



There were issues affecting this run of Lighthouse:

- The page loaded too slowly to finish within the time limit. Results may be incomplete.



Performance

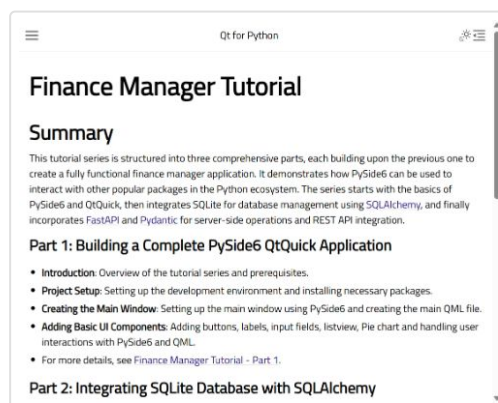
Values are estimated and may vary. The [performance score is calculated](#) directly from these metrics. [See calculator.](#)



0–49

50–89

90–100



METRICS

[Expand view](#)

First Contentful Paint

0.8 s

Largest Contentful Paint

1.0 s

Total Blocking Time

0 ms

Cumulative Layout Shift

0.029

Speed Index

0.8 s

View Treemap

<div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div> <td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div></td></td></td></td></td></td>	<div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div> <td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div></td></td></td></td></td>	<div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div> <td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div></td></td></td></td>	<div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div> <td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div></td></td></td>	<div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div> <td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div><td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div></td></td>	<div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div> <td><div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div></td>	<div><div>Q for Python</div><div>Finance Manager Tutorial</div><div>Summary</div><div><p>This tutorial series is structured into three comprehensive parts, each building upon a solid foundation of financial management applications. It demonstrates how Python, combined with other popular packages in the Python ecosystem, can be used to build a robust and scalable financial management system. The series starts with a high-level overview of the system architecture, followed by a detailed look at the data layer, and finally, the user interface.</p></div><div><div>Part 1: Building a Complete PySide6 QtQuick Application</div><div><ul style="list-style-type: none">• Introduction: Overview of the tutorial series and prerequisites.• Project Setup: Setting up the development environment and installing necessary dependencies.• Creating the Main Window: Setting up the main window using PySide6 and QtQuick.• Adding Basic UI Components: Adding buttons, labels, and text fields to the main window.• For more details, see Finance Manager Tutorial - Part 1.</div><div>Part 2: Integrating SQLite Database with SQLAlchemy</div></div></div>
---	--	---	--	---	--	--



Later this year, insights will replace performance audits. [Learn more and provide feedback](#)

[here.](#)

Try insights

Show audits relevant to: All FCP LCP TBT CLS

DIAGNOSTICS

▲ Use HTTP/2 — 23 requests not served via HTTP/2	▼
▲ Eliminate render-blocking resources — Est savings of 340 ms	▼
▲ Enable text compression — Est savings of 176 KiB	▼
▲ Minify JavaScript — Est savings of 155 KiB	▼
▲ Reduce unused JavaScript — Est savings of 167 KiB	▼
▲ Page prevented back/forward cache restoration — 2 failure reasons	▼
Image elements do not have explicit width and height	▼
Minify CSS — Est savings of 3 KiB	▼
Serve static assets with an efficient cache policy — 17 resources found	▼
Ensure text remains visible during webfont load	▼
Remove duplicate modules in JavaScript bundles — Est savings of 22 KiB	▼

Avoid serving legacy JavaScript to modern browsers — Est savings of 1 KiB	▼
Reduce unused CSS — Est savings of 18 KiB	▼
○ Avoid large layout shifts — 2 layout shifts found	▼
○ Avoid chaining critical requests — 17 chains found	▼
○ Minimize third-party usage — Third-party code blocked the main thread for 0 ms	▼
○ Largest Contentful Paint element — 1,030 ms	▼
○ Avoid long main-thread tasks — 2 long tasks found	▼

More information about the performance of your application. These numbers don't [directly affect](#) the Performance score.

PASSED AUDITS (20)

Show



Accessibility

These checks highlight opportunities to [improve the accessibility of your web app](#). Automatic detection can only detect a subset of issues and does not guarantee the accessibility of your web app, so [manual testing](#) is also encouraged.

NAMES AND LABELS

▲ Select elements do not have associated label elements.	▼
--	---

These are opportunities to improve the semantics of the controls in your application. This may enhance the experience for users of assistive technology, like a screen reader.

ADDITIONAL ITEMS TO MANUALLY CHECK (10)

Show

These items address areas which an automated testing tool cannot cover. Learn more in our guide on [conducting an accessibility review](#).

PASSED AUDITS (26)

Show

NOT APPLICABLE (30)

Show



Best Practices

GENERAL

▲ Browser errors were logged to the console	▼
---	---

TRUST AND SAFETY

○ Ensure CSP is effective against XSS attacks	▼
○ Use a strong HSTS policy	▼
○ Ensure proper origin isolation with COOP	▼
○ Mitigate clickjacking with XFO or CSP	▼

PASSED AUDITS (13)	Show
--------------------	------

NOT APPLICABLE (3)	Show
--------------------	------



SEO

These checks ensure that your page is following basic search engine optimization advice. There are many additional factors Lighthouse does not score here that may affect your search ranking, including performance on [Core Web Vitals](#). [Learn more about Google Search Essentials](#).

CONTENT BEST PRACTICES

▲ Document does not have a meta description	▼
---	---

Format your HTML in a way that enables crawlers to better understand your app’s content.

ADDITIONAL ITEMS TO MANUALLY CHECK (1)

Show

Run these additional validators on your site to check additional SEO best practices.

PASSED AUDITS (8)

Show

NOT APPLICABLE (1)

Show

Captured at Nov 2, 2025, 6:31 PM GMT+3
Initial page load

Emulated Desktop with
Lighthouse 12.6.0
Custom throttling

Single page session

Using Chromium 138.0.0.0 with devtools