MySQL for Developers

Duration: 5 Days

What you will learn

This MySQL for Developers training teaches developers how to develop console and web applications using MySQL with their choice of the PHP, Java, or Python programming languages. Expert Oracle University instructors will guide you through realistic hands-on activities to teach you how to use Connectors to access MySQL databases, query the database effectively, present data in different formats, and use MySQL support for “NoSQL”.

Learn To:

Program with Connectors.

Write console and web applications.

Optimize query performance.

Write stored routines and triggers.

Use the InnoDB memcached plugin for NoSQL access to your data.

Store, access, and analyze geospatial data.

Benefits to You

This course will teach you how to write applications that maximize your investment in MySQL. You will learn best practice techniques for writing programs that store and retrieve MySQL data and present it to users in a way that allows them to gain insight into that data.

Audience

Application Developers
Data Analyst
Developer
Support Engineer

Related Training

Required Prerequisites

Knowledgable in relational database concepts and MySQL data types

Knowledgable in using the mysql command-line client

Knowledgable in Joining tables

Knowledgable in executing basic DDL and DML queries using SQL

Copyright © 2013, Oracle. All rights reserved.

Suggested Prerequisites
Basic HTML

Developing simple web applications using PHP, Java/JSTL, or

Knowledge of JSON (javascript object notation)

MySQL Performance Tuning

MySQL Performance Tuning Ed 3

MySQL for Database Administrators Ed 3.1

Course Objectives

Secure your connections to the MySQL server

Use prepared statements

Guard against SQL injection

Investigate and handle errors and warnings

Create database-driven web applications

Enable fast text search

Respond to data changes and events

Modify table data

Use temporal and numeric functions in expressions

Write queries that contain nested queries

Safeguard concurrent queries by using transactions

Find and fix poorly-performing queries

Use NoSQL and JSON document stores

Write stored programs

Create MySQL client programs using Connectors

Analyze spatial data

Course Topics

Copyright © 2013, Oracle. All rights reserved.

Introduction to MySQL
MySQL overview
MySQL Enterprise Edition
MySQL on the Web
The MySQL community

Connectors and APIs
Introducing MySQL Connectors
PHP Connectors
Connector/J
Connector/Python
Embedding MySQL in an application
Integration with MySQL Fabric

Using Connectors
PHP, Java, and Python Connectors
Reusing connections
Dealing with special characters and null values
Storing and retrieving connection details
Creating secure connections to the MySQL server

Prepared Statements
Reasons for using prepared statements
User-defined variables
Preparing, executing, and deallocating prepared statements
Using prepared statements with Connectors

Handling Errors and Warnings
SQL modes
Interpreting errors and warnings
MySQL diagnostics
Error and exception handling with Connectors

Building Database-Driven Web Applications
Anatomy of a database-driven web application
Different methods of displaying query results
Web forms and processing requests
Using hyperlinks for navigation
Displaying query results across multiple pages
Dynamic sorting of query results

Tables and Views
Creating new tables from existing tables
Temporary tables
Using mysqldump
Working with views

Working with Strings
Overview of working with strings
String functions
Sorting and comparing strings
Matching string patterns with regular expressions

Copyright © 2013, Oracle. All rights reserved.

Using FULLTEXT search

Working with Numeric and Temporal Data
Numeric expressions
Temporal expressions
Interval arithmetic
Numeric functions
Temporal functions

Subqueries
Subquery: overview
Scalar subqueries
Row subqueries
Table subqueries
Correlated and non-correlated subqueries
Subqueries in data modification statements

Modifying Table Data
Using the INSERT statement
Retrieving the ID of the last-inserted record
Using the DELETE statement
Using the UPDATE and REPLACE statements
Using the TRUNCATE statement

Transactions
Overview of transactions
Transactional statements
Using transactions within programs
Consistency issues
Isolation levels
Locking

Query Optimization
Optimization strategies
The MySQL Query Optimizer
Indexing
Query analysis with the EXPLAIN statement
Rewriting queries
MySQL Enterprise Monitor

Stored Routines
Creating stored routines
Variables in stored routines
Parameters in stored routines
Managing stored routines in the database
Using flow control statements and constructs
Using cursors
Limitations of stored routines

Triggers and Scheduled Events
Triggers
Scheduled Events

Copyright © 2013, Oracle. All rights reserved.

Reporting

Aggregating data
Summary tables
Grouping data
Crosstab reports
Bar charts
Decision tables

NoSQL

NoSQL in MySQL
Memcached plugin for InnoDB
Memcached application development
The JSON native data type
JSON functions
Generated columns

Spatial Data

Spatial data support in MySQL
The MySQL spatial data types
Spatial data formats
Using spatial indexes for analysis
Useful spatial data functions

Conclusion