



ALL2

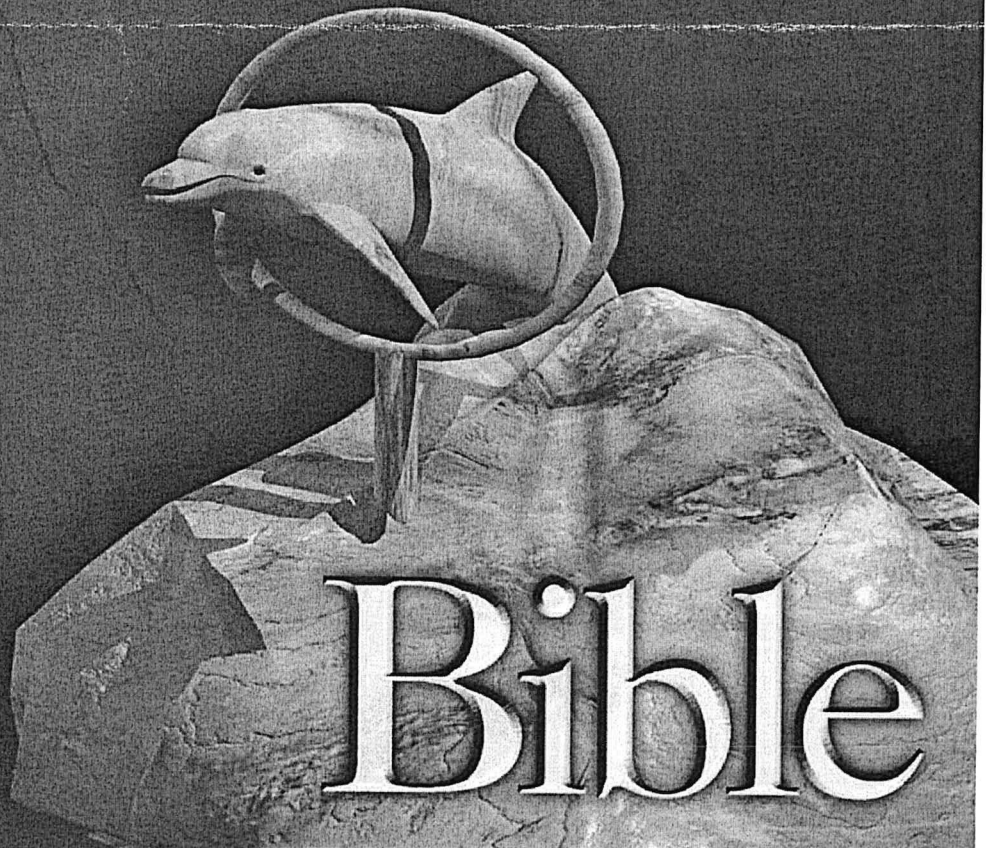
Steve Suehring, Tim Converse, and Joyce Park

# PHP6 and MySQL<sup>®</sup>

Explore PHP syntax,  
datatypes, and functions

Create database-driven,  
dynamic Web sites

Master server-side  
Web programming



The book you need to succeed!

[www.freepdf-books.com](http://www.freepdf-books.com)

This example retrieves all fields from the table `mytable` where the ID column value is less than the integer 100. WHERE clauses can get quite complex, and, frequently, multiple conditions are used together with the AND keyword.

## Joins

Joins are one of the main useful features of SQL.

A SELECT statement on a single table without joins might be visualized as being something like a row in a spreadsheet. But an SQL database is by definition relational. To understand the philosophy behind the relational database concept, you have to think back to some occasion on which you were forced to fill out a whole bunch of forms — such as applying for a loan, visiting a doctor's office for the first time, or dealing with some kind of governmental formality. (If you've never had this experience, it's because you're young enough to have lived entirely in a world of relational databases.) As you were writing down your name, address, phone, and Social Security number for the 15<sup>th</sup> time, you probably thought, "Why can't I just write my address down once, and then they could just look it up on a need-to-know basis?" That's exactly the concept behind a relational database.

The way a relational database differs from paper forms is the main identifier. Humans do well with text and prefer to categorize by textual identifiers such as names. If a dentist's office or auto body shop stored its paper files in numerical order, it would be difficult for anyone to lay his hands on John Johnson's forms when John next required service. Frankly, most paper file users these days ask for your Social Security number as a backup — it works solely to differentiate you from other people in their files with exactly the same first, last, and middle names.

Databases, on the other hand, work well with integers. You'll frequently use integer values to create unique identifiers or IDs within a database table. This field or column is then called a primary key, which indicates that each value in that column will be unique and that the rows within that column will always have a value in the primary key field. Because primary keys are unique by nature, a database needs only one to identify a person, place, or thing uniquely — no matter how many tables refer to that piece of information.

So instead of needing to repeat information several times, like this:

```
Name: John Johnson  
SS#: 123-45-6789
```

```
Name: John Johnson  
Fears: Cats, Friday the 13th, Flying
```

```
Name: Jane Jones  
SS#: 987-65-4321
```

```
Name: Jane Jones  
Fears: Heights, Flying
```

with a relational database you can write down each piece of information just once and then relate it to each other piece using integers, as shown in Tables 13-1 to 13-3.