Matlab Gui Tutorial For Beginners Blinkdagger

This step-by-step guide to explore database programming using Java is ideal for people with little or no programming experience. The goal of this concise book is not just to teach you Java, but to help you think like a programmer. Each brief chapter covers the material for one week of a college course to help you practice what you've learned. As you would expect, this book shows how to build from scratch two different databases: MariaDB and SQLite using Java. In designing a GUI and as an IDE, you will make use of the NetBeans tool. In the first chapter, you will learn the basics of cryptography using Java. Here, you will learn how to write a Java program to count Hash, MAC (Message Authentication Code), store keys in a KeyStore, generate PrivateKey and PublicKey, encrypt / decrypt data, and generate and verify digital prints. In the second chapter, you will learn how to create and store salt passwords and verify them. You will create a Login table. In this case, you will see how to create a Java GUI using NetBeans to implement it. In addition to the Login table, in this chapter you will also create a Client table. In the case of the Client table, you will learn how to generate and save public and private keys into a database. You will also learn

how to encrypt / decrypt data and save the results into a database. In the third chapter, you will create an Account table. This account table has the following ten fields: account_id (primary key), client_id (primarykey), account_number, account_date, account_type, plain_balance, cipher_balance, decipher_balance, digital_signature, and signature_verification. In this case, you will learn how to implement generating and verifying digital prints and storing the results into a database. In the fourth chapter, You create a table with the name of the Account, which has ten columns: account_id (primary key), client_id (primarykey), account_number, account_date, account_type, plain_balance, cipher_balance, decipher_balance, digital_signature, and signature_verification. In the fifth chapter, you will create a Client_Data table, which has the following seven fields: client_data_id (primary key), account_id (primary_key), birth_date, address, mother_name, telephone, and photo_path. In chapter six, you will be shown how to create SQLite database and tables with Java. In chapter seven, you will be taught how to extract image features, utilizing BufferedImage class, in Java GUI. Digital image techniques to extract image features used in this chapted are grascaling, sharpening, invertering, blurring, dilation, erosion, closing, opening, vertical prewitt, horizontal prewitt, Laplacian, horizontal sobel, and vertical sobel. For

readers, you can develop it to store other advanced image features based on descriptors such as SIFT and others for developing descriptor based matching. In chapter eight, you will be taught to create Java GUI to view, edit, insert, and delete Suspect table data. This table has eleven columns: suspect_id (primary key), suspect_name, birth_date, case_date, report_date, suspect_ status, arrest_date, mother_name, address, telephone, and photo. In chapter nine, you will be taught to create Java GUI to view, edit, insert, and delete Feature_Extraction table data. This table has eight columns: feature_id (primary key), suspect_id (foreign key), feature1, feature2, feature3, feature4, feature5, and feature6. All six fields (except keys) will have a BLOB data type, so that the image of the feature will be directly saved into this table. In chapter ten, you will add two tables: Police_Station and Investigator. These two tables will later be joined to Suspect table through another table, File_Case, which will be built in the seventh chapter. The Police_Station has six columns: police_station_id (primary key), location, city, province, telephone, and photo. The Investigator has eight columns: investigator_id (primary key), investigator_name, rank, birth_date, gender, address, telephone, and photo. Here, you will design a Java GUI to display, edit, fill, and delete data in both tables. In chapter eleven, you will add two tables: Victim and

Case_File. The File_Case table will connect four other tables: Suspect, Police_Station, Investigator and Victim. The Victim table has nine columns: victim_id (primary key), victim_name, crime_type, birth_date, crime_date, gender, address, telephone, and photo. The Case_File has seven columns: case_file_id (primary key), suspect_id (foreign key), police_station_id (foreign key), investigator_id (foreign key), victim_id (foreign key), status, and description. Here, you will also design a Java GUI to display, edit, fill, and delete data in both tables. Finally, this book is hopefully useful and can improve database programming skills for every Java/MariaDB/SQLite pogrammer.

Whether you're building GUI prototypes or full-fledged cross-platform GUI applications with native look-and-feel, PyQt 4 is your fastest, easiest, most powerful solution. Qt expert Mark Summerfield has written the definitive best-practice guide to PyQt 4 development. With Rapid GUI Programming with Python and Qt you'll learn how to build efficient GUI applications that run on all major operating systems, including Windows, Mac OS X, Linux, and many versions of Unix, using the same source code for all of them. Summerfield systematically introduces every core GUI development technique: from dialogs and windows to data handling; from events to printing; and more. Through the book's realistic examples you'll discover a completely new PyQt

4-based programming approach, as well as coverage of many new topics, from PyQt 4's rich text engine to advanced model/view and graphics/view programming. Every key concept is illuminated with realistic, downloadable examples–all tested on Windows, Mac OS X, and Linux with Python 2.5, Qt 4.2, and PyQt 4.2, and on Windows and Linux with Qt 4.3 and PyQt 4.3.

In this book, you will create three Java GUI applications using MySQL, MariaDB, and PostgreSQL. In this book, you will learn how to build from scratch a database management system using Java. In designing a GUI and as an IDE, you will make use of the NetBeans tool. Gradually and step by step, you will be taught how to utilize three different databases in Java. In chapter one, you will create School database and its six tables. In chapter two, you will study: Creating the initial three table projects in the school database: Teacher table, TClass table, and Subject table; Creating database configuration files; Creating a Java GUI for viewing and navigating the contents of each table; Creating a Java GUI for inserting and editing tables; and Creating a Java GUI to join and query the three tables. In chapter three, you will learn: Creating the main form to connect all forms; Creating a project will add three more tables to the school database: the Student table, the Parent table, and Tuition table; Creating a Java GUI to view and navigate the

contents of each table; Creating a Java GUI for editing, inserting, and deleting records in each table; Creating a Java GUI to join and query the three tables and all six. In chapter four, you will study how to query the six tables. In chapter five, you will learn the basics of cryptography using Java. Here, you will learn how to write a Java program to count Hash, MAC (Message Authentication Code), store keys in a KeyStore, generate PrivateKey and PublicKey, encrypt / decrypt data, and generate and verify digital prints. In chapter six, you will create Bank database and its tables. In chapter seven, you will learn how to create and store salt passwords and verify them. You will create a Login table. In this case, you will see how to create a Java GUI using NetBeans to implement it. In addition to the Login table, in this chapter you will also create a Client table. In the case of the Client table, you will learn how to generate and save public and private keys into a database. You will also learn how to encrypt / decrypt data and save the results into a database. In chapter eight, you will create an Account table. This account table has the following ten fields: account_id (primary key), client_id (primarykey), account_number, account_date, account_type, plain_balance, cipher_balance, decipher_balance, digital_signature, and signature_verification. In this case, you will learn how to implement generating and verifying digital prints and storing the results into

a database. In chapter nine, you will create a
Client_Data table, which has the following seven
fields: client_data_id (primary key), account_id
(primary_key), birth_date, address, mother_name,
telephone, and photo_path. In chapter ten, you will
be taught how to create Crime database and its
tables. In chapter eleven, you will be taught how to
extract image features, utilizing BufferedImage class,
in Java GUI. In chapter twelve, you will be taught to
create Java GUI to view, edit, insert, and delete
Suspect table data. This table has eleven columns:
suspect_id (primary key), suspect_name, birth_date,
case_date, report_date, suspect_ status,
arrest_date, mother_name, address, telephone, and
photo. In chapter thirteen, you will be taught to
create Java GUI to view, edit, insert, and delete
Feature_Extraction table data. This table has eight
columns: feature_id (primary key), suspect_id
(foreign key), feature1, feature2, feature3, feature4,
feature5, and feature6. In chapter fourteen, you will
add two tables: Police_Station and Investigator.
These two tables will later be joined to Suspect table
through another table, File_Case. The
Police_Station has six columns: police_station_id
(primary key), location, city, province, telephone, and
photo. The Investigator has eight columns:
investigator_id (primary key), investigator_name,
rank, birth_date, gender, address, telephone, and
photo. Here, you will design a Java GUI to display,

edit, fill, and delete data in both tables. In chapter
fifteen, you will add two tables: Victim and
File_Case. The File_Case table will connect four
other tables: Suspect, Police_Station, Investigator
and Victim. The Victim table has nine columns:
victim_id (primary key), victim_name, crime_type,
birth_date, crime_date, gender, address, telephone,
and photo. The File_Case has seven columns:
file_case_id (primary key), suspect_id (foreign key),
police_station_id (foreign key), investigator_id
(foreign key), victim_id (foreign key), status, and
description. Here, you will also design a Java GUI to
display, edit, fill, and delete data in both tables.
For a variety of reasons, the MATLAB®-Java
interface was never fully documented. This is really
quite unfortunate: Java is one of the most widely
used programming languages, having many times
the number of programmers and programming
resources as MATLAB. Also unfortunate is the
popular claim that while MATLAB is a fine
programming platform for prototyping, it is not
suitable for real-world, modern-looking applications.
Undocumented Secrets of MATLAB®-Java
Programming aims to correct this misconception.
This book shows how using Java can significantly
improve MATLAB program appearance and
functionality, and that this can be done easily and
even without any prior Java knowledge. Readers are
led step-by-step from simple to complex

customizations. Code snippets, screenshots, and
numerous online references are provided to enable
the utilization of this book as both a sequential
tutorial and as a random-access reference suited for
immediate use. Java-savvy readers will find it easy
to tailor code samples for their particular needs; for
Java newcomers, an introduction to Java and
numerous online references are provided. This book
demonstrates how The MATLAB programming
environment relies on Java for numerous tasks,
including networking, data-processing algorithms
and graphical user-interface (GUI) We can use
MATLAB for easy access to external Java
functionality, either third-party or user-created Using
Java, we can extensively customize the MATLAB
environment and application GUI, enabling the
creation of visually appealing and usable
applications

This book explains relational theory in practice, and
demonstrates through two projects how you can
apply it to your use of MySQL and SQLite
databases. This book covers the important
requirements of teaching databases with a practical
and progressive perspective. This book offers the
straightforward, practical answers you need to help
you do your job. This hands-on
tutorial/reference/guide to MySQL and SQLite is not
only perfect for students and beginners, but it also
works for experienced developers who aren't getting

the most from both databases. In designing a GUI
and as an IDE, you will make use Qt Designer. In the
first chapter, you will learn to use several widgets in
PyQt5: Display a welcome message; Use the Radio
Button widget; Grouping radio buttons; Displays
options in the form of a check box; and Display two
groups of check boxes. In chapter two, you will learn
to use the following topics: Using Signal / Slot Editor;
Copy and place text from one Line Edit widget to
another; Convert data types and make a simple
calculator; Use the Spin Box widget; Use scrollbars
and sliders; Using the Widget List; Select a number
of list items from one Widget List and display them
on another Widget List widget; Add items to the
Widget List; Perform operations on the Widget List;
Use the Combo Box widget; Displays data selected
by the user from the Calendar Widget; Creating a
hotel reservation application; and Display tabular
data using Table Widgets. In chapter three, you will
learn: How to create the initial three tables project in
the School database: Teacher, Class, and Subject
tables; How to create database configuration files;
How to create a Python GUI for inserting and editing
tables; How to create a Python GUI to join and query
the three tables. In chapter four, you will learn how
to: Create a main form to connect all forms; Create a
project will add three more tables to the school
database: Student, Parent, and Tuition tables;
Create a Python GUI for inserting and editing tables;

Create a Python GUI to join and query over the three
tables. In chapter five, you will join the six classes,
Teacher, TClass, Subject, Student, Parent, and
Tuition and make queries over those tables. In
chapter six, you will create dan configure database.
In this chapter, you will create Suspect table in crime
database. This table has eleven columns: suspect_id
(primary key), suspect_name, birth_date, case_date,
report_date, suspect_ status, arrest_date,
mother_name, address, telephone, and photo. You
will also create GUI to display, edit, insert, and
delete for this table. In chapter seven, you will create
a table with the name Feature_Extraction, which has
eight columns: feature_id (primary key), suspect_id
(foreign key), feature1, feature2, feature3, feature4,
feature5, and feature6. The six fields (except keys)
will have VARBINARY(MAX) data type. You will also
create GUI to display, edit, insert, and delete for this
table. In chapter eight, you will create two tables,
Police and Investigator. The Police table has six
columns: police_id (primary key), province, city,
address, telephone, and photo. The Investigator
table has eight columns: investigator_id (primary
key), investigator_name, rank, birth_date, gender,
address, telephone, and photo. You will also create
GUI to display, edit, insert, and delete for both
tables. In the last chapter, you will create two tables,
Victim and Case_File. The Victim table has nine
columns: victim_id (primary key), victim_name,

crime_type, birth_date, crime_date, gender, address, telephone, and photo. The Case_File table has seven columns: case_file_id (primary key), suspect_id (foreign key), police_id (foreign key), investigator_id (foreign key), victim_id (foreign key), status, and description. You will create GUI to display, edit, insert, and delete for both tables.

In this book, you will learn how to build from scratch a SQLite database management system using Java. In designing a GUI and as an IDE, you will make use of the NetBeans tool. Gradually and step by step, you will be taught how to use SQLite in Java. In the first chapter, you will learn: How to create SQLite database and six tables In the second chapter, you will study: Creating the initial three table projects in the school database: Teacher table, TClass table, and Subject table; Creating database configuration files; Creating a Java GUI for viewing and navigating the contents of each table; Creating a Java GUI for inserting and editing tables; and Creating a Java GUI to join and query the three tables. In the third chapter, you will learn: Creating the main form to connect all forms; Creating a project will add three more tables to the school database: the Student table, the Parent table, and Tuition table; Creating a Java GUI to view and navigate the contents of each table; Creating a Java GUI for editing, inserting, and deleting records in each table; Creating a Java GUI to join and query the three tables and all six tables.

In the last chapter, you will study how to query the six tables. Finally, this book is hopefully useful and can improve database programming skills for every Java/SQLite programmer.

Introduction to Modeling and Simulation with MATLAB and Python is intended for students and professionals in science, social science, and engineering that wish to learn the principles of computer modeling, as well as basic programming skills. The book content focuses on meeting a set of basic modeling and simulation competencies that were developed as part of several National Science Foundation grants. Even though computer science students are much more expert programmers, they are not often given the opportunity to see how those skills are being applied to solve complex science and engineering problems and may also not be aware of the libraries used by scientists to create those models. The book interleaves chapters on modeling concepts and related exercises with programming concepts and exercises. The authors start with an introduction to modeling and its importance to current practices in the sciences and engineering. They introduce each of the programming environments and the syntax used to represent variables and compute mathematical equations and functions. As students gain more programming expertise, the authors return to modeling concepts, providing starting code for a variety of exercises

where students add additional code to solve the problem and provide an analysis of the outcomes. In this way, the book builds both modeling and programming expertise with a "just-in-time" approach so that by the end of the book, students can take on relatively simple modeling example on their own. Each chapter is supplemented with references to additional reading, tutorials, and exercises that guide students to additional help and allows them to practice both their programming and analytical modeling skills. In addition, each of the programming related chapters is divided into two parts – one for MATLAB and one for Python. In these chapters, the authors also refer to additional online tutorials that students can use if they are having difficulty with any of the topics. The book culminates with a set of final project exercise suggestions that incorporate both the modeling and programming skills provided in the rest of the volume. Those projects could be undertaken by individuals or small groups of students. The companion website at http://www.intromodeling.com provides updates to instructions when there are substantial changes in software versions, as well as electronic copies of exercises and the related code. The website also offers a space where people can suggest additional projects they are willing to share as well as comments on the existing projects and exercises throughout the book. Solutions and lecture notes will

also be available for qualifying instructors.
In this book, you will create two desktop applications
using Python GUI and PostgreSQL. This book is a
Python/PostgreSQL version of the Python/MySQL
book which was written by the author. What
underlies the writing of this book is the growing
popularity of the PostgreSQL database server lately
and more and more programmers migrating from
MySQL to PostgreSQL. In this book, you will learn to
build a school database project, step by step. A
number of widgets from PyQt will be used for the
user interface. In the first and second chapter, you
will get introduction of postgresql. And then, you will
learn querying data from the postgresql using Python
including establishing a database connection,
creating a statement object, executing the query,
processing the resultset object, querying data using
a statement that returns multiple rows, querying data
using a statement that has parameters, inserting
data into a table using Python, updating data in
postgresql database using Python, calling postgresql
stored function using Python, deleting data from a
postgresql table using Python, and postgresql
Python transaction. In the fourth chapter, you will
study: Creating the initial three table in the School
database project: Teacher table, Class table, and
Subject table; Creating database configuration files;
Creating a Python GUI for viewing and navigating
the contents of each table. Creating a Python GUI

for inserting and editing tables; and Creating a
Python GUI to merge and query the three tables. In
chapter five, you will learn: Creating the main form to
connect all forms; Creating a project that will add
three more tables to the school database: the
Student table, the Parent table, and the Tuition table;
Creating a Python GUI to view and navigate the
contents of each table; Creating a Python GUI for
editing, inserting, and deleting records in each table;
Create a Python GUI to merge and query the three
tables and all six tables. In chapter six, you will
create dan configure PotgreSQL database. In this
chapter, you will create Suspect table in crime
database. This table has eleven columns: suspect_id
(primary key), suspect_name, birth_date, case_date,
report_date, suspect_ status, arrest_date,
mother_name, address, telephone, and photo. You
will also create GUI to display, edit, insert, and
delete for this table. In chapter seven, you will create
a table with the name Feature_Extraction, which has
eight columns: feature_id (primary key), suspect_id
(foreign key), feature1, feature2, feature3, feature4,
feature5, and feature6. The six fields (except keys)
will have a VARCHAR data type (200). You will also
create GUI to display, edit, insert, and delete for this
table. In chapter eight, you will create two tables,
Police and Investigator. The Police table has six
columns: police_id (primary key), province, city,
address, telephone, and photo. The Investigator

table has eight columns: investigator_id (primary key), investigator_name, rank, birth_date, gender, address, telephone, and photo. You will also create GUI to display, edit, insert, and delete for both tables. In chapter nine, you will create two tables, Victim and Case_File. The Victim table has nine columns: victim_id (primary key), victim_name, crime_type, birth_date, crime_date, gender, address, telephone, and photo. The Case_File table has seven columns: case_file_id (primary key), suspect_id (foreign key), police_id (foreign key), investigator_id (foreign key), victim_id (foreign key), status, and description. You will create GUI to display, edit, insert, and delete for both tables as well.

In this book, you will learn how to use NumPy, Pandas, OpenCV, Scikit-Learn and other libraries to how to plot graph and to process digital image. Then, you will learn how to classify features using Perceptron, Adaline, Logistic Regression (LR), Support Vector Machine (SVM), Decision Tree (DT), Random Forest (RF), and K-Nearest Neighbor (KNN) models. You will also learn how to extract features using Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), Kernel Principal Component Analysis (KPCA) algorithms and use them in machine learning. In Chapter 1, you will learn: Tutorial Steps To Create A Simple GUI Application, Tutorial Steps to Use Radio Button, Tutorial Steps to Group Radio Buttons, Tutorial Steps to Use CheckBox Widget, Tutorial Steps to Use Two CheckBox

Groups, Tutorial Steps to Understand Signals and Slots,
Tutorial Steps to Convert Data Types, Tutorial Steps to
Use Spin Box Widget, Tutorial Steps to Use ScrollBar
and Slider, Tutorial Steps to Use List Widget, Tutorial
Steps to Select Multiple List Items in One List Widget
and Display It in Another List Widget, Tutorial Steps to
Insert Item into List Widget, Tutorial Steps to Use
Operations on Widget List, Tutorial Steps to Use Combo
Box, Tutorial Steps to Use Calendar Widget and Date
Edit, and Tutorial Steps to Use Table Widget. In Chapter
2, you will learn: Tutorial Steps To Create A Simple Line
Graph, Tutorial Steps To Create A Simple Line Graph in
Python GUI, Tutorial Steps To Create A Simple Line
Graph in Python GUI: Part 2, Tutorial Steps To Create
Two or More Graphs in the Same Axis, Tutorial Steps To
Create Two Axes in One Canvas, Tutorial Steps To Use
Two Widgets, Tutorial Steps To Use Two Widgets, Each
of Which Has Two Axes, Tutorial Steps To Use Axes
With Certain Opacity Levels, Tutorial Steps To Choose
Line Color From Combo Box, Tutorial Steps To Calculate
Fast Fourier Transform, Tutorial Steps To Create GUI
For FFT, Tutorial Steps To Create GUI For FFT With
Some Other Input Signals, Tutorial Steps To Create GUI
For Noisy Signal, Tutorial Steps To Create GUI For
Noisy Signal Filtering, and Tutorial Steps To Create GUI
For Wav Signal Filtering. In Chapter 3, you will learn:
Tutorial Steps To Convert RGB Image Into Grayscale,
Tutorial Steps To Convert RGB Image Into YUV Image,
Tutorial Steps To Convert RGB Image Into HSV Image,
Tutorial Steps To Filter Image, Tutorial Steps To Display
Image Histogram, Tutorial Steps To Display Filtered

Image Histogram, Tutorial Steps To Filter Image With CheckBoxes, Tutorial Steps To Implement Image Thresholding, and Tutorial Steps To Implement Adaptive Image Thresholding. You will also learn: Tutorial Steps To Generate And Display Noisy Image, Tutorial Steps To Implement Edge Detection On Image, Tutorial Steps To Implement Image Segmentation Using Multiple Thresholding and K-Means Algorithm, Tutorial Steps To Implement Image Denoising, Tutorial Steps To Detect Face, Eye, and Mouth Using Haar Cascades, Tutorial Steps To Detect Face Using Haar Cascades with PyQt, Tutorial Steps To Detect Eye, and Mouth Using Haar Cascades with PyQt, Tutorial Steps To Extract Detected Objects, Tutorial Steps To Detect Image Features Using Harris Corner Detection, Tutorial Steps To Detect Image Features Using Shi-Tomasi Corner Detection, Tutorial Steps To Detect Features Using Scale-Invariant Feature Transform (SIFT), and Tutorial Steps To Detect Features Using Features from Accelerated Segment Test (FAST). In Chapter 4, In this tutorial, you will learn how to use Pandas, NumPy and other libraries to perform simple classification using perceptron and Adaline (adaptive linear neuron). The dataset used is Iris dataset directly from the UCI Machine Learning Repository. You will learn: Tutorial Steps To Implement Perceptron, Tutorial Steps To Implement Perceptron with PyQt, Tutorial Steps To Implement Adaline (ADAptive LInear NEuron), and Tutorial Steps To Implement Adaline with PyQt. In Chapter 5, you will learn how to use the scikit-learn machine learning library, which provides a wide variety of machine learning algorithms via a user-friendly Python

API and to perform classification using perceptron, Adaline (adaptive linear neuron), and other models. The dataset used is Iris dataset directly from the UCI Machine Learning Repository. You will learn: Tutorial Steps To Implement Perceptron Using Scikit-Learn, Tutorial Steps To Implement Perceptron Using Scikit-Learn with PyQt, Tutorial Steps To Implement Logistic Regression Model, Tutorial Steps To Implement Logistic Regression Model with PyQt, Tutorial Steps To Implement Logistic Regression Model Using Scikit-Learn with PyQt, Tutorial Steps To Implement Support Vector Machine (SVM) Using Scikit-Learn, Tutorial Steps To Implement Decision Tree (DT) Using Scikit-Learn, Tutorial Steps To Implement Random Forest (RF) Using Scikit-Learn, and Tutorial Steps To Implement K-Nearest Neighbor (KNN) Using Scikit-Learn. In Chapter 6, you will learn how to use Pandas, NumPy, Scikit-Learn, and other libraries to implement different approaches for reducing the dimensionality of a dataset using different feature selection techniques. You will learn about three fundamental techniques that will help us to summarize the information content of a dataset by transforming it onto a new feature subspace of lower dimensionality than the original one. Data compression is an important topic in machine learning, and it helps us to store and analyze the increasing amounts of data that are produced and collected in the modern age of technology. You will learn the following topics: Principal Component Analysis (PCA) for unsupervised data compression, Linear Discriminant Analysis (LDA) as a supervised dimensionality reduction technique for maximizing class

separability, Nonlinear dimensionality reduction via
Kernel Principal Component Analysis (KPCA). You will
learn: 6.1 Tutorial Steps To Implement Principal
Component Analysis (PCA), Tutorial Steps To Implement
Principal Component Analysis (PCA) Using Scikit-Learn,
Tutorial Steps To Implement Principal Component
Analysis (PCA) Using Scikit-Learn with PyQt, Tutorial
Steps To Implement Linear Discriminant Analysis (LDA),
Tutorial Steps To Implement Linear Discriminant
Analysis (LDA) with Scikit-Learn, Tutorial Steps To
Implement Linear Discriminant Analysis (LDA) Using
Scikit-Learn with PyQt, Tutorial Steps To Implement
Kernel Principal Component Analysis (KPCA) Using
Scikit-Learn, and Tutorial Steps To Implement Kernel
Principal Component Analysis (KPCA) Using Scikit-
Learn with PyQt. In Chapter 7, you will learn how to use
Keras, Scikit-Learn, Pandas, NumPy and other libraries
to perform prediction on handwritten digits using MNIST
dataset. You will learn: Tutorial Steps To Load MNIST
Dataset, Tutorial Steps To Load MNIST Dataset with
PyQt, Tutorial Steps To Implement Perceptron With PCA
Feature Extractor on MNIST Dataset Using PyQt,
Tutorial Steps To Implement Perceptron With LDA
Feature Extractor on MNIST Dataset Using PyQt,
Tutorial Steps To Implement Perceptron With KPCA
Feature Extractor on MNIST Dataset Using PyQt,
Tutorial Steps To Implement Logistic Regression (LR)
Model With PCA Feature Extractor on MNIST Dataset
Using PyQt, Tutorial Steps To Implement Logistic
Regression (LR) Model With LDA Feature Extractor on
MNIST Dataset Using PyQt, Tutorial Steps To Implement

Logistic Regression (LR) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement , Tutorial Steps To Implement Support Vector Machine (SVM) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Support Vector Machine (SVM) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Decision Tree (DT) Model With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Decision Tree (DT) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Decision Tree (DT) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Random Forest (RF) Model With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Random Forest (RF) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Random Forest (RF) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement K-Nearest Neighbor (KNN) Model With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement K-Nearest Neighbor (KNN) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, and Tutorial Steps To Implement K-Nearest Neighbor (KNN) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt.
This book is mariadb-based python programming Intentionally designed for various levels of interest and ability of learners, this book is suitable for students,

engineers, and even researchers in a variety of disciplines. No advanced programming experience is needed, and only a few school-level programming skill are needed. In the first chapter, you will learn to use several widgets in PyQt5: Display a welcome message; Use the Radio Button widget; Grouping radio buttons; Displays options in the form of a check box; and Display two groups of check boxes. In chapter two, you will learn to use the following topics: Using Signal / Slot Editor; Copy and place text from one Line Edit widget to another; Convert data types and make a simple calculator; Use the Spin Box widget; Use scrollbars and sliders; Using the Widget List; Select a number of list items from one Widget List and display them on another Widget List widget; Add items to the Widget List; Perform operations on the Widget List; Use the Combo Box widget; Displays data selected by the user from the Calendar Widget; Creating a hotel reservation application; and Display tabular data using Table Widgets. In third chapter, you will learn: How to create the initial three tables project in the School database: Teacher, Class, and Subject tables; How to create database configuration files; How to create a Python GUI for inserting and editing tables; How to create a Python GUI to join and query the three tables. In fourth chapter, you will learn how to: Create a main form to connect all forms; Create a project will add three more tables to the school database: Student, Parent, and Tuition tables; Create a Python GUI for inserting and editing tables; Create a Python GUI to join and query over the three tables. In the last chapter, you will join the six classes,

Teacher, TClass, Subject, Student, Parent, and Tuition
and make queries over those tables.
This book is a comprehensive guide to Python as one of
the fastest-growing computer languages including Web
and Internet applications. This clear and concise
introduction to the Python language is aimed at readers
who are already familiar with programming in at least
one language. This hands-on book introduces the
essential topic of coding and the Python computer
language to beginners and pogrammers of all ages. This
book explains relational theory in practice, and
demonstrates through two projects how you can apply it
to your use of PostgreSQL and SQL Server databases.
This book covers the important requirements of teaching
databases with a practical and progressive perspective.
This book offers the straightforward, practical answers
you need to help you do your job. This hands-on
tutorial/reference/guide to PostgreSQL and SQL Server
is not only perfect for students and beginners, but it also
works for experienced developers who aren't getting the
most from both databases. In designing a GUI and as an
IDE, you will make use Qt Designer. In the first chapter,
you will learn to use several widgets in PyQt5: Display a
welcome message; Use the Radio Button widget;
Grouping radio buttons; Displays options in the form of a
check box; and Display two groups of check boxes. In
chapter two, you will learn to use the following topics:
Using Signal / Slot Editor; Copy and place text from one
Line Edit widget to another; Convert data types and
make a simple calculator; Use the Spin Box widget; Use
scrollbars and sliders; Using the Widget List; Select a

number of list items from one Widget List and display
them on another Widget List widget; Add items to the
Widget List; Perform operations on the Widget List; Use
the Combo Box widget; Displays data selected by the
user from the Calendar Widget; Creating a hotel
reservation application; and Display tabular data using
Table Widgets. In chapter three, you will learn: How to
create the initial three tables project in the School
database: Teacher, Class, and Subject tables; How to
create database configuration files; How to create a
Python GUI for inserting and editing tables; How to
create a Python GUI to join and query the three tables. In
chapter four, you will learn how to: Create a main form to
connect all forms; Create a project will add three more
tables to the school database: Student, Parent, and
Tuition tables; Create a Python GUI for inserting and
editing tables; Create a Python GUI to join and query
over the three tables. In chapter five, you will join the six
classes, Teacher, TClass, Subject, Student, Parent, and
Tuition and make queries over those tables. In chapter
six, you will get introduction of postgresql. And then, you
will learn querying data from the postgresql using Python
including establishing a database connection, creating a
statement object, executing the query, processing the
resultset object, querying data using a statement that
returns multiple rows, querying data using a statement
that has parameters, inserting data into a table using
Python, updating data in postgresql database using
Python, calling postgresql stored function using Python,
deleting data from a postgresql table using Python, and
postgresql Python transaction. In chapter seven, you will

create dan configure PotgreSQL database. In this chapter, you will create Suspect table in crime database. This table has eleven columns: suspect_id (primary key), suspect_name, birth_date, case_date, report_date, suspect_ status, arrest_date, mother_name, address, telephone, and photo. You will also create GUI to display, edit, insert, and delete for this table. In chapter eight, you will create a table with the name Feature_Extraction, which has eight columns: feature_id (primary key), suspect_id (foreign key), feature1, feature2, feature3, feature4, feature5, and feature6. The six fields (except keys) will have a VARCHAR data type (200). You will also create GUI to display, edit, insert, and delete for this table. In chapter nine, you will create two tables, Police and Investigator. The Police table has six columns: police_id (primary key), province, city, address, telephone, and photo. The Investigator table has eight columns: investigator_id (primary key), investigator_name, rank, birth_date, gender, address, telephone, and photo. You will also create GUI to display, edit, insert, and delete for both tables. In chapter ten, you will create two tables, Victim and Case_File. The Victim table has nine columns: victim_id (primary key), victim_name, crime_type, birth_date, crime_date, gender, address, telephone, and photo. The Case_File table has seven columns: case_file_id (primary key), suspect_id (foreign key), police_id (foreign key), investigator_id (foreign key), victim_id (foreign key), status, and description. You will create GUI to display, edit, insert, and delete for both tables as well. This book is SQL Server-based python programming.

Microsoft SQL Server is robust relational database
management system used by so many organizations of
various sizes including top fortune 100 companies. SQL
Server is a relational database management system
(RDBMS) developed and marketed by Microsoft. As a
database server, the primary function of the SQL Server
is to store and retrieve data used by other applications.
Deliberately designed for various levels of programming
skill, this book is suitable for students, engineers, and
even researchers in various disciplines. There is no need
for advanced programming experience, and school-level
programming skills are needed. In the first chapter, you
will learn to use several widgets in PyQt5: Display a
welcome message; Use the Radio Button widget;
Grouping radio buttons; Displays options in the form of a
check box; and Display two groups of check boxes. In
chapter two, you will learn to use the following topics:
Using Signal / Slot Editor; Copy and place text from one
Line Edit widget to another; Convert data types and
make a simple calculator; Use the Spin Box widget; Use
scrollbars and sliders; Using the Widget List; Select a
number of list items from one Widget List and display
them on another Widget List widget; Add items to the
Widget List; Perform operations on the Widget List; Use
the Combo Box widget; Displays data selected by the
user from the Calendar Widget; Creating a hotel
reservation application; and Display tabular data using
Table Widgets. In third chapter, you will learn: How to
create the initial three tables project in the School
database: Teacher, Class, and Subject tables; How to
create database configuration files; How to create a

Python GUI for inserting and editing tables; How to
create a Python GUI to join and query the three tables. In
fourth chapter, you will learn how to: Create a main form
to connect all forms; Create a project will add three more
tables to the school database: Student, Parent, and
Tuition tables; Create a Python GUI for inserting and
editing tables; Create a Python GUI to join and query
over the three tables. In the last chapter, you will join the
six classes, Teacher, TClass, Subject, Student, Parent,
and Tuition and make queries over those tables.
Author Craig Lent's 1st edition of Learning to Program
with MATLAB: Building GUI Tools teaches the core
concepts of computer programming, such as arrays,
loops, function, basic data structures, etc., using
MATLAB. The text has a focus on the fundamentals of
programming and builds up to an emphasis on GUI tools,
covering text-based programs first, then programs that
produce graphics. This creates a visual expression of the
underlying mathematics of a problem or design.
BOOK 1: LEARN FROM SCRATCH MACHINE
LEARNING WITH PYTHON GUI In this book, you will
learn how to use NumPy, Pandas, OpenCV, Scikit-Learn
and other libraries to how to plot graph and to process
digital image. Then, you will learn how to classify
features using Perceptron, Adaline, Logistic Regression
(LR), Support Vector Machine (SVM), Decision Tree
(DT), Random Forest (RF), and K-Nearest Neighbor
(KNN) models. You will also learn how to extract features
using Principal Component Analysis (PCA), Linear
Discriminant Analysis (LDA), Kernel Principal
Component Analysis (KPCA) algorithms and use them in

machine learning. In Chapter 1, you will learn: Tutorial Steps To Create A Simple GUI Application, Tutorial Steps to Use Radio Button, Tutorial Steps to Group Radio Buttons, Tutorial Steps to Use CheckBox Widget, Tutorial Steps to Use Two CheckBox Groups, Tutorial Steps to Understand Signals and Slots, Tutorial Steps to Convert Data Types, Tutorial Steps to Use Spin Box Widget, Tutorial Steps to Use ScrollBar and Slider, Tutorial Steps to Use List Widget, Tutorial Steps to Select Multiple List Items in One List Widget and Display It in Another List Widget, Tutorial Steps to Insert Item into List Widget, Tutorial Steps to Use Operations on Widget List, Tutorial Steps to Use Combo Box, Tutorial Steps to Use Calendar Widget and Date Edit, and Tutorial Steps to Use Table Widget. In Chapter 2, you will learn: Tutorial Steps To Create A Simple Line Graph, Tutorial Steps To Create A Simple Line Graph in Python GUI, Tutorial Steps To Create A Simple Line Graph in Python GUI: Part 2, Tutorial Steps To Create Two or More Graphs in the Same Axis, Tutorial Steps To Create Two Axes in One Canvas, Tutorial Steps To Use Two Widgets, Tutorial Steps To Use Two Widgets, Each of Which Has Two Axes, Tutorial Steps To Use Axes With Certain Opacity Levels, Tutorial Steps To Choose Line Color From Combo Box, Tutorial Steps To Calculate Fast Fourier Transform, Tutorial Steps To Create GUI For FFT, Tutorial Steps To Create GUI For FFT With Some Other Input Signals, Tutorial Steps To Create GUI For Noisy Signal, Tutorial Steps To Create GUI For Noisy Signal Filtering, and Tutorial Steps To Create GUI For Wav Signal Filtering. In Chapter 3, you will learn:

Tutorial Steps To Convert RGB Image Into Grayscale, Tutorial Steps To Convert RGB Image Into YUV Image, Tutorial Steps To Convert RGB Image Into HSV Image, Tutorial Steps To Filter Image, Tutorial Steps To Display Image Histogram, Tutorial Steps To Display Filtered Image Histogram, Tutorial Steps To Filter Image With CheckBoxes, Tutorial Steps To Implement Image Thresholding, and Tutorial Steps To Implement Adaptive Image Thresholding. You will also learn: Tutorial Steps To Generate And Display Noisy Image, Tutorial Steps To Implement Edge Detection On Image, Tutorial Steps To Implement Image Segmentation Using Multiple Thresholding and K-Means Algorithm, Tutorial Steps To Implement Image Denoising, Tutorial Steps To Detect Face, Eye, and Mouth Using Haar Cascades, Tutorial Steps To Detect Face Using Haar Cascades with PyQt, Tutorial Steps To Detect Eye, and Mouth Using Haar Cascades with PyQt, Tutorial Steps To Extract Detected Objects, Tutorial Steps To Detect Image Features Using Harris Corner Detection, Tutorial Steps To Detect Image Features Using Shi-Tomasi Corner Detection, Tutorial Steps To Detect Features Using Scale-Invariant Feature Transform (SIFT), and Tutorial Steps To Detect Features Using Features from Accelerated Segment Test (FAST). In Chapter 4, In this tutorial, you will learn how to use Pandas, NumPy and other libraries to perform simple classification using perceptron and Adaline (adaptive linear neuron). The dataset used is Iris dataset directly from the UCI Machine Learning Repository. You will learn: Tutorial Steps To Implement Perceptron, Tutorial Steps To Implement Perceptron with PyQt, Tutorial

Steps To Implement Adaline (ADAptive LInear NEuron), and Tutorial Steps To Implement Adaline with PyQt. In Chapter 5, you will learn how to use the scikit-learn machine learning library, which provides a wide variety of machine learning algorithms via a user-friendly Python API and to perform classification using perceptron, Adaline (adaptive linear neuron), and other models. The dataset used is Iris dataset directly from the UCI Machine Learning Repository. You will learn: Tutorial Steps To Implement Perceptron Using Scikit-Learn, Tutorial Steps To Implement Perceptron Using Scikit-Learn with PyQt, Tutorial Steps To Implement Logistic Regression Model, Tutorial Steps To Implement Logistic Regression Model with PyQt, Tutorial Steps To Implement Logistic Regression Model Using Scikit-Learn with PyQt, Tutorial Steps To Implement Support Vector Machine (SVM) Using Scikit-Learn, Tutorial Steps To Implement Decision Tree (DT) Using Scikit-Learn, Tutorial Steps To Implement Random Forest (RF) Using Scikit-Learn, and Tutorial Steps To Implement K-Nearest Neighbor (KNN) Using Scikit-Learn. In Chapter 6, you will learn how to use Pandas, NumPy, Scikit-Learn, and other libraries to implement different approaches for reducing the dimensionality of a dataset using different feature selection techniques. You will learn about three fundamental techniques that will help us to summarize the information content of a dataset by transforming it onto a new feature subspace of lower dimensionality than the original one. Data compression is an important topic in machine learning, and it helps us to store and analyze the increasing amounts of data that are

produced and collected in the modern age of technology. You will learn the following topics: Principal Component Analysis (PCA) for unsupervised data compression, Linear Discriminant Analysis (LDA) as a supervised dimensionality reduction technique for maximizing class separability, Nonlinear dimensionality reduction via Kernel Principal Component Analysis (KPCA). You will learn: Tutorial Steps To Implement Principal Component Analysis (PCA), Tutorial Steps To Implement Principal Component Analysis (PCA) Using Scikit-Learn, Tutorial Steps To Implement Principal Component Analysis (PCA) Using Scikit-Learn with PyQt, Tutorial Steps To Implement Linear Discriminant Analysis (LDA), Tutorial Steps To Implement Linear Discriminant Analysis (LDA) with Scikit-Learn, Tutorial Steps To Implement Linear Discriminant Analysis (LDA) Using Scikit-Learn with PyQt, Tutorial Steps To Implement Kernel Principal Component Analysis (KPCA) Using Scikit-Learn, and Tutorial Steps To Implement Kernel Principal Component Analysis (KPCA) Using Scikit-Learn with PyQt. In Chapter 7, you will learn how to use Keras, Scikit-Learn, Pandas, NumPy and other libraries to perform prediction on handwritten digits using MNIST dataset. You will learn: Tutorial Steps To Load MNIST Dataset, Tutorial Steps To Load MNIST Dataset with PyQt, Tutorial Steps To Implement Perceptron With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Perceptron With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Perceptron With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement

Logistic Regression (LR) Model With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Logistic Regression (LR) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Logistic Regression (LR) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement , Tutorial Steps To Implement Support Vector Machine (SVM) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Support Vector Machine (SVM) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Decision Tree (DT) Model With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Decision Tree (DT) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Decision Tree (DT) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Random Forest (RF) Model With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Random Forest (RF) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement Random Forest (RF) Model With KPCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement K-Nearest Neighbor (KNN) Model With PCA Feature Extractor on MNIST Dataset Using PyQt, Tutorial Steps To Implement K-Nearest Neighbor (KNN) Model With LDA Feature Extractor on MNIST Dataset Using PyQt, and Tutorial Steps To Implement K-Nearest Neighbor (KNN) Model With KPCA Feature Extractor on MNIST

Dataset Using PyQt. BOOK 2: THE PRACTICAL
GUIDES ON DEEP LEARNING USING SCIKIT-LEARN,
KERAS, AND TENSORFLOW WITH PYTHON GUI In
this book, you will learn how to use TensorFlow, Keras,
Scikit-Learn, OpenCV, Pandas, NumPy and other
libraries to implement deep learning on recognizing
traffic signs using GTSRB dataset, detecting brain tumor
using Brain Image MRI dataset, classifying gender, and
recognizing facial expression using FER2013 dataset In
Chapter 1, you will learn to create GUI applications to
display line graph using PyQt. You will also learn how to
display image and its histogram. In Chapter 2, you will
learn how to use TensorFlow, Keras, Scikit-Learn,
Pandas, NumPy and other libraries to perform prediction
on handwritten digits using MNIST dataset with PyQt.
You will build a GUI application for this purpose. In
Chapter 3, you will learn how to perform recognizing
traffic signs using GTSRB dataset from Kaggle. There
are several different types of traffic signs like speed
limits, no entry, traffic signals, turn left or right, children
crossing, no passing of heavy vehicles, etc. Traffic signs
classification is the process of identifying which class a
traffic sign belongs to. In this Python project, you will
build a deep neural network model that can classify
traffic signs in image into different categories. With this
model, you will be able to read and understand traffic
signs which are a very important task for all autonomous
vehicles. You will build a GUI application for this
purpose. In Chapter 4, you will learn how to perform
detecting brain tumor using Brain Image MRI dataset
provided by Kaggle (https://www.kaggle.com/navoneel/br

ain-mri-images-for-brain-tumor-detection) using CNN model. You will build a GUI application for this purpose. In Chapter 5, you will learn how to perform classifying gender using dataset provided by Kaggle (https://www.ka ggle.com/cashutosh/gender-classification-dataset) using MobileNetV2 and CNN models. You will build a GUI application for this purpose. In Chapter 6, you will learn how to perform recognizing facial expression using FER2013 dataset provided by Kaggle (https://www.kaggl e.com/nicolejyt/facialexpressionrecognition) using CNN model. You will also build a GUI application for this purpose. BOOK 3: STEP BY STEP TUTORIALS ON DEEP LEARNING USING SCIKIT-LEARN, KERAS, AND TENSORFLOW WITH PYTHON GUI In this book, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to implement deep learning on classifying fruits, classifying cats/dogs, detecting furnitures, and classifying fashion. In Chapter 1, you will learn to create GUI applications to display line graph using PyQt. You will also learn how to display image and its histogram. Then, you will learn how to use OpenCV, NumPy, and other libraries to perform feature extraction with Python GUI (PyQt). The feature detection techniques used in this chapter are Harris Corner Detection, Shi-Tomasi Corner Detector, and Scale-Invariant Feature Transform (SIFT). In Chapter 2, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform classifying fruits using Fruits 360 dataset provided by Kaggle (https://www.kaggle.com/moltean/fruits/code) using

Transfer Learning and CNN models. You will build a GUI application for this purpose. In Chapter 3, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform classifying cats/dogs using dataset provided by Kaggle (https://www.kaggle.com/chetankv/dogs-cats-images) using Using CNN with Data Generator. You will build a GUI application for this purpose. In Chapter 4, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform detecting furnitures using Furniture Detector dataset provided by Kaggle (https://www.kaggle.com/akkithetechie/furniture-detector) using VGG16 model. You will build a GUI application for this purpose. In Chapter 5, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform classifying fashion using Fashion MNIST dataset provided by Kaggle (https://www.kaggle.com/zalando-research/fashionmnist/code) using CNN model. You will build a GUI application for this purpose. BOOK 4: Project-Based Approach On DEEP LEARNING Using Scikit-Learn, Keras, And TensorFlow with Python GUI In this book, implement deep learning on detecting vehicle license plates, recognizing sign language, and detecting surface crack using TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries. In Chapter 1, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform detecting vehicle license plates using Car License Plate Detection dataset provided by Kaggle (https://www.kaggl

e.com/andrewmvd/car-plate-detection/download). In Chapter 2, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform sign language recognition using Sign Language Digits Dataset provided by Kaggle (https://www.kaggle.com/ardamavi/sign-language-digits-dataset/download). In Chapter 3, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform detecting surface crack using Surface Crack Detection provided by Kaggle (https://www.kaggle.com/arunrk7/surface-crack-detection/download). BOOK 5: Hands-On Guide To IMAGE CLASSIFICATION Using Scikit-Learn, Keras, And TensorFlow with PYTHON GUI In this book, implement deep learning-based image classification on detecting face mask, classifying weather, and recognizing flower using TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries. In Chapter 1, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform detecting face mask using Face Mask Detection Dataset provided by Kaggle (https://www.kaggle.com/omkargurav/face-mask-dataset/download). In Chapter 2, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform how to classify weather using Multiclass Weather Dataset provided by Kaggle (https://www.kaggle.com/pratik2901/multiclass-weather-dataset/download). In Chapter 3, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform how to recognize

flower using Flowers Recognition dataset provided by Kaggle (https://www.kaggle.com/alxmamaev/flowers-recognition/download). BOOK 6: Step by Step Tutorial IMAGE CLASSIFICATION Using Scikit-Learn, Keras, And TensorFlow with PYTHON GUI In this book, implement deep learning-based image classification on classifying monkey species, recognizing rock, paper, and scissor, and classify airplane, car, and ship using TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries. In Chapter 1, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform how to classify monkey species using 10 Monkey Species dataset provided by Kaggle (https://www.kaggle.com/slothkong/10-monkey-species/download). In Chapter 2, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform how to recognize rock, paper, and scissor using 10 Monkey Species dataset provided by Kaggle (https://www.kaggle.com/sanikamal/rock-paper-scissors-dataset/download). In Chapter 3, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform how to classify airplane, car, and ship using Multiclass-image-dataset-airplane-car-ship dataset provided by Kaggle (https://www.kaggle.com/abtabm/multiclassimagedatasetairplanecar).

MATLAB is one of the most widely used tools in the field of engineering today. Its broad appeal lies in its interactive environment with hundreds of built-in functions. This book is designed to get you up and

running in just a few hours.

Go from total MATLAB newbie to plotting graphs and solving equations in a flash! MATLAB is one of the most powerful and commonly used tools in the STEM field. But did you know it doesn't take an advanced degree or a ton of computer experience to learn it? MATLAB For Dummies is the roadmap you've been looking for to simplify and explain this feature-filled tool. This handy reference walks you through every step of the way as you learn the MATLAB language and environment inside-and-out. Starting with straightforward basics before moving on to more advanced material like Live Functions and Live Scripts, this easy-to-read guide shows you how to make your way around MATLAB with screenshots and newly updated procedures. It includes: A comprehensive introduction to installing MATLAB, using its interface, and creating and saving your first file Fully updated to include the 2020 and 2021 updates to MATLAB, with all-new screenshots and up-to-date procedures Enhanced debugging procedures and use of the Symbolic Math Toolbox Brand new instruction on working with Live Scripts and Live Functions, designing classes, creating apps, and building projects Intuitive walkthroughs for MATLAB's advanced features, including importing and exporting data and publishing your work Perfect for STEM students and new professionals ready to master one of the most powerful tools in the fields of engineering, mathematics, and computing, MATLAB For Dummies is the simplest way to go from complete newbie to power user faster than you would have thought possible.

In this book, implement deep learning-based image classification on classifying monkey species, recognizing rock, paper, and scissor, and classify airplane, car, and ship using TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries. In Chapter 1, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform how to classify monkey species using 10 Monkey Species dataset provided by Kaggle (https://www.kaggle. com/slothkong/10-monkey-species/download). In Chapter 2, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform how to recognize rock, paper, and scissor using 10 Monkey Species dataset provided by Kaggle (https://www.kaggle.com/sanikamal/rock-paper-scissors-dataset/download). In Chapter 3, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform how to classify airplane, car, and ship using Multiclass-image-dataset-airplane-car-ship dataset provided by Kaggle (https://www.kaggle.com/abtabm/multiclassimagedatasetairplanecar).

Practical Guide for Biomedical Signals Analysis Using Machine Learning Techniques: A MATLAB Based Approach presents how machine learning and biomedical signal processing methods can be used in biomedical signal analysis. Different machine learning applications in biomedical signal analysis, including those for electrocardiogram, electroencephalogram and electromyogram are described in a practical and comprehensive way, helping readers with limited

knowledge. Sections cover biomedical signals and
machine learning techniques, biomedical signals, such
as electroencephalogram (EEG), electromyogram (EMG)
and electrocardiogram (ECG), different signal-processing
techniques, signal de-noising, feature extraction and
dimension reduction techniques, such as PCA, ICA,
KPCA, MSPCA, entropy measures, and other statistical
measures, and more. This book is a valuable source for
bioinformaticians, medical doctors and other members of
the biomedical field who need a cogent resource on the
most recent and promising machine learning techniques
for biomedical signals analysis. Provides comprehensive
knowledge in the application of machine learning tools in
biomedical signal analysis for medical diagnostics, brain
computer interface and man/machine interaction
Explains how to apply machine learning techniques to
EEG, ECG and EMG signals Gives basic knowledge on
predictive modeling in biomedical time series and
advanced knowledge in machine learning for biomedical
time series
This hands-on book introduces the essential topic of
coding and the Python computer language to beginners
and pogrammers of all ages. This book explains
relational theory in practice, and demonstrates through
two projects how you can apply it to your use of MySQL
and SQL Server databases. This book covers the
important requirements of teaching databases with a
practical and progressive perspective. This book offers
the straightforward, practical answers you need to help
you do your job. This hands-on tutorial/reference/guide
to MySQL and SQL Server is not only perfect for

students and beginners, but it also works for experienced developers who aren't getting the most from both databases. In designing a GUI and as an IDE, you will make use Qt Designer. In the first chapter, you will learn to use several widgets in PyQt5: Display a welcome message; Use the Radio Button widget; Grouping radio buttons; Displays options in the form of a check box; and Display two groups of check boxes. In chapter two, you will learn to use the following topics: Using Signal / Slot Editor; Copy and place text from one Line Edit widget to another; Convert data types and make a simple calculator; Use the Spin Box widget; Use scrollbars and sliders; Using the Widget List; Select a number of list items from one Widget List and display them on another Widget List widget; Add items to the Widget List; Perform operations on the Widget List; Use the Combo Box widget; Displays data selected by the user from the Calendar Widget; Creating a hotel reservation application; and Display tabular data using Table Widgets. In chapter three, you will learn: How to create the initial three tables project in the School database: Teacher, Class, and Subject tables; How to create database configuration files; How to create a Python GUI for inserting and editing tables; How to create a Python GUI to join and query the three tables. In chapter four, you will learn how to: Create a main form to connect all forms; Create a project will add three more tables to the school database: Student, Parent, and Tuition tables; Create a Python GUI for inserting and editing tables; Create a Python GUI to join and query over the three tables. In chapter five, you will join the six classes,

Teacher, TClass, Subject, Student, Parent, and Tuition
and make queries over those tables. In chapter six, you
will create dan configure database. In this chapter, you
will create Suspect table in crime database. This table
has eleven columns: suspect_id (primary key),
suspect_name, birth_date, case_date, report_date,
suspect_ status, arrest_date, mother_name, address,
telephone, and photo. You will also create GUI to
display, edit, insert, and delete for this table. In chapter
seven, you will create a table with the name
Feature_Extraction, which has eight columns: feature_id
(primary key), suspect_id (foreign key), feature1,
feature2, feature3, feature4, feature5, and feature6. The
six fields (except keys) will have VARBINARY(MAX) data
type. You will also create GUI to display, edit, insert, and
delete for this table. In chapter eight, you will create two
tables, Police and Investigator. The Police table has six
columns: police_id (primary key), province, city, address,
telephone, and photo. The Investigator table has eight
columns: investigator_id (primary key),
investigator_name, rank, birth_date, gender, address,
telephone, and photo. You will also create GUI to
display, edit, insert, and delete for both tables. In the last
chapter, you will create two tables, Victim and Case_File.
The Victim table has nine columns: victim_id (primary
key), victim_name, crime_type, birth_date, crime_date,
gender, address, telephone, and photo. The Case_File
table has seven columns: case_file_id (primary key),
suspect_id (foreign key), police_id (foreign key),
investigator_id (foreign key), victim_id (foreign key),
status, and description. You will create GUI to display,

edit, insert, and delete for both tables.
In this book, you will implement two data science
projects using Scikit-Learn, Scipy, and other libraries
with Python GUI. In Chapter 1, you will learn how to use
Scikit-Learn, Scipy, and other libraries to perform how to
predict traffic (number of vehicles) in four different
junctions using Traffic Prediction Dataset provided by
Kaggle (https://www.kaggle.com/fedesoriano/traffic-
prediction-dataset/download). This dataset contains
48.1k (48120) observations of the number of vehicles
each hour in four different junctions: 1) DateTime; 2)
Juction; 3) Vehicles; and 4) ID. In Chapter 2, you will
learn how to use Scikit-Learn, NumPy, Pandas, and
other libraries to perform how to analyze and predict
heart attack using Heart Attack Analysis & Prediction
Dataset provided by Kaggle (https://www.kaggle.com/ras
hikrahmanpritom/heart-attack-analysis-prediction-
dataset/download). In Chapter 3, you will learn how to
use Scikit-Learn, SVM, NumPy, Pandas, and other
libraries to perform how to predict early stage diabetes
using Early Stage Diabetes Risk Prediction Dataset
provided by Kaggle (https://www.kaggle.com/ishandutta/
early-stage-diabetes-risk-prediction-dataset/download).
This dataset contains the sign and symptpom data of
newly diabetic or would be diabetic patient. This has
been collected using direct questionnaires from the
patients of Sylhet Diabetes Hospital in Sylhet,
Bangladesh and approved by a doctor.
In this book, you will learn how to use OpenCV, NumPy
library and other libraries to perform signal processing,
image processing, object detection, and feature

extraction with Python GUI (PyQt). You will learn how to filter signals, detect edges and segments, and denoise images with PyQt. You will also learn how to detect objects (face, eye, and mouth) using Haar Cascades and how to detect features on images using Harris Corner Detection, Shi-Tomasi Corner Detector, Scale-Invariant Feature Transform (SIFT), and Features from Accelerated Segment Test (FAST). In Chapter 1, you will learn: Tutorial Steps To Create A Simple GUI Application, Tutorial Steps to Use Radio Button, Tutorial Steps to Group Radio Buttons, Tutorial Steps to Use CheckBox Widget, Tutorial Steps to Use Two CheckBox Groups, Tutorial Steps to Understand Signals and Slots, Tutorial Steps to Convert Data Types, Tutorial Steps to Use Spin Box Widget, Tutorial Steps to Use ScrollBar and Slider, Tutorial Steps to Use List Widget, Tutorial Steps to Select Multiple List Items in One List Widget and Display It in Another List Widget, Tutorial Steps to Insert Item into List Widget, Tutorial Steps to Use Operations on Widget List, Tutorial Steps to Use Combo Box, Tutorial Steps to Use Calendar Widget and Date Edit, and Tutorial Steps to Use Table Widget. In Chapter 2, you will learn: Tutorial Steps To Create A Simple Line Graph, Tutorial Steps To Create A Simple Line Graph in Python GUI, Tutorial Steps To Create A Simple Line Graph in Python GUI: Part 2, Tutorial Steps To Create Two or More Graphs in the Same Axis, Tutorial Steps To Create Two Axes in One Canvas, Tutorial Steps To Use Two Widgets, Tutorial Steps To Use Two Widgets, Each of Which Has Two Axes, Tutorial Steps To Use Axes With Certain Opacity Levels, Tutorial Steps To Choose

Line Color From Combo Box, Tutorial Steps To Calculate Fast Fourier Transform, Tutorial Steps To Create GUI For FFT, Tutorial Steps To Create GUI For FFT With Some Other Input Signals, Tutorial Steps To Create GUI For Noisy Signal, Tutorial Steps To Create GUI For Noisy Signal Filtering, and Tutorial Steps To Create GUI For Wav Signal Filtering. In Chapter 3, you will learn: Tutorial Steps To Convert RGB Image Into Grayscale, Tutorial Steps To Convert RGB Image Into YUV Image, Tutorial Steps To Convert RGB Image Into HSV Image, Tutorial Steps To Filter Image, Tutorial Steps To Display Image Histogram, Tutorial Steps To Display Filtered Image Histogram, Tutorial Steps To Filter Image With CheckBoxes, Tutorial Steps To Implement Image Thresholding, and Tutorial Steps To Implement Adaptive Image Thresholding. In Chapter 4, you will learn: Tutorial Steps To Generate And Display Noisy Image, Tutorial Steps To Implement Edge Detection On Image, Tutorial Steps To Implement Image Segmentation Using Multiple Thresholding and K-Means Algorithm, and Tutorial Steps To Implement Image Denoising. In Chapter 5, you will learn: Tutorial Steps To Detect Face, Eye, and Mouth Using Haar Cascades, Tutorial Steps To Detect Face Using Haar Cascades with PyQt, Tutorial Steps To Detect Eye, and Mouth Using Haar Cascades with PyQt, and Tutorial Steps To Extract Detected Objects. In Chapter 6, you will learn: Tutorial Steps To Detect Image Features Using Harris Corner Detection, Tutorial Steps To Detect Image Features Using Shi-Tomasi Corner Detection, Tutorial Steps To Detect Features Using Scale-Invariant Feature Transform (SIFT), and Tutorial

Steps To Detect Features Using Features from Accelerated Segment Test (FAST).

This step-by-step guide to explore database programming using Java is ideal for people with little or no programming experience. The goal of this concise book is not just to teach you Java, but to help you think like a programmer. Each brief chapter covers the material for one week of a college course to help you practice what you've learned. As you would expect, this book shows how to build from scratch two different databases: PostgreSQL and SQLite using Java. In designing a GUI and as an IDE, you will make use of the NetBeans tool. In the first chapter, you will learn: How to install NetBeans, JDK 11, and the PostgreSQL connector; How to integrate external libraries into projects; How the basic PostgreSQL commands are used; How to query statements to create databases, create tables, fill tables, and manipulate table contents is done.In the first chapter, you will learn: How to install NetBeans, JDK 11, and the PostgreSQL connector; How to integrate external libraries into projects; How the basic PostgreSQL commands are used; How to query statements to create databases, create tables, fill tables, and manipulate table contents is done. In the second chapter, you will learn querying data from the postgresql using jdbc including establishing a database connection, creating a statement object, executing the query, processing the resultset object, querying data using a statement that returns multiple rows, querying data using a statement that has parameters, inserting data into a table using jdbc, updating data in postgresql database

using jdbc, calling postgresql stored function using jdbc, deleting data from a postgresql table using jdbc, and postgresql jdbc transaction. In chapter three, you will create a PostgreSQL database, named School, and its tables. In chapter four, you will study: Creating the initial three table projects in the school database: Teacher table, TClass table, and Subject table; Creating database configuration files; Creating a Java GUI for viewing and navigating the contents of each table; Creating a Java GUI for inserting and editing tables; and Creating a Java GUI to join and query the three tables. In chapter five, you will learn: Creating the main form to connect all forms; Creating a project will add three more tables to the school database: the Student table, the Parent table, and Tuition table; Creating a Java GUI to view and navigate the contents of each table; Creating a Java GUI for editing, inserting, and deleting records in each table; Creating a Java GUI to join and query the three tables and all six. In chapter six, you will study how to query the six tables. In chapter seven, you will be shown how to create SQLite database and tables with Java. In chapter eight, you will be taught how to extract image features, utilizing BufferedImage class, in Java GUI. Digital image techniques to extract image features used in this chapted are grascaling, sharpening, invertering, blurring, dilation, erosion, closing, opening, vertical prewitt, horizontal prewitt, Laplacian, horizontal sobel, and vertical sobel. For readers, you can develop it to store other advanced image features based on descriptors such as SIFT and others for developing descriptor based matching. In chapter nine, you will be taught to create Java GUI to

view, edit, insert, and delete Suspect table data. This
table has eleven columns: suspect_id (primary key),
suspect_name, birth_date, case_date, report_date,
suspect_ status, arrest_date, mother_name, address,
telephone, and photo. In chapter ten, you will be taught
to create Java GUI to view, edit, insert, and delete
Feature_Extraction table data. This table has eight
columns: feature_id (primary key), suspect_id (foreign
key), feature1, feature2, feature3, feature4, feature5, and
feature6. All six fields (except keys) will have a BLOB
data type, so that the image of the feature will be directly
saved into this table. In chapter eleven, you will add two
tables: Police_Station and Investigator. These two tables
will later be joined to Suspect table through another
table, File_Case, which will be built in the seventh
chapter. The Police_Station has six columns:
police_station_id (primary key), location, city, province,
telephone, and photo. The Investigator has eight
columns: investigator_id (primary key),
investigator_name, rank, birth_date, gender, address,
telephone, and photo. Here, you will design a Java GUI
to display, edit, fill, and delete data in both tables. In
chapter twelve, you will add two tables: Victim and
Case_File. The File_Case table will connect four other
tables: Suspect, Police_Station, Investigator and Victim.
The Victim table has nine columns: victim_id (primary
key), victim_name, crime_type, birth_date, crime_date,
gender, address, telephone, and photo. The Case_File
has seven columns: case_file_id (primary key),
suspect_id (foreign key), police_station_id (foreign key),
investigator_id (foreign key), victim_id (foreign key),

status, and description. Here, you will also design a Java GUI to display, edit, fill, and delete data in both tables. Finally, this book is hopefully useful and can improve database programming skills for every Java/PostgreSL/SQLite pogrammer.

This book is designed to introduce programmers to programming and computational thinking through the lens of exploring database. This book offers Python programmers one place to look when they need help guiding to Python as one of the fastest-growing computer languages including Web and Internet applications. This clear and concise introduction to the Python language is aimed at readers who are already familiar with programming in at least one language. This hands-on book introduces the essential topic of coding and the Python computer language to beginners and pogrammers of all ages. This book explains relational theory in practice, and demonstrates through two projects how you can apply it to your use of MariaDB and SQL Server databases. This book covers the important requirements of teaching databases with a practical and progressive perspective. This book offers the straightforward, practical answers you need to help you do your job. This hands-on tutorial/reference/guide to MariaDB and SQL Server is not only perfect for students and beginners, but it also works for experienced developers who aren't getting the most from both databases. In designing a GUI and as an IDE, you will make use Qt Designer. In the first chapter, you will learn to use several widgets in PyQt5: Display a welcome message; Use the Radio Button widget; Grouping radio

buttons; Displays options in the form of a check box; and Display two groups of check boxes. In chapter two, you will learn to use the following topics: Using Signal / Slot Editor; Copy and place text from one Line Edit widget to another; Convert data types and make a simple calculator; Use the Spin Box widget; Use scrollbars and sliders; Using the Widget List; Select a number of list items from one Widget List and display them on another Widget List widget; Add items to the Widget List; Perform operations on the Widget List; Use the Combo Box widget; Displays data selected by the user from the Calendar Widget; Creating a hotel reservation application; and Display tabular data using Table Widgets. In chapter three, you will learn: How to create the initial three tables project in the School database: Teacher, Class, and Subject tables; How to create database configuration files; How to create a Python GUI for inserting and editing tables; How to create a Python GUI to join and query the three tables. In chapter four, you will learn how to: Create a main form to connect all forms; Create a project will add three more tables to the school database: Student, Parent, and Tuition tables; Create a Python GUI for inserting and editing tables; Create a Python GUI to join and query over the three tables. In chapter five, you will join the six classes, Teacher, TClass, Subject, Student, Parent, and Tuition and make queries over those tables. In chapter six, you will create dan configure database. In this chapter, you will create Suspect table in crime database. This table has eleven columns: suspect_id (primary key), suspect_name, birth_date, case_date, report_date,

suspect_ status, arrest_date, mother_name, address, telephone, and photo. You will also create GUI to display, edit, insert, and delete for this table. In chapter seven, you will create a table with the name Feature_Extraction, which has eight columns: feature_id (primary key), suspect_id (foreign key), feature1, feature2, feature3, feature4, feature5, and feature6. The six fields (except keys) will have a VARCHAR data type (200). You will also create GUI to display, edit, insert, and delete for this table. In chapter eight, you will create two tables, Police and Investigator. The Police table has six columns: police_id (primary key), province, city, address, telephone, and photo. The Investigator table has eight columns: investigator_id (primary key), investigator_name, rank, birth_date, gender, address, telephone, and photo. You will also create GUI to display, edit, insert, and delete for both tables. In chapter nine, you will create two tables, Victim and Case_File. The Victim table has nine columns: victim_id (primary key), victim_name, crime_type, birth_date, crime_date, gender, address, telephone, and photo. The Case_File table has seven columns: case_file_id (primary key), suspect_id (foreign key), police_id (foreign key), investigator_id (foreign key), victim_id (foreign key), status, and description. You will create GUI to display, edit, insert, and delete for both tables as well.

A handbook for MATLAB which gives a focused approach to the software for students and professional researchers.

This hands-on tutorial/reference/guide to MySQL and SQL Server is not only perfect for students and

beginners, but it also works for experienced developers who aren't getting the most from MySQL and SQL Server. As you would expect, this book shows how to build from scratch two different databases: MySQL and SQL Server using Java. In designing a GUI and as an IDE, you will make use of the NetBeans tool. In the first chapter, you will learn: How to install NetBeans, JDK 11, and MySQL Connector/J; How to integrate external libraries into projects; How the basic MySQL commands are used; How to query statements to create databases, create tables, fill tables, and manipulate table contents is done. In the second chapter, you will study: Creating the initial three table projects in the school database: Teacher table, TClass table, and Subject table; Creating database configuration files; Creating a Java GUI for viewing and navigating the contents of each table; Creating a Java GUI for inserting and editing tables; and Creating a Java GUI to join and query the three tables. In the third chapter, you will learn: Creating the main form to connect all forms; Creating a project will add three more tables to the school database: the Student table, the Parent table, and Tuition table; Creating a Java GUI to view and navigate the contents of each table; Creating a Java GUI for editing, inserting, and deleting records in each table; Creating a Java GUI to join and query the three tables and all six. In chapter four, you will study how to query the six

tables. In chapter five, you will be taught how to create Crime database and its tables. In chapter six, you will be taught how to extract image features, utilizing BufferedImage class, in Java GUI. In chapter seven, you will be taught to create Java GUI to view, edit, insert, and delete Suspect table data. This table has eleven columns: suspect_id (primary key), suspect_name, birth_date, case_date, report_date, suspect_ status, arrest_date, mother_name, address, telephone, and photo. In chapter eight, you will be taught to create Java GUI to view, edit, insert, and delete Feature_Extraction table data. This table has eight columns: feature_id (primary key), suspect_id (foreign key), feature1, feature2, feature3, feature4, feature5, and feature6. In chapter nine, you will add two tables: Police_Station and Investigator. These two tables will later be joined to Suspect table through another table, File_Case, which will be built in the seventh chapter. The Police_Station has six columns: police_station_id (primary key), location, city, province, telephone, and photo. The Investigator has eight columns: investigator_id (primary key), investigator_name, rank, birth_date, gender, address, telephone, and photo. Here, you will design a Java GUI to display, edit, fill, and delete data in both tables. In chapter ten, you will add two tables: Victim and File_Case. The File_Case table will connect four other tables: Suspect, Police_Station,

Investigator and Victim. The Victim table has nine
columns: victim_id (primary key), victim_name,
crime_type, birth_date, crime_date, gender, address,
telephone, and photo. The File_Case has seven
columns: file_case_id (primary key), suspect_id
(foreign key), police_station_id (foreign key),
investigator_id (foreign key), victim_id (foreign key),
status, and description. Here, you will also design a
Java GUI to display, edit, fill, and delete data in both
tables. Finally, this book is hopefully useful and can
improve database programming skills for every
Java/MySQL/SQL SERVER programmer.
Highlighting the new aspects of MATLAB® 7.10 and
expanding on many existing features, MATLAB®
Primer, Eighth Edition shows you how to solve
problems in science, engineering, and mathematics.
Now in its eighth edition, this popular primer
continues to offer a hands-on, step-by-step
introduction to using the powerful tools of MATLAB.
New to the Eighth Edition A new chapter on object-
oriented programming Discussion of the MATLAB
File Exchange window, which provides direct access
to over 10,000 submissions by MATLAB users Major
changes to the MATLAB Editor, such as code folding
and the integration of the Code Analyzer (M-Lint)
into the Editor Explanation of more powerful Help
tools, such as quick help popups for functions via the
Function Browser The new bsxfun function A
synopsis of each of the MATLAB Top 500 most

frequently used functions, operators, and special
characters The addition of several useful features,
including sets, logical indexing, isequal, repmat,
reshape, varargin, and varargout The book takes
you through a series of simple examples that
become progressively more complex. Starting with
the core components of the MATLAB desktop, it
demonstrates how to handle basic matrix operations
and expressions in MATLAB. The text then
introduces commonly used functions and explains
how to write your own functions, before covering
advanced features, such as object-oriented
programming, calling other languages from
MATLAB, and MATLAB graphics. It also presents an
in-depth look at the Symbolic Toolbox, which solves
problems analytically rather than numerically.
A practical guide to problem solving using MATLAB.
Designed to complement a taught course introducing
MATLAB but ideally suited for any beginner. This
book provides a brief tour of some of the tasks that
MATLAB is perfectly suited to instead of focusing on
any particular topic. Providing instruction, guidance
and a large supply of exercises, this book is meant
to stimulate problem-solving skills rather than
provide an in-depth knowledge of the MATLAB
language.
This book explains relational theory in practice, and
demonstrates through two projects how you can
apply it to your use of MariaDB and SQLite

databases. This book covers the important requirements of teaching databases with a practical and progressive perspective. This book offers the straightforward, practical answers you need to help you do your job. This hands-on tutorial/reference/guide to MariaDB and SQLite is not only perfect for students and beginners, but it also works for experienced developers who aren't getting the most from both databases. In designing a GUI and as an IDE, you will make use Qt Designer. In the first chapter, you will learn to use several widgets in PyQt5: Display a welcome message; Use the Radio Button widget; Grouping radio buttons; Displays options in the form of a check box; and Display two groups of check boxes. In chapter two, you will learn to use the following topics: Using Signal / Slot Editor; Copy and place text from one Line Edit widget to another; Convert data types and make a simple calculator; Use the Spin Box widget; Use scrollbars and sliders; Using the Widget List; Select a number of list items from one Widget List and display them on another Widget List widget; Add items to the Widget List; Perform operations on the Widget List; Use the Combo Box widget; Displays data selected by the user from the Calendar Widget; Creating a hotel reservation application; and Display tabular data using Table Widgets. In chapter three, you will learn: How to create the initial three tables project in the School database: Teacher, Class, and Subject

tables; How to create database configuration files; How to create a Python GUI for inserting and editing tables; How to create a Python GUI to join and query the three tables. In chapter four, you will learn how to: Create a main form to connect all forms; Create a project will add three more tables to the school database: Student, Parent, and Tuition tables; Create a Python GUI for inserting and editing tables; Create a Python GUI to join and query over the three tables. In chapter five, you will join the six classes, Teacher, TClass, Subject, Student, Parent, and Tuition and make queries over those tables. In chapter six, you will create dan configure database. In this chapter, you will create Suspect table in crime database. This table has eleven columns: suspect_id (primary key), suspect_name, birth_date, case_date, report_date, suspect_ status, arrest_date, mother_name, address, telephone, and photo. You will also create GUI to display, edit, insert, and delete for this table. In chapter seven, you will create a table with the name Feature_Extraction, which has eight columns: feature_id (primary key), suspect_id (foreign key), feature1, feature2, feature3, feature4, feature5, and feature6. The six fields (except keys) will have a VARCHAR data type (200). You will also create GUI to display, edit, insert, and delete for this table. In chapter eight, you will create two tables, Police and Investigator. The Police table has six columns: police_id (primary key), province, city,

address, telephone, and photo. The Investigator table has eight columns: investigator_id (primary key), investigator_name, rank, birth_date, gender, address, telephone, and photo. You will also create GUI to display, edit, insert, and delete for both tables. In chapter nine, you will create two tables, Victim and Case_File. The Victim table has nine columns: victim_id (primary key), victim_name, crime_type, birth_date, crime_date, gender, address, telephone, and photo. The Case_File table has seven columns: case_file_id (primary key), suspect_id (foreign key), police_id (foreign key), investigator_id (foreign key), victim_id (foreign key), status, and description. You will create GUI to display, edit, insert, and delete for both tables as well.

This book covers microsoft acces and SQL Server based GUI programming using pyqt. Intentionally designed for various levels of interest and ability of learners, this book is suitable for students, engineers, and even researchers in a variety of disciplines. No advanced programming experience is needed, and only a few school-level programming skill are needed. In the first chapter, you will learn to use several widgets in PyQt5: Display a welcome message; Use the Radio Button widget; Grouping radio buttons; Displays options in the form of a check box; and Display two groups of check boxes. In chapter two, you will learn to use the following

topics: Using Signal / Slot Editor; Copy and place text from one Line Edit widget to another; Convert data types and make a simple calculator; Use the Spin Box widget; Use scrollbars and sliders; Using the Widget List; Select a number of list items from one Widget List and display them on another Widget List widget; Add items to the Widget List; Perform operations on the Widget List; Use the Combo Box widget; Displays data selected by the user from the Calendar Widget; Creating a hotel reservation application; and Display tabular data using Table Widgets. In third chapter, you will learn: How to create the initial three tables project in the School database: Teacher, Class, and Subject tables; How to create database configuration files; How to create a Python GUI for inserting and editing tables; How to create a Python GUI to join and query the three tables. In fourth chapter, you will learn how to: Create a main form to connect all forms; Create a project will add three more tables to the school database: Student, Parent, and Tuition tables; Create a Python GUI for inserting and editing tables; Create a Python GUI to join and query over the three tables. In chapter five, you will join the six classes, Teacher, TClass, Subject, Student, Parent, and Tuition and make queries over those tables. In chapter six, you will create dan configure database. In this chapter, you will create Suspect table in crime database. This table has eleven columns: suspect_id

(primary key), suspect_name, birth_date, case_date, report_date, suspect_ status, arrest_date, mother_name, address, telephone, and photo. You will also create GUI to display, edit, insert, and delete for this table. In chapter seven, you will create a table with the name Feature_Extraction, which has eight columns: feature_id (primary key), suspect_id (foreign key), feature1, feature2, feature3, feature4, feature5, and feature6. The six fields (except keys) will have VARBINARY(MAX) data type. You will also create GUI to display, edit, insert, and delete for this table. In chapter eight, you will create two tables, Police and Investigator. The Police table has six columns: police_id (primary key), province, city, address, telephone, and photo. The Investigator table has eight columns: investigator_id (primary key), investigator_name, rank, birth_date, gender, address, telephone, and photo. You will also create GUI to display, edit, insert, and delete for both tables. In the last chapter, you will create two tables, Victim and Case_File. The Victim table has nine columns: victim_id (primary key), victim_name, crime_type, birth_date, crime_date, gender, address, telephone, and photo. The Case_File table has seven columns: case_file_id (primary key), suspect_id (foreign key), police_id (foreign key), investigator_id (foreign key), victim_id (foreign key), status, and description. You will create GUI to display, edit, insert, and delete for both tables as

well.

This book covers how to implement MATLAB GUI
from scratch: Discrete Signals And Systems, IIR
Filter: Direct Form I, IIR Filter: Direct Form II, IIR
Filter: Lattice Form, Odd Length Symmetric Linear-
Phase Filter, Hamming-Window-Based FIR Filter
And Its Implementation On Audio File, and Various
Windows Based FIR Filter And Its Implementation
On Audio Signal. Primarily aimed at a first course in
programming for high school and undergraduate
students, this book teaches the practical concepts of
GUI programming. The chapter sequence covers
programs that produce graphics, building up to an
emphasis on GUI tools for signal processing. Topics
include programming basics, creating GUI with
GUIDE, and graphics and GUI techniques.
MATLAB Programming for Biomedical Engineers
and Scientists provides an easy-to-learn introduction
to the fundamentals of computer programming in
MATLAB. This book explains the principles of good
programming practice, while demonstrating how to
write efficient and robust code that analyzes and
visualizes biomedical data. Aimed at the biomedical
engineer, biomedical scientist, and medical
researcher with little or no computer programming
experience, it is an excellent resource for learning
the principles and practice of computer programming
using MATLAB. This book enables the reader to:
Analyze problems and apply structured design

methods to produce elegant, efficient and well-structured program designs Implement a structured program design in MATLAB, making good use of incremental development approaches Write code that makes good use of MATLAB programming features, including control structures, functions and advanced data types Write MATLAB code to read in medical data from files and write data to files Write MATLAB code that is efficient and robust to errors in input data Write MATLAB code to analyze and visualize medical data, including imaging data For a firsthand interview with the authors, please visit http://scitechconnect.elsevier.com/matlab-programming-biomedical-engineers-scientists/ To access student materials, please visit https://www.elsevier.com/books-and-journals/book-companion/9780128122037 To register and access instructor materials, please visit http://textbooks.elsevier.com/web/Manuals.aspx?isbn=9780128122037 Many real world biomedical problems and data show the practical application of programming concepts Two whole chapters dedicated to the practicalities of designing and implementing more complex programs An accompanying website containing freely available data and source code for the practical code examples, activities, and exercises in the book For instructors, there are extra teaching materials including a complete set of slides, notes for a course based on the book, and course work suggestions

This is a short, focused introduction to MATLAB, a comprehensive software system for mathematical and technical computing. It contains concise explanations of essential MATLAB commands, as well as easily understood instructions for using MATLAB's programming features, graphical capabilities, simulation models, and rich desktop interface. Written for MATLAB 7, it can also be used with earlier (and later) versions of MATLAB. This book teaches how to graph functions, solve equations, manipulate images, and much more. It contains explicit instructions for using MATLAB's companion software, Simulink, which allows graphical models to be built for dynamical systems. MATLAB's new "publish" feature is discussed, which allows mathematical computations to be combined with text and graphics, to produce polished, integrated, interactive documents. For the beginner it explains everything needed to start using MATLAB, while experienced users making the switch to MATLAB 7 from an earlier version will also find much useful information here.

In this book, you will create two MariaDB and PostgreSQL driven projects using PyQt. The step-by-step guide in this book is expected to help the reader's confidence to become a programmer who can solve database programming problems. A progressive project is provided to demonstrate how to apply the concepts of MariaDB and PostgreSQL using Python. In second chapter, you will learn PyQt that consists of a number of Python bindings for cross-platform applications that

combine all the strengths of Qt and Python. By using PyQt, you can include all Qt libraries in Python code, so you can write GUI applications in Python. In other words, you can use PyQt to access all the features provided by Qt through Python code. Because PyQt depends on the Qt libraries at run time, you need to install PyQt. In third chapter, you will learn: How to create the initial three tables project in the School database: Teacher, Class, and Subject tables; How to create database configuration files; How to create a Python GUI for inserting and editing tables; How to create a Python GUI to join and query the three tables. In fourth chapter, you will learn how to: Create a main form to connect all forms; Create a project will add three more tables to the school database: Student, Parent, and Tuition tables; Create a Python GUI for inserting and editing tables; Create a Python GUI to join and query over the three tables. In this chapter, you will join the six classes, Teacher, TClass, Subject, Student, Parent, and Tuition and make queries over those tables. In chapter five, you will create dan configure PotgreSQL database. In this chapter, you will create Suspect table in crime database. This table has eleven columns: suspect_id (primary key), suspect_name, birth_date, case_date, report_date, suspect_ status, arrest_date, mother_name, address, telephone, and photo. You will also create GUI to display, edit, insert, and delete for this table. In chapter six, you will create a table with the name Feature_Extraction, which has eight columns: feature_id (primary key), suspect_id (foreign key), feature1, feature2, feature3, feature4, feature5, and feature6. The six fields (except keys) will have a VARCHAR data type (200). You will also create GUI to display, edit, insert, and delete for this table. In chapter seven, you will create two tables, Police and Investigator. The Police table has six columns: police_id (primary key), province, city, address, telephone, and photo.

The Investigator table has eight columns: investigator_id (primary key), investigator_name, rank, birth_date, gender, address, telephone, and photo. You will also create GUI to display, edit, insert, and delete for both tables. In chapter eight, you will create two tables, Victim and Case_File. The Victim table has nine columns: victim_id (primary key), victim_name, crime_type, birth_date, crime_date, gender, address, telephone, and photo. The Case_File table has seven columns: case_file_id (primary key), suspect_id (foreign key), police_id (foreign key), investigator_id (foreign key), victim_id (foreign key), status, and description. You will create GUI to display, edit, insert, and delete for both tables as well.

This second edition provides illustrative example sets to simplify the process of learning and mastering the powerful, flexible, and easy-to-use MATLAB graphics environment. It shows how to maximize the high performance and open-environment capabilities for generating, displaying, and analyzing numerical data as well as how to quickly create interesting and beautiful graphics. The book covers plotting, color, animation, the new z buffer algorithm, new functions for generating graphics for presentations, and GUI programming techniques. Designed as both an introduction as well as an advanced learning tool, the book uses step-by-step tutorials with a level of detail, explanation, and instruction that allows readers to discover the full potential of the MATLAB graphics programming capability.

This book explains relational theory in practice, and demonstrates through two projects how you can apply it to your use of MariaDB and SQL Server databases. This book covers the important requirements of teaching databases with a practical and progressive perspective. This book offers the straightforward, practical answers you need to help you do your job. This hands-on tutorial/reference/guide to MariaDB

and SQL Server is not only perfect for students and beginners, but it also works for experienced developers who aren't getting the most from MariaDB and SQL Server. As you would expect, this book shows how to build from scratch two different databases: MariaDB and SQL Server using Java. In designing a GUI and as an IDE, you will make use of the NetBeans tool. In chapter one, you will learn the basics of cryptography using Java. Here, you will learn how to write a Java program to count Hash, MAC (Message Authentication Code), store keys in a KeyStore, generate PrivateKey and PublicKey, encrypt / decrypt data, and generate and verify digital prints. You will also learn how to create and store salt passwords and verify them. In chapter two, you will create a PostgreSQL database, named Bank, and its tables. In chapter three, you will create a Login table. In this case, you will see how to create a Java GUI using NetBeans to implement it. In addition to the Login table, in this chapter you will also create a Client table. In the case of the Client table, you will learn how to generate and save public and private keys into a database. You will also learn how to encrypt / decrypt data and save the results into a database. In chapter four, you will create an Account table. This account table has the following ten fields: account_id (primary key), client_id (primarykey), account_number, account_date, account_type, plain_balance, cipher_balance, decipher_balance, digital_signature, and signature_verification. In this case, you will learn how to implement generating and verifying digital prints and storing the results into a database. In chapter five, you create a table named Client_Data, which has seven columns: client_data_id (primary key), account_id (primary_key), birth_date, address, mother_name, telephone, and photo_path. In chapter six, you will be taught how to create a SQL Server database, named Crime, and its tables. In chapter seven, you will be taught how to extract image

features, utilizing BufferedImage class, in Java GUI. In chapter eight, you will be taught to create Java GUI to view, edit, insert, and delete Suspect table data. This table has eleven columns: suspect_id (primary key), suspect_name, birth_date, case_date, report_date, suspect_ status, arrest_date, mother_name, address, telephone, and photo. In chapter nine, you will be taught to create Java GUI to view, edit, insert, and delete Feature_Extraction table data. This table has eight columns: feature_id (primary key), suspect_id (foreign key), feature1, feature2, feature3, feature4, feature5, and feature6. In chapter ten, you will add two tables: Police_Station and Investigator. These two tables will later be joined to Suspect table through another table, File_Case, which will be built in the seventh chapter. The Police_Station has six columns: police_station_id (primary key), location, city, province, telephone, and photo. The Investigator has eight columns: investigator_id (primary key), investigator_name, rank, birth_date, gender, address, telephone, and photo. Here, you will design a Java GUI to display, edit, fill, and delete data in both tables. In chapter eleven, you will add two tables: Victim and File_Case. The File_Case table will connect four other tables: Suspect, Police_Station, Investigator and Victim. The Victim table has nine columns: victim_id (primary key), victim_name, crime_type, birth_date, crime_date, gender, address, telephone, and photo. The File_Case has seven columns: file_case_id (primary key), suspect_id (foreign key), police_station_id (foreign key), investigator_id (foreign key), victim_id (foreign key), status, and description. Here, you will also design a Java GUI to display, edit, fill, and delete data in both tables. Finally, this book is hopefully useful and can improve database programming skills for every Java/MariaDB/SQL Server programmer.
This book is SQLite-based python programming. Deliberately

designed for various levels of programming skill, this book is suitable for students, engineers, and even researchers in various disciplines. There is no need for advanced programming experience, and school-level programming skills are needed. In the first chapter, you will learn to use several widgets in PyQt5: Display a welcome message; Use the Radio Button widget; Grouping radio buttons; Displays options in the form of a check box; and Display two groups of check boxes. In chapter two, you will learn to use the following topics: Using Signal / Slot Editor; Copy and place text from one Line Edit widget to another; Convert data types and make a simple calculator; Use the Spin Box widget; Use scrollbars and sliders; Using the Widget List; Select a number of list items from one Widget List and display them on another Widget List widget; Add items to the Widget List; Perform operations on the Widget List; Use the Combo Box widget; Displays data selected by the user from the Calendar Widget; Creating a hotel reservation application; and Display tabular data using Table Widgets. In third chapter, you will learn: How to create the initial three tables project in the School database: Teacher, Class, and Subject tables; How to create database configuration files; How to create a Python GUI for inserting and editing tables; How to create a Python GUI to join and query the three tables. In fourth chapter, you will learn how to: Create a main form to connect all forms; Create a project will add three more tables to the school database: Student, Parent, and Tuition tables; Create a Python GUI for inserting and editing tables; Create a Python GUI to join and query over the three tables. In the last chapter, you will join the six classes, Teacher, TClass, Subject, Student, Parent, and Tuition and make queries over those tables. In this book, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to implement deep learning on classifying fruits, classifying

cats/dogs, detecting furnitures, and classifying fashion. In Chapter 1, you will learn to create GUI applications to display line graph using PyQt. You will also learn how to display image and its histogram. Then, you will learn how to use OpenCV, NumPy, and other libraries to perform feature extraction with Python GUI (PyQt). The feature detection techniques used in this chapter are Harris Corner Detection, Shi-Tomasi Corner Detector, and Scale-Invariant Feature Transform (SIFT). In Chapter 2, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform classifying fruits using Fruits 360 dataset provided by Kaggle (https://www.kaggle.com/moltean/fruits/code) using Transfer Learning and CNN models. You will build a GUI application for this purpose. In Chapter 3, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform classifying cats/dogs using dataset provided by Kaggle (https://www.kaggle.com/chetankv/dogs-cats-images) using Using CNN with Data Generator. You will build a GUI application for this purpose. In Chapter 4, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform detecting furnitures using Furniture Detector dataset provided by Kaggle (https://www.kaggle.com/akkithetechie/furniture-detector) using VGG16 model. You will build a GUI application for this purpose. In Chapter 5, you will learn how to use TensorFlow, Keras, Scikit-Learn, OpenCV, Pandas, NumPy and other libraries to perform classifying fashion using Fashion MNIST dataset provided by Kaggle (https://www.kaggle.com/zalando-research/fashionmnist/code) using CNN model. You will build a GUI application for this purpose.