponde.

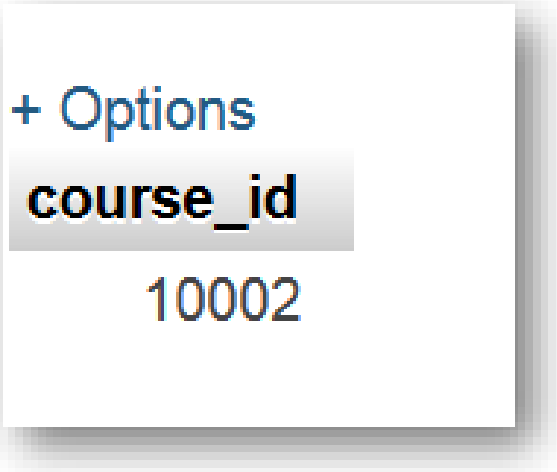
Ejemplos colaborativos

- https://codeshare.io/COW
- Base de datos: “simpsons.sql” disponible en el campus



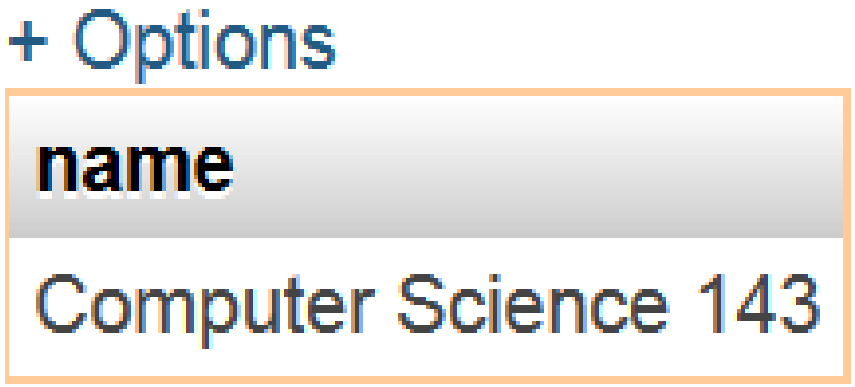
Ejemplos para practicar

- https://codeshare.io/COW
- Ejercicio: Cual cursos han sido elegidos por Bart y Lisa?



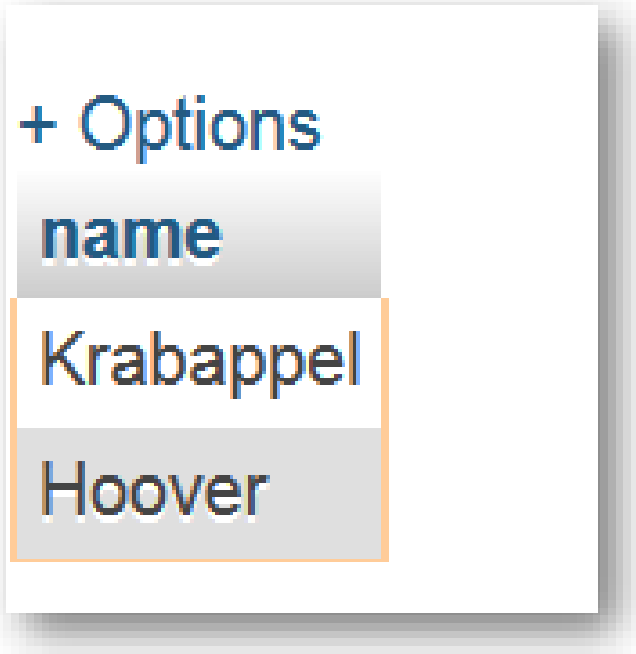
Ejemplos para practicar

- Esta versión recupera los nombre de cursos course *names*, y necesita solo saber los nombres de Bart/Lisa, y no sus IDs.



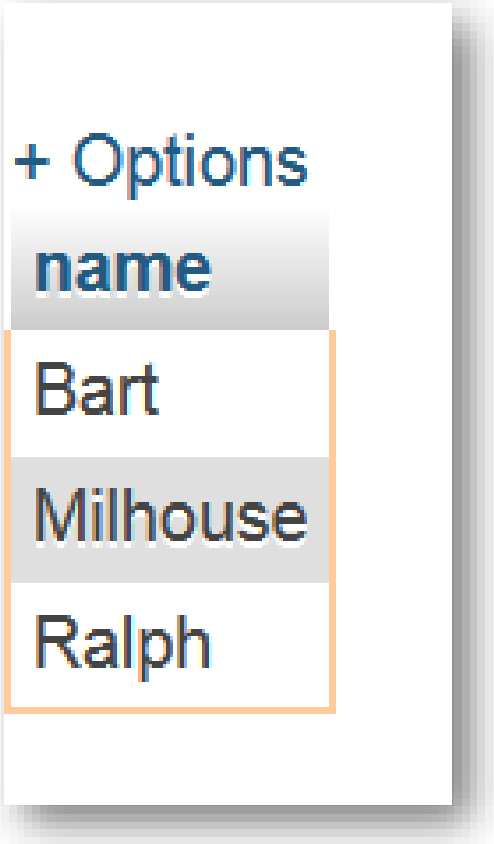
Ejemplos para practicar

- Cual son los nombres de todos los profesores que Bart ha tenido?

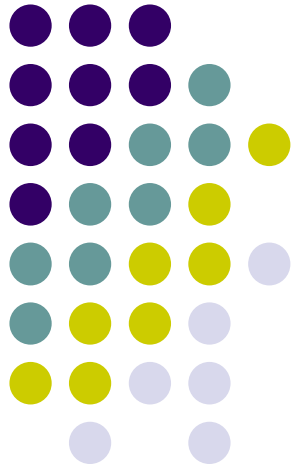


Ejemplos para practicar

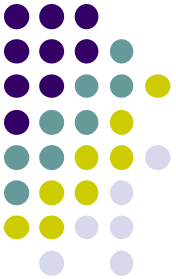
- A cuantos estudiantes en total Ms. Krabappel ha ensenado y cual son sus nombres?



PHP y mySQL



Conexión a un data base (método antiguo)



```
mysql_connect(“host”,”user”,”password”)
    or die(“Database is not available. Try again later.”);
```

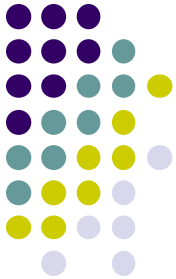
- Alternativamente (para tratar el caso que genera el error)

```
if(!mysql_connect(“host”,”user”,”password”))
    {
    echo “Database is not available. Try again later\n”;
    exit();
    }
```

Efectuar una petición (query)

```
$result=mysql_query("SELECT id,titel,interpret,jahr FROM cds
    ORDER BY interpret;");
```

Query usando PDO (librería PHP)



- Remplaza el comando mas viejo ***mysql_connect***

```
$name = new PDO("dbprogram:dbname=database;host=server; charset=utf8",
    username, password);

$name->query("SQL query");
```

- Ejemplo

```
$db = new PDO("mysql:dbname=world;host=localhost", "traveler",
    "packmybags");
$conn = new PDO("mysql:host=$servername;dbname=$dbname;charset=utf8",
    $username, $password);

$db->query("SELECT * FROM countries WHERE population > 100000000;");
foreach ($rows as $row) {
    do something with $row;
}
```

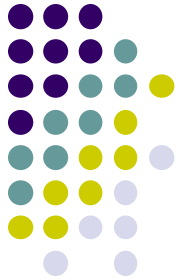
Otro ejemplo PDO

```
<?php
$db = new PDO("mysql:dbname=imdb_small", "jessica", "guinness");
$rows = $db->query("SELECT * FROM actors WHERE last_name LIKE
    'Del%'");
foreach ($rows as $row) {
  ?>
  <li> First name: <?= $row["first_name"] ?>,
      Last name:  <?= $row["last_name"]  ?> </li>
  <?php
}
?>
```

Resultado:

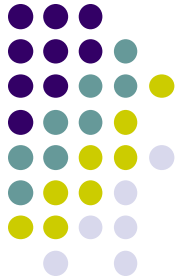
- First name: Benicio, Last name: Del Toro
- First name: Michael, Last name: Delano
- …

Comandos de PDO



nombre	Descripción
query	Efectua una query SQL SELECT en el database (DB)
exec	Efectua una query SQL que modifique el database (INSERT, DELETE, UPDATE, etc.)
getAttribute,setAttribute	Recupera/define las propriedades de connection con el DB connection
quote	Codifica un valor que se usa adentro de un query para evitar inyección SQL

Ejemplos PDO



- Incrustar variables

```
# get query parameter for name of movie
$title = $_GET["movietitle"];
$rows = $db->query("SELECT year FROM movies WHERE
    name = '$title'");
```

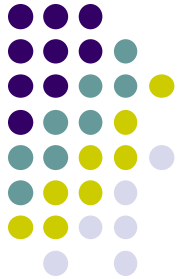
CUIDADO peligros porque permite incluir codigo pirata!!!

- Alternativa (mas correcta):

```
$title = $_GET["movietitle"];
$title = $db->quote($title);
$rows = $db->query("SELECT year FROM movies WHERE
    name = $title");
```

- Quote elimina todos caracteres ilegale y rodea el string con comillas “ ”

Gestión de los errores



```
$db = new PDO("mysql:dbname=imdb_small", "jessica", "guinness");
$db->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
$rows = $db->query("SEEELECT * FROM movies WHERE year = 2000");
```

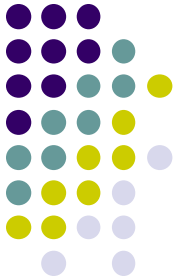
- El PDO en caso de error devuelve `FALSE` o `NULL`
- Sintaxis para recuperar error:

```
try {
    comando(s);
} catch (ExceptionType $name) {
    código de gestión del error;
}
```

Ejemplo conexión

```
try {
    $db = new PDO("mysql:dbname=imdb_small", "jessica",
    "guinness");
    $db->setAttribute(PDO::ATTR_ERRMODE,
    PDO::ERRMODE_EXCEPTION);
    $rows = $db->query("SEEELECT * FROM movies WHERE year =
    2000");
    foreach ($rows as $row) { ... }
} catch (PDOException $ex) {
    ?>
    <p>Sorry, a database error occurred. Please try again later.</p>
    <p>(Error details: <?= $ex->getMessage() ?>)</p>
    <?php
}
```

Ejemplo conexión (2)

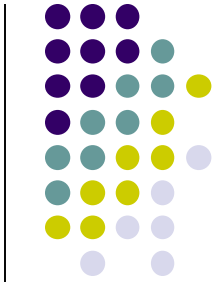


```php
<?php
$servername = "localhost";
$username = "root";
$password = "";

try {
    $conn = new PDO("mysql:host=$servername;dbname=world", $username, $password);
    // set the PDO error mode to exception
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
    echo "Connected successfully";
    }
catch(PDOException $e)
    {
    echo "Connection failed: " . $e->getMessage();
    }
?>
```

Fuente: www.w3schools.com/

Analizar el output de PDO



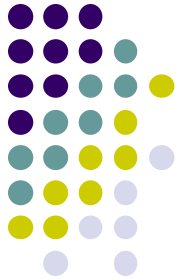
la variable $rows devuelta con la query PDO no es un array pero un objecto de tipo PDOStatement. Esto son los métodos disponibles:

- columnCount() numero de columnas del resultado
- fetch() devuelve la línea siguiente del resultado
- fetchColumn(number) devuelve la columna siguiente del resultado
- rowCount() devuelve el numero de líneas devuelta por la query

- Ejemplo:

```
if ($db->rowCount() > 0) {
    $first_row = $db->fetch();
    ...
}
```

Cerrar la conexión



- La conexión se cierra automáticamente cuando el script acaba. Para cerrar la conexión antes se puede usar:

```
$conn = null;
```

Ejemplo creación tabla

```php
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";

try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);
    // set the PDO error mode to exception
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    // sql to create table
    $sql = "CREATE TABLE MyGuests (
    id INT(6) UNSIGNED AUTO_INCREMENT PRIMARY KEY,
    firstname VARCHAR(30) NOT NULL,
    lastname VARCHAR(30) NOT NULL,
    email VARCHAR(50),
    reg_date TIMESTAMP
    )";

    // use exec() because no results are returned
    $conn->exec($sql);
    echo "Table MyGuests created successfully";
    }
catch(PDOException $e)
    {
    echo $sql . "<br>" . $e->getMessage();
    }

$conn = null;
?>
```

Fuente: www.w3schools.com/

Ejemplo insert en una tabla

código

```php
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";

try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname",
      $username, $password);
    // set the PDO error mode to exception
    $conn->setAttribute(PDO::ATTR_ERRMODE,
      PDO::ERRMODE_EXCEPTION);
    $sql = "INSERT INTO MyGuests (firstname, lastname, email)
    VALUES ('John', 'Doe', 'john@example.com')";
    // use exec() because no results are returned
    $conn->exec($sql);
    $last_id = $conn->lastInsertId();
    echo "New record created successfully. Last inserted ID is:
      " . $last_id;
    }
catch(PDOException $e)
    {
    echo $sql . "<br>" . $e->getMessage();
    }

$conn = null;
?>
```

zoom

```php
$sql = "INSERT INTO MyGuests (firstname,
    lastname, email)
    VALUES ('John', 'Doe',
    'john@example.com')";


// use exec() because no results are
    returned
    $conn->exec($sql);


$last_id = $conn-
  >lastInsertId();
    echo "New record created
  successfully. Last inserted
  ID is: " . $last_id;
```

Fuente: www.w3schools.com/

Ejemplo de múltiples insert

código

```php
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";

try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname", $username, $password);
    // set the PDO error mode to exception
    $conn->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    // begin the transaction
    $conn->beginTransaction();
    // our SQL statememtns
    $conn->exec("INSERT INTO MyGuests (firstname, lastname, email)
    VALUES ('John', 'Doe', 'john@example.com')");
    $conn->exec("INSERT INTO MyGuests (firstname, lastname, email)
    VALUES ('Mary', 'Moe', 'mary@example.com')");
    $conn->exec("INSERT INTO MyGuests (firstname, lastname, email)
    VALUES ('Julie', 'Dooley', 'julie@example.com')");

    // commit the transaction
    $conn->commit();
    echo "New records created successfully";
    }
catch(PDOException $e)
    {
    // roll back the transaction if something failed
    $conn->rollback();
    echo "Error: " . $e->getMessage();
    }

$conn = null;
?>
```

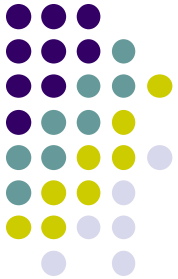
zoom

```php
// begin the transaction
    $conn->beginTransaction();

// our SQL statememtns
    $conn->exec("INSERT INTO
    MyGuests (firstname, lastname,
    email)
    VALUES ('John', 'Doe',
    'john@example.com')");

// commit the transaction
    $conn->commit();
    echo "New records created
    successfully";
}
```

Ejemplo borrar un elemento



código

```php
<?php
$servername = "localhost";
$username = "username";
$password = "password";
$dbname = "myDBPDO";

try {
    $conn = new PDO("mysql:host=$servername;dbname=$dbname",
      $username, $password);
    // set the PDO error mode to exception
    $conn->setAttribute(PDO::ATTR_ERRMODE,
      PDO::ERRMODE_EXCEPTION);

    // sql to delete a record
    $sql = "DELETE FROM MyGuests WHERE id=3";

    // use exec() because no results are returned
    $conn->exec($sql);
    echo "Record deleted successfully";
    }
catch(PDOException $e)
    {
    echo $sql . "<br>" . $e->getMessage();
    }

$conn = null;
?>
```

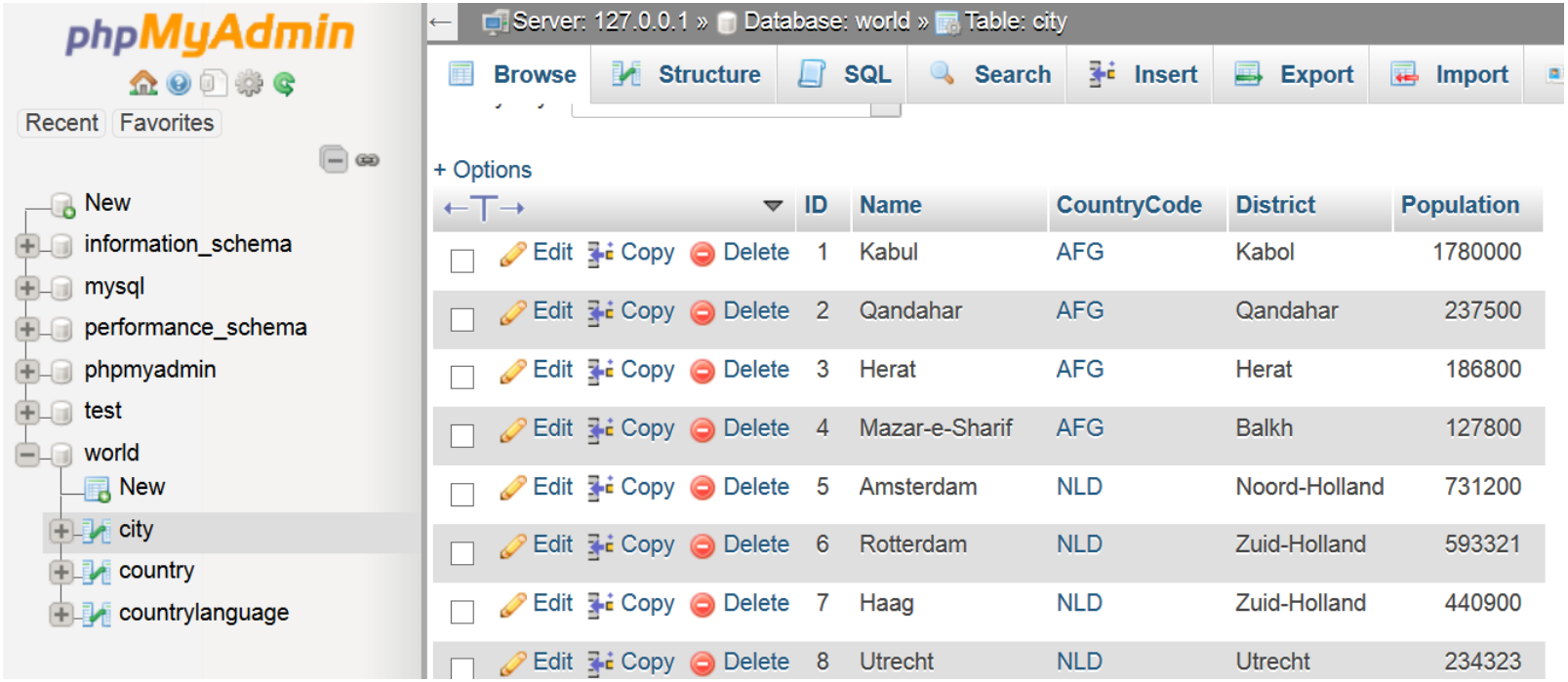
zoom

```
// sql to delete a record
    $sql = "DELETE FROM
     MyGuests WHERE id=3";

    // use exec() because no
     results are returned
    $conn->exec($sql);
```

Ejercicios colaborativos

- https://codeshare.io/COW
- Base de datos: “world.sql” disponible en el campus

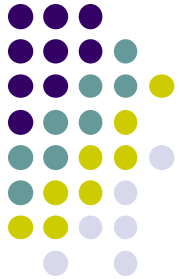


Ejercicios colaborativos

- Escribir un script que permita:
  - Conectarse a la base datos world
  - Visualizar en la lista de las primeras 5 ciudades

id	name	country_code	district	population
1	Kabul	AFG	Kabol	1780000
2	Qandahar	AFG	Qandahar	237500
3	Herat	AFG	Herat	186800
4	Mazar-e-Sharif	AFG	Balkh	127800
5	Amsterdam	NLD	Noord-Holland	731200

Ejercicios para casa



- Escribir un script que permita:
  - Conectarse a la base datos world
  - Visualizar en una tabla la lista de ciudades (city-country code-population)
  - Visualizar solo las ciudades de Afganistán (query cruzada con tabla country)
  - Insertar la ciudad “Kunduz” de Afganistán con 304,600 habitantes