

Python GUI with PyQt

*Learn to build modern and stunning GUIs in
Python with PyQt5 and Qt Designer*

Saurabh Chandrakar

Dr. Nilesh Bhaskarrao Bahadure



www.bpbonline.com

Copyright © 2024 BPB Online

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, without the prior written permission of the publisher, except in the case of brief quotations embedded in critical articles or reviews.

Every effort has been made in the preparation of this book to ensure the accuracy of the information presented. However, the information contained in this book is sold without warranty, either express or implied. Neither the author, nor BPB Online or its dealers and distributors, will be held liable for any damages caused or alleged to have been caused directly or indirectly by this book.

BPB Online has endeavored to provide trademark information about all of the companies and products mentioned in this book by the appropriate use of capitals. However, BPB Online cannot guarantee the accuracy of this information.

First published: 2024

Published by BPB Online

WeWork

119 Marylebone Road

London NW1 5PU

UK | UAE | INDIA | SINGAPORE

ISBN 978-93-55515-575

www.bpbonline.com

[Kup książkę](#)

Dedicated to

My Parents

***Dr. Surendra Kumar Chandrakar and
Smt. Bhuneshwari Chandrakar***

brother

Shri Pranav Chandrakar,

Sister in law

Smt. Silky Chandrakar

to my wife

Smt. Priyanka Chandrakar

and to my lovely son

Master Yathartha Chandrakar

- Saurabh Chandrakar

My Parents

Smt. Kamal B. Bahadure and Late Bhaskarrao M. Bahadure
to my in-laws

Smt. Saroj R. Lokhande and Shri Ravikant A. Lokhande
and to my wife

Shilpa N. Bahadure

and to beautiful daughters

Nishita and Mrunmayee

and to all my beloved students

- Dr. Nilesh Bhaskarrao Bahadure

About the Authors

- **Saurabh Chandrakar** is a Research & Development Engineer (Dy. Manager) at **Bharat Heavy Electricals Limited (BHEL)** Hyderabad. He is the winner of the best executive award on Operations Division by BHEL Hyderabad. Recently, he has been awarded the prestigious BHEL Excellence Award under Anusandhan category for Redundant Composite Monitoring System of Power Transformers project. He has 23 copyrights and 3 patent granted. Additionally, he has 4 patents filed. Moreover, he has published 5 books in reputed publications such as BPB New Delhi (*Programming Techniques using Python, Python for everyone. Building Modern GUIs with tkinter and python*), Scitech Publications Chennai (*Programming Techniques using matlab*) and IK International publishers (*Microcontrollers and Embedded System Design*). Additionally, he has also launched one video course on BPB titled “First Time Play with Basic, Advanced Python concepts and complete guide for different python certification exams all in one umbrella.”
- **Dr. Nilesh Bhaskarrao Bahadure** received the Bachelor of Engineering degree in Electronics Engineering in 2000, the Master of Engineering degree in Digital Electronics in 2005, and the Ph.D. degree in Electronics in 2017 from KIIT Deemed to be University, Bhubaneswar, India. He is currently an Associate Professor in the Department of Computer Science and Engineering at Symbiosis Institute of Technology (SIT), Nagpur, Symbiosis International (Deemed University) (SIU), Pune, Maharashtra, India. He has more than 20 years of experience. Dr. Bahadure is a life member of IE(I), IETE, ISTE, ISCA, SESI, ISRS, and IAENG professional organizations. He has published more than 40 articles in reputed international journals and conferences, and has five books to his credit. He is the reviewer of many indexed journals such as IEEE Access, IET, Springer, Elsevier, Wiley and so on. His research interests are in the areas of Sensor Technology, the Internet of Things, and Biomedical Image Processing.

About the Reviewer

Prasenjeet Damodar Patil is currently working as an Associate Professor at the School of Computing, M.I.T A.D.T University, Pune. He has completed his B.E. in E&TC Engg. and M. Tech. in Electronics from Walchand College of Engineering Sangli, India. He did his Ph.D. in E&TC Engineering in 2018. Prasenjeet has 14+ years of teaching experience and has published more than 16 papers in reputed Journals. His research interest includes Computational Electromagnetics applications in Integrated Optics, IoT, and digital Image Processing.

Acknowledgements

- First and foremost, I would like to thank you all for selecting this book. It has been written with the beginner reader in mind. First of all, I take this opportunity to greet and thank my mentor "Prof. Nilesh Bahadure Sir" for motivating me and always communicating his expertise fully on topics related to Python. I am very thankful for being his protégé. I appreciate his belief in me for always standing behind me and pushing me to achieve more. The phrase "Journey of a Thousand Miles Begins with a Single Step" is something he always reminds me of.

My parents, Dr. Surendra Kumar Chandrakar and Smt. Bhuneshwari Chandrakar, my brother, Shri Pranav Chandrakar, my sister-in-law Silky Chandrakar, my beloved wife, Mrs. Priyanka Chandrakar, my adorable son Yathartha Chandrakar, and all of my friends have inspired me and given me confidence over the years. Last but not the least, I would like to express my sincere gratitude to the staff at BPB Publications for their contributions and insights that made parts of this book possible.

— Saurabh Chandrakar

- It was my privilege to thank Dr. S. B. Mujumdar, Chancellor of the Symbiosis International University, Pune, and Shri. Vijay Kumar Gupta, Chairman of Beekay Industries Bhilai and BIT Trust, for his encouragement and support. I thank my mentors, Dr. Arun Kumar Ray, Dean, School of Electronics Engineering, KIIT Deemed to be University, Bhubaneswar, and Dr. Anupam Shukla, Director, SVNIT Surat. I would like to thank Dr. Vidya Yeravdekar, Principal Director of Symbiosis Society and the Pro Chancellor of Symbiosis International University, Pune, Dr. Ramakrishnan Raman, Vice Chancellor of the Symbiosis International University, Pune, and Dr. Ketan Kotecha, Dean, Faculty of Engineering, Symbiosis International University, Pune, and Dr. Nitin Rakesh, Director, SIT Nagpur, for their advice, and encouragement throughout the preparation of the book.

I would also like to thank Dr. Sanjeev Khandal, HOD, Department of Aeronautical Engineering, SGU Kolhapur, my well-wisher Dr. Prasenjeet D. Patil, Associate Professor, MIT ADT University, Pune, and my colleagues in Symbiosis Institute of Technology Nagpur for providing valuable suggestions and lots of encouragement throughout the project.

I am thankful to Prof. Dr. N. Raju, Sr. Assistant Professor, SASTRA University, Thanjavur, Tamil Nadu, for his support, assistance during writing, and for his valuable suggestions.

I would also like to thank Dr. Ravi M. Potdar, Sr. Associate Professor, BIT Durg, and Dr. Md. Khwaja Mohiddin, Associate Professor, BIT Raipur, for providing valuable suggestions and encouragement throughout the project. Writing a beautiful, well-balanced, and acquainted book is not a work of one or two days or a month; it takes a lot of time and patience, as well as hours of hard work. Many thanks to my family members, my parents, wife, children, and well-wishers for their kind support. Without them and their faith and support, writing this classic book would have remained just a dream. I also like to thank my students, who have always been with me, for relating problems and finding solutions too. The perfection in any work does not come in a day. It needs a lot of effort, time, hard work, and sometimes, proper guidance.

It is my privilege to thank Prof. (Dr.) Ram Dhekekar, Professor, Department of Electronics & Telecommunication Engineering, SSGMCE Shegaon, and Dr. C. G. Dethe, Director UGC Staff College Nagpur. Last but not the least, I would like to offer an extra special thanks to the staff at "BPB Publications Private Limited" for their insight and contribution in polishing this book.

Most significantly, I want to thank Lord Ganesha for all of the work I was able to put into the book's preparation. I would not be as zealous as I am now if it weren't for God's amazing creation of the universe.

*For since the creation of the world God's invisible qualities - his eternal power and
divine nature - have been clearly seen, being understood from what
has been made, so that men are without excuse."*

— Dr. Nilesch Bhaskarrao Bahadure

Preface

The purpose of this book is to introduce readers with little to no programming experience in Python Graphical User Interface (GUI) to use Python binding for the GUI toolkit called PyQt5. A GUI application can be created in any programming language, whether VB.Net, C#.Net etc. In this book, we will see how to create a GUI application using PyQt5. The readers will get the foundational knowledge and skills necessary to start writing code for creating a GUI app in Python language. We will use a Graphical tool for creating user interfaces in Qt called Qt Designer. By mastering PyQt5, readers can apply this knowledge to solve real-world problems and create various useful applications according to their needs.

The first part of the book covers PyQt5 library and the overall layout of Qt Designer tool. Then we will see some insights into layout management and the concept of event-driven programming and its implementation in the context of Python programming with the usage of the concept of signals and slots. Finally, in the later part of the book, we shall see an insight into various widgets related to buttons, container item views, container, input, and display widgets.

This book covers a wide range of topics, from the basic definitions of different widgets to various solved examples with well-explanatory code. Overall, the book provides a solid foundation for beginners to start their journey for getting trained in Python GUI using the PyQt5 library along with the Qt Designer layout tool.

This book is divided into **9 chapters**. Each chapter description is listed as follows.

Chapter 1: Introduction to PyQt5 and Qt Designer Tool – This chapter will start by comparing powerful and cross-platform toolkit graphical PyQt5 with the tkinter library. You will learn how to install PyQt5 framework installation along with the creation of a basic GUI form using PyQt5 without and with using the class. We shall explore the components inside Qt Designer along with different pre-defined templates. In the latter half of the chapter, we will create a user credential app initially focusing on view in Qt Designer (.ui file, which is an XML file), then converting it into a python code (.py) using pyuic5 command and finally creating a new python file which will be importing the python code for user interface design and adding some useful logic to create a basic Login application for the user.

Chapter 2: Getting Insights of Layout Management – will cover the concept on widgets placement using absolute positioning approach. We shall see widgets placement using layout classes where initially we will see how to organize widgets either horizontally or vertically

using `QBoxLayout` class. We will explore how to arrange the widgets in a row side by side using `QHBoxLayout` with the usage of methods viz `addStretch`, `addWidget`, `addLayout`. Alternatively, we will look into arranging the widgets vertically with `addStretch` method using `QVBoxLayout`. Then we shall find out arranging widgets in a grid of rows and columns using `QGridLayout`. Also, we will look into creating an application using `QFormLayout`. In the end, we will be confident to create "User Credential App" application using absolute positioning, `QBoxLayout`, `QGridLayout` and `QFormLayout` class.

Chapter 3: Getting Insights of Events, Signals and Slots – will explore the concept of event-driven programming and how it is implemented in the context of Python programming using signals and slots. The above chapter will guide us through utilizing signals and slots to create and manage events. We will discover how to define signals and connect them to slots so that our program can respond to events in a useful way. We will also investigate various events and signals and how to use them to create interactive user interfaces or handle external inputs.

Chapter 4: Getting Insights of Button Widgets in Qt Designer – will cover the concept of button widgets which are commonly used for creating interactive user interfaces and make them understand their properties, functionality, and customization options. By exploring the features and settings related to button widgets, user can effectively design and implement user-friendly interfaces in our Qt applications. Qt Designer offers a number of different button widgets, including `CheckBox`, `Push Button`, `Tool Button`, `Radio Button`, `Command Link Button` etc and shall explore description, properties, important methods, important signals and an application example with output display of each button widget in detail. The important properties of `QObject`, `QWidget` and `QAbstractButton` will be covered with images at the end as an add-on information.

Chapter 5: Getting Insights of Item Views in Qt Designer – will cover the concept of Item Views in Qt Designer which are commonly used for creating user interactive interfaces. We will look into their properties, functionality, and customization options. We will explore Qt Designer's item view widgets, like `QTableView`, `QTreeView`, and `QListView`, which are effective tools for presenting data in an organized and ordered way. User can use these widgets to display data in simple list formats (`QListView`), hierarchical tree structures (`QTreeView`), or rows and columns (`QTableView`).

Chapter 6: Getting Insights of Item Widgets (Item-Based) in Qt Designer – will cover the concept of Item Widgets in Qt Designer which are commonly used for creating interactive user interfaces. We shall look to understand their properties, functionality, customization options and also will gain a thorough grasp of how to create and operate item-based widgets within the Qt Designer environment. User will be able to create dynamic, interactive user interfaces using the power of item-based widgets and will also discover different features and characteristics to

personalize the look and behavior of widgets viz list widget, tree widget and table widget. In order to facilitate user interaction and implement functionality, user will learn to manage the events and signals connected to item-based widgets.

Chapter 7: Getting Insights of Containers in Qt Designer – will cover the concept of container widgets offered by Qt Designer, their features, and how to customize them to design aesthetically pleasing and user-friendly interfaces. User will investigate various container widget types and will obtain knowledge of the precise features and capabilities provided by each widget. They will get to know what container widgets are and how they work, its different types, its usage to create layouts and shall also explore customizing appearance of container widgets.

Chapter 8: Getting Insights of Input Widgets in Qt Designer – will deal with concept of many input widgets available and how to utilize them effectively to create interactive user interfaces. Users will have a solid understanding of input widgets including QLineEdit, QSpinBox, QComboBox, QTextEdit etc. as well as their corresponding characteristics, functionality, and customization choices. They all shall experience the knowledge needed to incorporate these input widgets into their designs so that they may enter data, select options, and interact with the program. The use of input validation methods, processing user input events, and linking signals and slots to accomplish required functionality are also covered for the benefit of the readers. In the end, they will be able to develop simple, user-friendly interfaces which will effectively collect user input and deliver a smooth user experience by mastering input widgets in Qt Designer.

Chapter 9: Getting Insights of Display Widgets in Qt Designer – will explain the concept of display widgets in Qt Designer. Initially we will learn how to display static text or images using labels and how to change their font, color, alignment, and size. And then understand how labels can improve the GUI's visual presentation of information. We will examine the TextBrowser widget's features to learn how to display and control rich text content. Also, we will learn how to add hyperlinks, graphics, and formatting choices to text displays to make them dynamic and interactive. Next, we will explore how to add a calendar widget to our GUI application. We shall discover how to customize the calendar widget's look, structure, and behavior to meet specific application requirements. We will explore how to display numerical values, such as counters using the LCDNumber widget and also how to modify the LCDNumber widget's digit count, decimal accuracy, look, and style. Finally, we will look into exploring ProgressBar widget to show how a task or operation is progressing. We will learn how to dynamically update the progress bar based on our application.

Code Bundle and Coloured Images

Please follow the link to download the
Code Bundle and the *Coloured Images* of the book:

<https://rebrand.ly/s2nzbop>

The code bundle for the book is also hosted on GitHub at:

<https://github.com/bpbpublications/Python-GUI-with-PyQt>.

In case there's an update to the code, it will be updated on the existing GitHub repository.

We have code bundles from our rich catalogue of books and videos available at **<https://github.com/bpbpublications>**. Check them out!

Errata

We take immense pride in our work at BPB Publications and follow best practices to ensure the accuracy of our content to provide with an indulging reading experience to our subscribers. Our readers are our mirrors, and we use their inputs to reflect and improve upon human errors, if any, that may have occurred during the publishing processes involved. To let us maintain the quality and help us reach out to any readers who might be having difficulties due to any unforeseen errors, please write to us at :

errata@bpbonline.com

Your support, suggestions and feedbacks are highly appreciated by the BPB Publications' Family.

Did you know that BPB offers eBook versions of every book published, with PDF and ePub files available? You can upgrade to the eBook version at www.bpbonline.com and as a print book customer, you are entitled to a discount on the eBook copy. Get in touch with us at :

business@bpbonline.com for more details.

At **www.bpbonline.com**, you can also read a collection of free technical articles, sign up for a range of free newsletters, and receive exclusive discounts and offers on BPB books and eBooks.

Piracy

If you come across any illegal copies of our works in any form on the internet, we would be grateful if you would provide us with the location address or website name. Please contact us at **business@bpbonline.com** with a link to the material.

If you are interested in becoming an author

If there is a topic that you have expertise in, and you are interested in either writing or contributing to a book, please visit **www.bpbonline.com**. We have worked with thousands of developers and tech professionals, just like you, to help them share their insights with the global tech community. You can make a general application, apply for a specific hot topic that we are recruiting an author for, or submit your own idea.

Reviews

Please leave a review. Once you have read and used this book, why not leave a review on the site that you purchased it from? Potential readers can then see and use your unbiased opinion to make purchase decisions. We at BPB can understand what you think about our products, and our authors can see your feedback on their book. Thank you!

For more information about BPB, please visit **www.bpbonline.com**.

Join our book's Discord space

Join the book's Discord Workspace for Latest updates, Offers, Tech happenings around the world, New Release and Sessions with the Authors:

<https://discord.bpbonline.com>



Table of Contents

1. Introduction to PyQt5 and Qt Designer Tool.....	1
Introduction	1
Structure	1
Objectives	2
Comparison of PyQt5 with tkinter library	2
PyQt5 framework installation.....	4
First GUI form creation using PyQt5 without using class	5
GUI form creation using PyQt5 by using class.....	8
Installation of Qt Designer with pre-defined templates.....	13
Components of Qt Designer	16
User credential app demo.....	17
Conclusion	29
Points to remember.....	29
Questions.....	30
2. Getting Insights of Layout Management.....	31
Introduction	31
Structure	32
Objectives	32
Widgets placement using absolute positioning.....	32
Widgets placement using layout classes.....	37
QBoxLayout	37
QHBoxLayout	37
QVBoxLayout	47
QGridLayout.....	54
Basic QGridLayout.....	54
QGridLayout span	57
QGridLayout Stretch	59
QFormLayout.....	63
Conclusion	77

Points to remember	78
Questions.....	78
3. Getting Insights of Events, Signals and Slots.....	79
Introduction	79
Structure	79
Objectives	80
Introduction to Events, Signals, and Slots.....	80
Usage of Toolbar icons in Qt Designer.....	92
Signal slot examples in Qt Designer.....	93
Conclusion	106
Points to remember.....	106
Questions.....	107
4. Getting Insights of Button Widgets in Qt Designer.....	109
Introduction	109
Structure	110
Objectives	110
Push Button.....	110
<i>Important properties</i>	<i>111</i>
<i>autoDefault.....</i>	<i>111</i>
<i>default</i>	<i>111</i>
<i>flat.....</i>	<i>112</i>
<i>Important methods</i>	<i>112</i>
<i>Important signals</i>	<i>112</i>
Tool Button.....	121
<i>Important properties</i>	<i>121</i>
<i>popupMode</i>	<i>121</i>
<i>toolButtonStyle</i>	<i>122</i>
<i>autoRaise</i>	<i>122</i>
<i>arrowType</i>	<i>122</i>
<i>Important methods</i>	<i>123</i>
<i>Important signals</i>	<i>123</i>

Radio Button	131
<i>Important properties</i>	131
<i>Important methods</i>	132
<i>Important signals</i>	132
Check Box	136
<i>Important properties</i>	136
<i>tristate</i>	136
<i>Important methods</i>	137
<i>Important signals</i>	137
Command Link Button	144
<i>Important properties</i>	144
<i>Important methods</i>	144
<i>setDescription</i>	144
<i>Important signals</i>	145
Dialog Button Box	147
<i>Important properties</i>	148
<i>orientation</i>	148
<i>standardButtons</i>	148
<i>centerButtons</i>	149
<i>Important methods</i>	149
<i>Important signals</i>	150
Common properties for Button widgets	153
<i>objectName</i>	154
<i>enabled</i>	154
<i>geometry</i>	154
<i>sizePolicy</i>	155
<i>minimumSize</i>	156
<i>maximumSize</i>	156
<i>sizeIncrement</i>	157
<i>baseSize</i>	157
<i>palette</i>	157
<i>font</i>	158

<i>cursor</i>	158
<i>mouseTracking</i>	158
<i>tabletTracking</i>	159
<i>focusPolicy</i>	159
<i>contextMenuPolicy</i>	159
<i>acceptDrops</i>	160
<i>toolTip</i>	160
<i>toolTipDuration</i>	160
<i>statusTip</i>	161
<i>whatsThis</i>	161
<i>accessibleName</i>	161
<i>accessibleDescription</i>	161
<i>layoutDirection</i>	162
<i>autoFillBackground</i>	162
<i>stylesheet</i>	162
<i>locale</i>	163
<i>inputMethodHints</i>	163
<i>text</i>	164
<i>icon</i>	164
<i>iconSize</i>	165
<i>shortcut</i>	165
<i>checkable</i>	165
<i>checked</i>	166
<i>autoRepeat</i>	166
<i>autoExclusive</i>	166
<i>autoRepeatDelay</i>	167
<i>autoRepeatInterval</i>	167
Conclusion	167
Points to remember	168
Questions	169

5. Getting Insights of Item Views in Qt Designer.....	171
Introduction	171
Structure	172
Objectives	172
Data presentation	172
List view	173
Important properties	173
movement	173
flow	174
isWrapping	174
resizeMode	174
layoutMode	174
spacing	175
gridSize	175
viewMode	175
modelColumn	175
uniformItemSizes	176
batchSize	176
wordWrap	176
selectionRectVisible	176
Important methods of QAbstractItemView base class	176
Important signal of QAbstractItemView base class	177
Tree view	179
Important properties	180
autoExpandDelay	180
indentation	180
rootIsDecorated	180
uniformRowHeights	180
itemsExpandable	181
sortingEnabled	181
animated	181
allColumnsShowFocus	181

<i>wordWrap</i>	182
<i>headerHidden</i>	182
<i>expandsOnDoubleClick</i>	182
Table view	186
<i>Important properties</i>	187
<i>showGrid</i>	187
<i>gridStyle</i>	187
<i>sortingEnabled</i>	187
<i>wordWrap</i>	187
<i>cornerButtonEnabled</i>	188
Column view	191
<i>Important properties</i>	191
<i>resizeGripsVisible</i>	191
QFrame	194
<i>frameShape</i>	194
<i>frameShadow</i>	195
<i>lineWidth</i>	196
<i>midLineWidth</i>	196
QAbstractScrollArea	196
<i>verticalScrollBarPolicy</i>	196
<i>horizontalScrollBarPolicy</i>	197
<i>sizeAdjustPolicy</i>	197
QAbstractItemView	197
<i>autoScroll</i>	198
<i>autoScrollMargin</i>	198
<i>editTriggers</i>	198
<i>tabKeyNavigation</i>	199
<i>showDropIndicator</i>	199
<i>dragEnabled</i>	199
<i>dragDropOverwriteMode</i>	200
<i>dragDropMode</i>	200
<i>defaultDropAction</i>	201

<i>alternatingRowColors</i>	201
<i>selectionMode</i>	201
<i>selectionBehavior</i>	202
<i>iconSize</i>	203
<i>textElideMode</i>	203
<i>verticalScrollMode</i>	203
<i>horizontalScrollMode</i>	204
QStandardItemModel	204
Conclusion	204
Points to remember	205
Questions	205
6. Getting Insights of Item Widgets (Item-Based) in Qt Designer	207
Introduction	207
Structure	208
Objectives	208
List widget	208
<i>Important properties</i>	208
<i>currentRow</i>	208
<i>sortingEnabled</i>	209
<i>Important methods</i>	209
<i>Important signals</i>	209
Tree widget	218
<i>Important properties</i>	218
<i>Important methods</i>	219
<i>Important signals</i>	219
Table widget	227
<i>Important properties</i>	227
<i>rowCount</i>	228
<i>columnCount</i>	228
<i>Important methods</i>	228
<i>Important signals</i>	229
Conclusion	236

Points to remember	236
Questions.....	237
7. Getting Insights of Containers in Qt Designer	239
Introduction	239
Structure	240
Objectives	240
Group Box	240
<i>Important properties</i>	241
<i>title</i>	241
<i>alignment</i>	241
<i>flat</i>	241
<i>checkable</i>	241
<i>checked</i>	242
<i>Important methods</i>	242
<i>Important signals</i>	242
Scroll area	246
<i>Important properties</i>	246
<i>widgetResizable</i>	246
<i>alignment</i>	247
<i>Important methods</i>	247
<i>Important signals</i>	247
<i>scrollContentsBy(int dx, int dy)</i>	247
Toolbox	251
<i>Important properties</i>	251
<i>currentIndex</i>	251
<i>currentItemText</i>	251
<i>currentItemName</i>	251
<i>currentItemIcon</i>	252
<i>currentItemToolTip</i>	252
<i>tabSpacing</i>	252
<i>Important methods</i>	252
<i>Important signals</i>	253

<i>currentChanged(index)</i>	253
Tab widget.....	256
<i>Important properties</i>	257
<i>tabPosition</i>	257
<i>tabShape</i>	257
<i>currentIndex</i>	257
<i>iconSize</i>	257
<i>elideMode</i>	258
<i>userScrollButtons</i>	258
<i>documentMode</i>	258
<i>tabsClosable</i>	258
<i>movable</i>	259
<i>tabBarAutoHide</i>	259
<i>currentTabText</i>	259
<i>currentTabName</i>	259
<i>currentTabIcon</i>	259
<i>currentTabToolTip</i>	260
<i>currentTabWhatsThis</i>	260
<i>Important methods</i>	260
<i>Important signals</i>	261
<i>currentChanged</i>	261
<i>tabCloseRequested</i>	261
Stacked widget	265
<i>Important properties</i>	265
<i>currentIndex</i>	266
<i>Important methods</i>	266
<i>Important signals</i>	266
<i>currentChanged(arg_1)</i>	266
<i>widgetRemoved</i>	266
Frame	269
<i>Important properties</i>	270
<i>Important methods</i>	270

Widget.....	272
<i>Important properties</i>	272
<i>Important methods</i>	273
<i>Important signals</i>	273
MDI Area.....	275
<i>Important properties</i>	275
<i>background</i>	275
<i>activationOrder</i>	275
<i>viewMode</i>	276
<i>documentMode</i>	276
<i>tabsClosable</i>	276
<i>tabsMovable</i>	276
<i>tabShape</i>	277
<i>tabPosition</i>	277
<i>Important methods</i>	277
<i>Important signals</i>	278
<i>subWindowActivated(arg_1)</i>	278
Dock widget.....	281
<i>Important properties</i>	281
<i>floating</i>	281
<i>features</i>	281
<i>allowedAreas</i>	282
<i>windowTitle</i>	282
<i>dockWidgetArea</i>	282
<i>docked</i>	283
<i>Important methods</i>	283
<i>Important signals</i>	283
Conclusion	286
Points to remember.....	286
Questions.....	287

8. Getting Insights of Input Widgets in Qt Designer	289
Introduction	289
Structure	290
Objectives	291
Combo Box	291
Important properties	291
<code>editable</code>	291
<code>currentText</code>	292
<code>currentIndex</code>	292
<code>maxVisibleItems</code>	292
<code>maxCount</code>	292
<code>insertPolicy</code>	292
<code>sizeAdjustPolicy</code>	293
<code>minimumContentsLength</code>	293
<code>iconSize</code>	293
<code>duplicatesEnabled</code>	293
Frame	294
<code>modelColumn</code>	294
Important methods	294
Important signals	294
Font Combo Box	298
Important properties	298
<code>writingSystem</code>	298
<code>fontFilters</code>	298
<code>currentFont</code>	298
Important methods	299
Important signals	299
<code>currentFontChanged(QFont)</code>	299
Line Edit	301
Important properties	302
<code>inputMask</code>	302
<code>text</code>	303

<i>maxLength</i>	303
<i>frame</i>	303
<i>echoMode</i>	303
<i>cursorPosition</i>	304
<i>alignment</i>	304
<i>dragEnabled</i>	304
<i>readOnly</i>	304
<i>placeholderText</i>	305
<i>cursorMoveStyle</i>	305
<i>clearButtonEnabled</i>	305
<i>Important methods</i>	305
<i>Important signals</i>	306
TextEdit.....	313
<i>Important properties</i>	313
<i>autoFormatting</i>	313
<i>tabChangeFocus</i>	313
<i>documentTitle</i>	313
<i>undoRedoEnabled</i>	314
<i>lineWrapMode</i>	314
<i>lineWrapColumnOrWidth</i>	314
<i>readOnly</i>	314
<i>html</i>	314
<i>overwriteMode</i>	315
<i>tabStopWidth</i>	315
<i>tabStopDistance</i>	315
<i>acceptRichText</i>	315
<i>cursorWidth</i>	316
<i>textInteractionFlags</i>	316
<i>placeholderText</i>	316
<i>Important methods</i>	316
<i>Important signals</i>	317
Plain Text Edit.....	322

Important properties	322
plainText	322
maximumBlockCount	323
backgroundVisible	323
centerOnScroll	323
Important methods	324
Important signals	324
Spin Box	328
Important properties	328
wrapping	328
frame	329
alignment	329
readOnly	329
buttonSymbols	329
specialValueText	329
accelerated	330
correctionMode	330
keyboardTracking	330
showGroupSeparator	330
suffix	331
prefix	331
minimum	331
maximum	331
singlestep	332
value	332
displayIntegerBase	332
Important methods	332
Important signals	333
valueChanged(arg__1)	333
Double Spin Box	337
Important properties	337
decimals	337

Important methods	337
setDecimals(prec)	338
Important signals	338
valueChanged(arg__1)	338
Date/Time Edit.....	342
Important properties	342
dateTime	342
date	343
time.....	343
maximumDateTime.....	343
minimumDateTime	343
maximumDate.....	344
minimumDate	344
maximumTime.....	344
minimumTime	344
currentSection	345
displayFormat.....	345
calendarPopup.....	345
currentSectionIndex	345
timeSpec.....	346
Important methods/signals	346
Dial.....	350
Important properties	350
minimum	350
maximum.....	351
singleStep	351
pageStep.....	351
value	351
sliderPosition.....	351
tracking.....	352
orientation	352
invertedAppearance.....	352

<i>invertedControls</i>	352
<i>Properties of QDial</i>	353
<i>wrapping</i>	353
<i>notchTarget</i>	353
<i>notchesVisible</i>	353
<i>Important methods/signals</i>	353
QScrollBar	356
QSlider.....	361
<i>Important properties</i>	361
<i>tickPosition</i>	361
<i>tickInterval</i>	361
<i>Important methods/signals</i>	361
<i>Important properties</i>	362
Key Sequence Edit	365
<i>Important properties</i>	365
<i>keySequence</i>	365
<i>Important methods/signals</i>	366
Conclusion	367
Points to remember.....	368
Questions.....	368
9. Getting Insights of Display Widgets in Qt Designer	371
Introduction.....	371
Structure	371
Objectives	372
Introduction to the display widgets in Qt Designer	372
Label.....	372
<i>Important properties</i>	373
<i>text</i>	373
<i>textFormat</i>	373
<i>pixmap</i>	373
<i>scaledContents</i>	373
<i>alignment</i>	374

<i>wordWrap</i>	374
<i>margin</i>	374
<i>indent</i>	374
<i>openExternalLinks</i>	374
<i>textInteractionFlags</i>	375
<i>buddy</i>	375
<i>Important methods/signals</i>	375
Text browser.....	378
<i>Important properties</i>	379
<i>Source</i>	379
<i>searchPaths</i>	379
<i>openExternalLinks</i>	379
<i>openLinks</i>	379
<i>Important methods/signals</i>	380
Calendar widget.....	382
<i>Important properties</i>	383
<i>selectedDate</i>	383
<i>minimumDate</i>	383
<i>maximumDate</i>	384
<i>firstDayOfWeek</i>	384
<i>gridVisible</i>	384
<i>selectionMode</i>	385
<i>horizontalHeaderFormat</i>	385
<i>verticalHeaderFormat</i>	385
<i>navigationBarVisible</i>	385
<i>dateEditEnabled</i>	386
<i>dateEditAcceptDelay</i>	386
<i>Important methods/signals</i>	386
<i>selectionChanged()</i>	386
LCD number	389
<i>Important properties</i>	389
<i>smallDecimalPoint</i>	389

<i>digitCount</i>	389
<i>mode</i>	389
<i>segmentStyle</i>	390
<i>value</i>	390
<i>intValue</i>	390
<i>Important methods/signals</i>	391
<i>display(num)</i>	391
Progress bar	393
<i>Important properties</i>	394
<i>minimum</i>	394
<i>maximum</i>	394
<i>value</i>	394
<i>alignment</i>	394
<i>textVisible</i>	395
<i>orientation</i>	395
<i>invertedAppearance</i>	395
<i>textDirection</i>	395
<i>format</i>	396
<i>Important methods/signals</i>	396
<i>valueChanged(value)</i>	396
Conclusion	398
Points to remember	399
Questions	399
Index	401-409

CHAPTER 1

Introduction to PyQt5 and Qt Designer Tool

Introduction

In our previously launched book *Building Modern GUIs with tkinter and Python*, we learned how to create GUI forms using the Tk interface library. Now, we will be looking at another approach to create the same GUI forms using one of the most popular cross-platform libraries, known as PyQt5 library, which was developed by Riverbank Computing. A robust and cross-platform graphical toolkit called Qt has a Python binding called PyQt5. Python is a well-known and simple-to-learn programming language, and with PyQt5 developers, we can simply construct GUI applications using Python. With PyQt5, a visual layout tool called Qt Designer is included. Without having to write any code explicitly, it enables developers to construct GUI layouts quickly and easily by dragging and dropping widgets onto a canvas. Additionally, Qt Designer offers a selection of editable widgets that may be used to design distinctive user interfaces with eye-catching visuals.

Structure

In this chapter, we will discuss the following topics:

- Comparison of PyQt5 with tkinter library
- PyQt5 framework installation

- First GUI form creation using PyQt5 without using class
- GUI form creation using PyQt5 by using class
- Installation of Qt Designer tool with pre-defined templates
- Components of Qt Designer
- User Credential App Demo

Objectives

By the end of this chapter, the reader will be able to compare the powerful and cross-platform toolkit graphical PyQt5, with the tkinter library. We shall see how to install the PyQt5 framework installation, along with the creation of a basic GUI form using PyQt5, first without, and then using the class. Moreover, we will be looking into the components inside Qt Designer along with different pre-defined templates. Finally, we will be creating a user credential app, initially focusing on the view in Qt Designer (**.ui** file which is a XML file), then converting it into a Python code (**.py**) using **pyuic5** command, and finally creating a new Python file, which will be importing the Python code for user interface design and adding some useful logic to create a basic login application for the user.

Comparison of PyQt5 with tkinter library

There are numerous Qt classes in a set of Python modules, which are compatible with many operating systems such as iOS, Windows, Linux, Unix, Android, and so on, which is a part of the top-level Python package, that is, PyQt5 library. Qt5 in PyQt5 stands for **Qt version 5**. This library provides us the advantage of Python binding with the Qt C++ toolkit. An important point to note is that under the **GNU General Public License (GNU GPL or GPL) v3** license, the PyQt5 license is being released. One may wonder that if we have already learned about tkinter, then why we should learn about PyQt5 as well? To better understand this, let us check the advantages of PyQt5 over other library packages such as tkinter:

- **Flexibility in coding:** In order to establish simple communication between objects, there is a concept of signals and slots, which gives us flexibility for GUI programming using PyQt5, when dealing with events.
- **More than GUI toolkit:** Using PyQt5, we can build entire applications using its graphics, printer support, networking, database access, and so on. It is like an application framework.
- **Numerous UI components:** PyQt5 offers numerous widgets such as **QLabel**, **QPushButton**, **QCombobox**, and so on, such that each widget has some basic image well suited in all the platforms. Numerous advanced widgets are also available on the topic in this library.

- **Numerous study resources:** If there is no documentation, then you might wonder what the point of learning is. PyQt5 comes with a rich array of documentation since it is one of the most commonly used Python packages for GUI creation.
- **Easy to understand:** We can easily use previous knowledge of either Python, Qt or C++, thus making PyQt5 easy to understand.
- **Preferred choice of GUI developers:** Due to simplicity and ease of use, many GUI developers opt for functionalities that come with PyQt5 to develop their own applications.
- **GUI widget appearance:** The appearance of PyQt5 is nice and pleasing to the eyes.

Now, we may wonder which library to choose to create GUI forms using PyQt5 or tkinter. This generally depends on user application and the willingness to learn and explore. *Table 1.1* further shows the differences between the libraries:

Fundamental points	PyQt5	tkinter
Licensing	Commercial license is applicable if not complying to provide an application to the end user under GPL license.	It is free if needed to supply commercially.
Library	Vast	Small compared to PyQt5.
Understanding time	More	Less
Widgets appearance	Nice and modern look	Orthodox and quite old-fashioned
Provision of advanced widgets	Yes	No
Interfacing with other designer tool	Yes, Qt Designer	Qt Designer is not there for tkinter
Default supply of inbuilt libraries	No, we need to install it separately	Yes, as it comes along with default standard Python library
Concept of signals and slots for communication	Yes	No

Table 1.1: Differences between libraries PyQt5 and tkinter

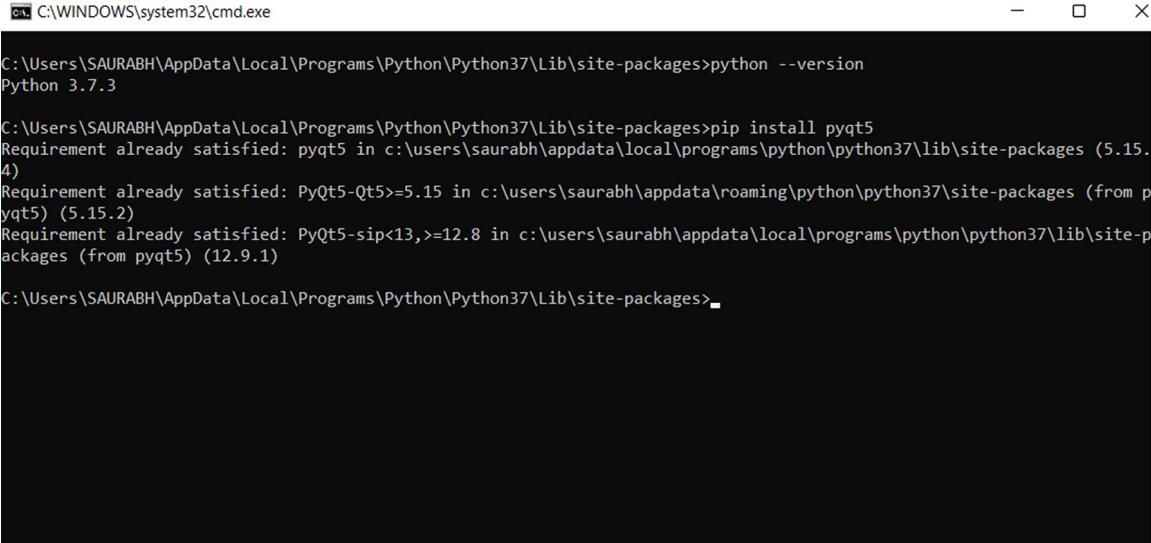
You can decide which library to choose to create GUI forms, depending on your application. We will later discuss how to create a **UI form** using Qt Designer in detail, as well as the widgets associated with it and the Python code with logic written by importing the auto-code generated by creating the UI form in detail.

PyQt5 framework installation

The Python version we will be using for discussing PyQt5 will be 3.7.3. (You can try working with new version as on today, that is, 3.11.5 dated Aug 24, 2023) From the location where we have installed our Python, first type the command **python --version** to check the version installed and then type the following command to install pyqt5:

pip install pyqt5

Refer to *Figure 1.1*:



```
C:\WINDOWS\system32\cmd.exe

C:\Users\SAURABH\AppData\Local\Programs\Python\Python37\Lib\site-packages>python --version
Python 3.7.3

C:\Users\SAURABH\AppData\Local\Programs\Python\Python37\Lib\site-packages>pip install pyqt5
Requirement already satisfied: pyqt5 in c:\users\saurabh\appdata\local\programs\python\python37\lib\site-packages (5.15.4)
Requirement already satisfied: PyQt5-Qt5>=5.15 in c:\users\saurabh\appdata\roaming\python\python37\site-packages (from pyqt5) (5.15.2)
Requirement already satisfied: PyQt5-sip<13,>=12.8 in c:\users\saurabh\appdata\local\programs\python\python37\lib\site-packages (from pyqt5) (12.9.1)

C:\Users\SAURABH\AppData\Local\Programs\Python\Python37\Lib\site-packages>
```

Figure 1.1: PyQt5 installation

Once installed, we can verify the installation of PyQt5 under Python site-packages folder, as shown in *Figure 1.2*:

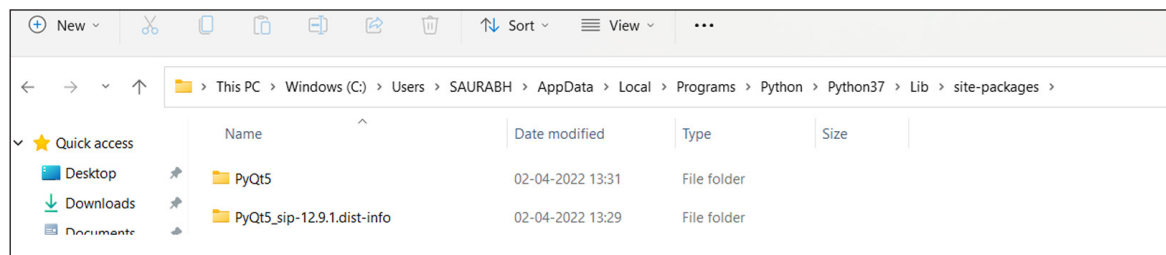


Figure 1.2: PyQt5 folder

We can check some of the modules present in PyQt5. If we do not get any error, then we can again cross verify that PyQt5 is successfully installed, as shown in *Figure 1.3*:

```
C:\Users\SAURABH>python
Python 3.7.3 (v3.7.3:ef4ec6ed12, Mar 25 2019, 22:22:05) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> from PyQt5 import QtGui
>>>
```

Figure 1.3: PyQt5 package verification

First GUI form creation using PyQt5 without using class

Now, we shall see a basic UI form creation using PyQt5 without using class. This is quite important. As you might have realized, most of the UI forms which we will be creating will be using the concept of class and object. It is highly recommended that you go through the concept of class before moving further in this chapter. It is well explained in our previous books, *Programming Techniques using Python* and *Python for Everyone*. Here, we will only focus on our basic UI form creation. So, without any further delay, let us start. Refer to the following code which can be checked in some IDEs like VSCode, Spyder, Anaconda and so on:

```
import sys # L1
from PyQt5.QtWidgets import QWidget, QApplication # L2

myapp = QApplication(sys.argv) # L3
mywindow = QWidget() # L4
mywindow.show() # L5
sys.exit(myapp.exec_()) # L6
```

Output:

The output can be seen in the following Figure 1.4:

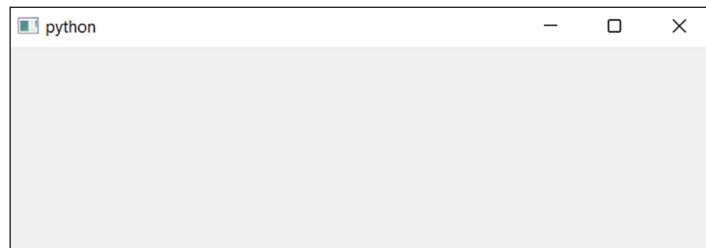


Figure 1.4: Output of Chap1_Example1.py

Note: The preceding code is covered in Program Name: Chap1_Example1.py