

1. Record Nr.	UNINA9910971354203321
Autore	Lukaszewski Albert
Titolo	MySQL for Python : Database Access Made Easy
Pubbl/distr/stampa	Birmingham, : Packt Publishing, 2010
ISBN	9781849510196 1849510199
Edizione	[1st edition]
Descrizione fisica	1 online resource (440 p.)
Collana	Community experience distilled
Disciplina	005.2 005.2/76 005.7585
Soggetti	Internet programming Web sites -- Design Web sites
Lingua di pubblicazione	Inglese
Formato	Materiale a stampa
Livello bibliografico	Monografia
Note generali	Description based upon print version of record.
Nota di contenuto	Cover; Copyright; Credits; About the Author; About the Reviewers; Table of Contents; Preface; Chapter 1: Getting Up and Running with MySQL for Python; Getting MySQL for Python; Using a package manager (only on Linux); Using RPMs and yum; Using RPMs and urpm; Using apt tools on Debian-like systems; Using an installer for Windows; Using an egg file; Using a tarball (tar.gz file); Importing MySQL for Python; Accessing online help when you need it; MySQLdb; _mysql; Connecting with a database; Creating a connection object; Creating a cursor object; Interacting with the database Closing the connectionMultiple database connections; Summary; Chapter 2: Simple Querying; A brief introduction to CRUD; Forming a query in MySQL; SELECT; * (asterisk); FROM; staff; ; (semicolon); Other helpful quantifiers; WHERE; GROUP BY; HAVING; ORDER BY; LIMIT; INTO OUTFILE; Passing a query to MySQL; A simple SELECT statement; Modifying the results; Using user-defined variables; Determining characteristics of a database and its tables; Determining what tables exist; Assigning each table a number; Offering the options to the user; Allowing the user to detail a search query Changing queries dynamicallyPattern matching in MySQL queries;

Putting it into practice; Project: a command-line search utility;
Preparing a database for searching; Planning your work, then working
your plan; Develop a well-abstracted search functionality.; Specifying
the search term from the command-line; Implementing and
incorporating the other functions: -t, -f, and -o; Including an option
for an output file; Room to grow; Summary; Chapter 3: Simple
Insertion; Forming a MySQL insertion statement; INSERT; INTO; Table
name; Column names; VALUES; ; ; (semicolon)
Helpful ways to nuance an INSERT statement INSERT...SELECT...; INSERT
DELAYED...; INSERT...ON DUPLICATE KEY UPDATE...; Passing an
insertion through MySQL for Python; Setting up the preliminaries; A
simple INSERT statement; More complex INSERT commands; Using
user-defined variables; Using metadata; Querying the database for its
structure; Retrieving the table structure; Changing insertion values
dynamically; Validating the value of name; Validating the value of price;
Querying the user for a correction; Passing fish and price for validation;
Essentials: close and commit; In need of some closure
What happened to commit? Why are these essentials non-essential?;
Project: a command-line insertion utility; The necessary modules; The
main() thing; Coding the flag system; Testing the values passed by the
user; Try to establish a database connection; Showing the tables;
Showing the table structure, if desired; Accepting user input for the
INSERT statement; Building the INSERT statement from the user input
and execute it; Committing changes and closing the connection;
Coding the other functions; valid_digit() and valid_string(); valid_table();
query(); Calling main(); Room to grow; Summary
Chapter 4: Exception Handling

Sommario/riassunto

This is a practical, tutorial-style book that includes many examples to demonstrate the full potential of MySQL for Python. Every chapter starts with an explanation of the various areas for using MySQL for Python and ends with work on a sample application using the programming calls just learned. All complicated concepts are broken down to be very easy to understand. Everything in the book is designed to help you learn and use MySQL for Python to address your programming needs in the fastest way possible. This book is meant for intermediate users of Python who want hassle-free access to their
