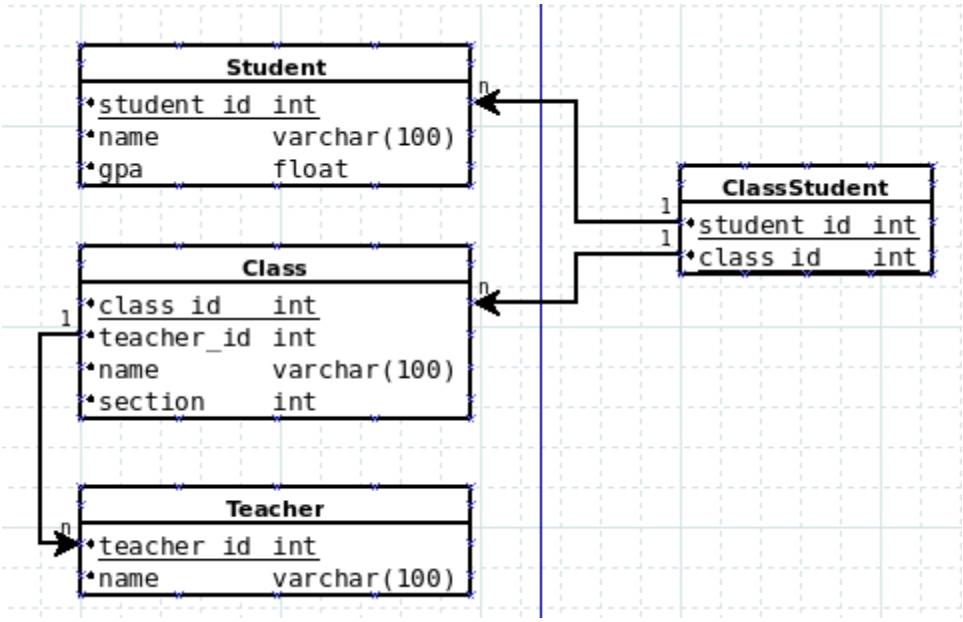
In Class Practice: Simple school database

Database Schema

In the database manager (http://www.alketo.info/database), create four tables. Make sure to prefix them with your initials or something, if your teammates are also working in this database.



Notice that the Class → Teacher table has a 1:N relationship (a teacher may have many classes, but a class only has one teacher).
The Student ↔ Class tables have an N:N relationship, which requires the intermediate table ClassStudent.

Student Table

Column name	Data type	Properties
student_id	int	primary key auto_increment
name	varchar(100)	not null
gpa	float	not null

Teacher Table

Column name	Data type	Properties
teacher_id	int	primary key auto_increment
name	varchar(100)	not null

Class Table

Column name	Data type	Properties
class_id	int	primary key auto_increment
teacher_id	Int	foreign key, references Teacher table
name	varchar(100)	not null
section	int	not null, unique

ClassStudent Table

Column name	Data type	Properties
student_id	int	primary key
class_id	int	primary key

Insert data

Copy-paste each of these INSERT queries into the database manager to fill the database with information we can query on. **Make sure to rename table names in the queries, if you have a prefix on your table!**

If you printed out this document, you oughta download it off the class website so you don't have to type all of this out!!

```
INSERT INTO Teacher
( name )
VALUES
( "Pratt" ),
( "Johnson" ),
( "Herbert" ),
( "Satterfield" ),
( "Pallet" ),
( "Wyatt" )
```

```
INSERT INTO Student
( name, gpa )
VALUES
( "Wullem Granville", 3.0 ),
( "Moray Aarle", 4.0 ),
( "Maja Descoteaux", 2.5 ),
( "Bedelia Apperlo", 3.5 ),
( "Ptolemais Neroni", 4.0 ),
( "Palmer Appeldoorn", 3.2 )
```

```
INSERT INTO Class
( teacher_id, name, section )
VALUES
( ( SELECT teacher_id FROM Teacher WHERE name = 'Pratt' ),
"C++ Programming I", "12345" ),

( ( SELECT teacher_id FROM Teacher WHERE name = 'Pratt' ),
"C++ Programming II", "12346" ),

( ( SELECT teacher_id FROM Teacher WHERE name = 'Johnson' ),
"Visual Basic .NET", "12347" ),

( ( SELECT teacher_id FROM Teacher WHERE name = 'Herbert' ),
"A+ Computer Hardware", "12348" ),

( ( SELECT teacher_id FROM Teacher WHERE name = 'Satterfield' ),
"English Composition I", "54321" ),

( ( SELECT teacher_id FROM Teacher WHERE name = 'Satterfield' ),
"English Composition II", "54322" ),

( ( SELECT teacher_id FROM Teacher WHERE name = 'Wyatt' ),
"Precalculus", "11111" ),

( ( SELECT teacher_id FROM Teacher WHERE name = 'Pallet' ),
"Calculus I", "11112" ),

( ( SELECT teacher_id FROM Teacher WHERE name = 'Wyatt' ),
"Calculus I", "11113" ),

( ( SELECT teacher_id FROM Teacher WHERE name = 'Pallet' ),
"Calculus II", "11114" ),

( ( SELECT teacher_id FROM Teacher WHERE name = 'Wyatt' ),
"Calculus III", "11115" ),

( ( SELECT teacher_id FROM Teacher WHERE name = 'Pallet' ),
"Differential Equations", "11116" )
```

```
INSERT INTO ClassStudent
( student_id, class_id )
VALUES
( ( SELECT student_id FROM Student WHERE name = "Wullem Granville" ),
( SELECT class_id FROM Class WHERE section = 12345 ) ),

( ( SELECT student_id FROM Student WHERE name = "Wullem Granville" ),
( SELECT class_id FROM Class WHERE section = 54321 ) ),

( ( SELECT student_id FROM Student WHERE name = "Wullem Granville" ),
( SELECT class_id FROM Class WHERE section = 11111 ) ),

( ( SELECT student_id FROM Student WHERE name = "Moray Aarle" ),
( SELECT class_id FROM Class WHERE section = 12347 ) ),

( ( SELECT student_id FROM Student WHERE name = "Maja Descoteaux" ),
( SELECT class_id FROM Class WHERE section = 12345 ) ),

( ( SELECT student_id FROM Student WHERE name = "Maja Descoteaux" ),
( SELECT class_id FROM Class WHERE section = 11112 ) ),

( ( SELECT student_id FROM Student WHERE name = "Bedelia Apperlo" ),
( SELECT class_id FROM Class WHERE section = 12345 ) ),

( ( SELECT student_id FROM Student WHERE name = "Bedelia Apperlo" ),
( SELECT class_id FROM Class WHERE section = 11113 ) ),

( ( SELECT student_id FROM Student WHERE name = "Ptolemais Neroni" ),
( SELECT class_id FROM Class WHERE section = 11116 ) ),

( ( SELECT student_id FROM Student WHERE name = "Ptolemais Neroni" ),
( SELECT class_id FROM Class WHERE section = 54322 ) ),

( ( SELECT student_id FROM Student WHERE name = "Ptolemais Neroni" ),
( SELECT class_id FROM Class WHERE section = 12348 ) ),

( ( SELECT student_id FROM Student WHERE name = "Palmer Appeldoorn" ),
( SELECT class_id FROM Class WHERE section = 11116 ) ),

( ( SELECT student_id FROM Student WHERE name = "Palmer Appeldoorn" ),
( SELECT class_id FROM Class WHERE section = 12347 ) ),

( ( SELECT student_id FROM Student WHERE name = "Palmer Appeldoorn" ),
( SELECT class_id FROM Class WHERE section = 54322 ) )
```

Queries

Write the following queries in the database manager, and then copy each one to a text file, so you can reuse the same query in the PHP code.

*Hint: When I say "Select all columns", that means "Select *". Otherwise, I will specify which columns to select for.*

1. Select all columns from every record in the Class table.

2. Select all columns from records in the Class table, where the name includes the word "Calculus".

Hint: To look at a substring in a MySQL query, use:

```
WHERE name LIKE '%Calculus%'
```

3. Select all columns from every record in the Class table and corresponding Teacher records.

Hint: This will require an inner join between Class and Teacher tables.

4. Select only the "**name**" column from Class, and the "**name**" column from Teacher, for all records in the Class table. Rename Class.name to "class_name" and rename Teacher.name to "teacher_name".

Hint: You will need to use the AS keyword to give the columns an alias.

5. Select all columns from any students whose GPA is greater than or equal to 3.0

Hint: This requires a WHERE statement

6. Select all columns from any students who are taking a class whose name is "Calculus I"

Hint: You will need to do an inner join between Student, ClassStudent, and Class.

PHP with MySQL

Download the **database.php** sample file from under **Sample Code** > **MySQL**. It has the code required to connect to a database, but no queries written for our SELECT statements.
If you need the code for the Database class, it is attached at the end of this document.

Remember to fill in your database information, under $hostname, $database, $username, and $password.

You will create six functions to correspond to the queries we wrote before:

1. Select_All_Classes()
2. Select_Classes_With_Name($name)
3. Select_All_Classes_And_Teachers()
4. Select_Class_Name_And_Teacher_Name()
5. Select_All_Students_With_Gpa_Greater_Than($gpa)
6. Select_Students_Taking_Class($name)

Note that some of these functions have parameter variables. We will swap out hard-coded string values like “Calculus” or 3.0 for these parameter values.

With each query function, it will look like:

```
function SelectThing()
{
     $query = ""; // You fill this out
     return $this->Query( $query );
}
```

After you create your query, you will need to call the `return $this->Query( $query );`

For Select_Classes_With_Name($name), you will replace:

```
WHERE name LIKE '%Calculus%'
```

with

```
WHERE name LIKE '%$name%'
```

For Select_All_Students_With_Gpa_Greater_Than($gpa), you will replace the 3.0 with $gpa

For Students_Taking_Class($name), you will replace, 'Calculus I' with '$name'.

Don't forget to enclose strings within single or double-quotes ' '
And you can concatenate strings with the dot-operator: . Or .=

Query solutions

1.	`SELECT * FROM Class`
2.	`SELECT * FROM Class` `WHERE name LIKE '%Calculus%'`

3.	`SELECT *`
	`FROM Class`
	`INNER JOIN Teacher ON Teacher.teacher_id = Class.teacher_id`

4.	`SELECT Class.name as class_name, Teacher.name as teacher_name`
	`FROM Class`
	`INNER JOIN Teacher ON Teacher.teacher_id = Class.teacher_id`

5.	`SELECT * FROM Student` `WHERE gpa >= 3.0`

6.	`SELECT *`
	`FROM Student AS s`
	`INNER JOIN ClassStudent AS cs ON cs.student_id = s.student_id`
	`INNER JOIN Class AS c ON cs.class_id = c.class_id`
	`WHERE c.name = "Calculus I"`

Database class PHP code

```
<?
class Database
{
	private $hostname = '';
	private $database = '';
	private $username = '';
	private $password = '';
	private $dbHandler = null;
	private $debug = true;

	function __construct()
	{
		$this->OutputDebug( "Constructor" );
		$connection = "mysql:host=" . $this->hostname . ";";
		$connection .= "dbname=" . $this->database;

		try
		{
			$this->dbHandler = new PDO(
				$connection,
				$this->username,
				$this->password
			);
		}
		catch( PDOException $error )
		{
			$this->OutputError( $error->getMessage() );
		}
	}

	function OutputError( $error )
	{
		echo( "<div class='error'>" );
		print_r( $error );
		echo( "</div>" );
	}

	function OutputDebug( $message )
	{
		if ( $this->debug )
		{
			echo( "<pre>" );
			print_r( $message );
			echo( "</pre>" );
		}
	}

	private function SubmitQuery( $query )
	{
		$this->OutputDebug( $query );

		$rows = array();
```

```
            foreach( $this->dbHandler->query( $query ) as $row )
            {
                  array_push( $rows, $row );
            }

            if ( $this->dbHandler->errorInfo()[2] != "" )
            {
                  $this->OutputError( $this->dbHandler->errorInfo() );
            }

            $this->OutputDebug( $rows );
            return $rows;
      }

      private function Query( $query )
      {
            $this->OutputDebug( "Insert" );
            $this->OutputDebug( $query );

            $result = $this->dbHandler->exec( $query );

            $this->OutputDebug( $this->dbHandler->errorInfo() );
            $this->OutputDebug( $result );
      }
}
?>
```