

Cross Platform Desktop Applications Using Node Electron And Nw Js

In this book, you will create two desktop applications using Python GUI and MariaDB. 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 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 project-based guide to help you create, package, and deploy desktop applications on multiple platforms using modern JavaScript frameworks Key Features Use your web development skills with JavaScript and Node.js to build desktop applications for macOS and Windows Develop desktop versions of popular mobile applications that are similar to Slack, Spotify, and more Design desktop apps with automatic updates and real-time analytics capabilities Book Description The Electron framework allows you to use modern web technologies to build applications that share the same code across all operating systems and platforms. This also helps designers to easily transition from the web to the desktop. Electron Projects guides you through building cross-platform Electron apps with modern web technologies and JavaScript frameworks such as Angular, React.js, and Vue.js. You'll explore the process of configuring modern JavaScript frameworks and UI libraries, real-time analytics and automatic updates, and interactions with the operating system. You'll get hands-on with building a basic Electron app, before moving on to implement a Markdown Editor. In addition to this, you'll be able to experiment with major JavaScript frameworks such as Angular and Vue.js, discovering ways to integrate them with Electron apps for building cross-platform desktop apps. Later, you'll learn to build a screenshot snipping tool, a mini-game, and a music player, while also gaining insights into analytics, bug tracking, and licensing. You'll then get to grips with building a chat app, an eBook generator and finally a simple digital wallet app. By the end of this book, you'll have experience in building a variety of projects and project templates that will help you to apply your knowledge when creating your own cross-platform applications. What you will learn Initialize Node.js, Node Package Manager (NPM), and JavaScript to set up your app Integrate Phaser with Electron to build a simple 2D game Improve app quality by adding an error tracking system and crash reports Implement group chat features and event handling capabilities using Firebase Integrate a WordPress-like rich-text editor into your app Build Electron applications using a single codebase Who this book is for This book is for JavaScript developers who want to explore the Electron framework for building desktop apps. Working knowledge of modern frontend JavaScript frameworks and Node.js is assumed. No prior knowledge of desktop development is required.

"In this course, you will learn how to effectively apply your web development skills to the desktop paradigm, so you can get your desktop idea to market without any friction. The web platform is rapidly evolving, however there are plenty of applications that need more than what a web browser can offer. Electron is a popular open source framework that allows you to build desktop applications using the skills you already possess as a front-end developer. Electron is by far the easiest and fastest way to develop high quality cross-platform desktop applications. This course covers everything you need to know about the Electron framework, but most importantly, how to effectively translate your web front-end skills to the desktop paradigm."--Resource description page.

Modern Cross-Platform Development About This Book Build modern, cross-platform applications with .NET Core Get up to speed with C#, and up to date with all the latest features of C# 7 Start creating professional web applications with ASP.NET Core Who This Book Is For This book is targeted towards readers who have some prior programming experience or have a science, technology, engineering, or mathematics (STEM) background, and want to gain a solid foundation with C# and to be introduced to the types of applications they could build and will work cross-platform on Windows, Linux, and macOS. What You Will Learn Build cross-platform applications using C# 7 and .NET Core Explore ASP.NET Core and learn how to create professional web applications Improve your application's performance using multitasking Use Entity Framework Core and find out how to build code-first databases Master object-oriented programming with C# to increase code reuse and efficiency Familiarize yourself with cross-device app development using the Universal Windows Platform and XAML Query and manipulate data using LINQ Protect your data by using encryption and hashing In Detail If you want to build powerful cross-platform applications with C# 7 and .NET Core, then this book is for you. First, we'll run you through the basics of C#, as well as object-oriented programming, before taking a quick tour through the latest features of C# 7 such as tuples, pattern matching, out variables, and so on. After quickly taking you through C# and how .NET works, we'll dive into the .NET Standard 1.6 class libraries, covering topics such as performance, monitoring, debugging, serialization and encryption. The final section will demonstrate the major types of application that you can build and deploy cross-device and cross-platform. In this section, we'll cover Universal Windows Platform (UWP) apps, web applications, mobile apps, and web services. Lastly, we'll look at how you can package and deploy your applications so that they can be hosted on all of today's most popular platforms, including Linux and Docker. By the end of the book, you'll be armed with all the knowledge you need to build modern, cross-platform applications using C# and .NET Core. Style and approach This book takes a step-by-step approach and is filled with exciting projects and fascinating theory. It uses three high-impact sections to equip you with all the tools you'll need to build modern, cross-platform applications using C# and .NET Core.

Discover Golang's GUI libraries such as Go-GTK (GIMP Toolkit) and Go-Qt and build beautiful, performant, and responsive graphical applications. Key Features: Conceptualize and build state-of-art GUI applications with Golang (Go). Tackle the complexity of varying GUI application sizes with a structured and scalable approach. Get hands-on experience of GUI development with Shiny, and labs/ui, Fyne, and Walk. Book Description: Go is often compared to C++ when it comes to low-level programming and implementations that require faster processing, such as Graphical User Interfaces (GUIs). In fact, many claim that Go is superior to C++ in terms of its concurrency and ease of use. Most graphical application toolkits, though, are still written using C or C++, and so they don't enjoy the benefits of using a modern programming language such as Go. This guide to programming GUIs with Go 1.11 explores the various toolkits available, including UI, Walk, Shiny, and Fyne. The book compares the vision behind each project to help you pick the right approach for your project. Each framework is described in detail, outlining how you can build performant applications that users will love. To aid you further in creating applications using these emerging technologies, you'll be able to easily refer to code samples and screenshots featured in the book. In addition to toolkit-specific discussions, you'll cover more complex topics, such as how to structure growing graphical applications, and how cross-platform applications can integrate with each desktop operating system to create a seamless user experience. By delving into techniques and best practices for organizing and scaling Go-based graphical applications, you'll also glimpse Go's impressive concurrency system. In the concluding chapters, you'll discover how to distribute to the main desktop marketplaces and distribution channels. By the end of this book, you'll be a confident GUI developer who can use the Go language to boost the performance of your applications. What you will learn: Understand the benefits and complexities of building native graphical applications. Gain insights into how Go makes cross-platform graphical application development simple. Build platform-native GUI applications using andlabs/ui. Develop graphical Windows applications using Walk. Create multiplatform GUI applications using Shiny, Nuklear, and Fyne. Use Go wrappers for GTK and Qt for GUI application development. Streamline your requirements to pick the correct toolkit strategy. Who this book is for: This book is designed for Go developers who are interested in building native graphical applications for desktop computers and beyond. Some knowledge of building applications using Go is useful, but not essential. Experience in developing GUIs is not required as the book explores the benefits and challenges they pose. This book will also be beneficial for GUI application developers who are interested in trying Go.

Creating Cross-Platform C# Applications with Uno shows you how the Uno Platform helps developers familiar with developing Windows apps build applications for all operating systems and browsers. By learning how to develop apps for various business scenarios, you'll gain the confidence and knowledge needed to create your own cross-platform app.

Create 3 useful desktop applications with web technologies using Electron. About This Video: Gain proficiency in building desktop applications with Electron. Understand how to use Vanilla JS and React with Electron. Create custom menu items, system tray apps, and more. In Detail: Be ready to get hands-on in this interactive project-based course designed to help you build and package cross-platform desktop applications using Electron. If you didn't already know, many popular apps such as VS Code, Slack, and Skype are built on Electron! Throughout this course, you'll be building 3 apps - ImageShrink - An app to optimize images for websites. SystTop - A real-time CPU monitor with notifications and system tray. BugLogger - A CRUD app to track logs, which uses React and the MongoDB Atlas cloud database. In the first section of this course, you will start with an introduction to Electron and how it works. As you progress, you will start creating your first app- ImageShrink. You will also learn how to create menus with custom items. The next section will take you through the steps to create the interface, log files, and IPC communication. In addition to this, you will go on to build a menu template. In later sections, you'll get up to speed with working through 2 more applications, SystTop and BugLogger. The course will also help you focus on how to work with Interval/Dynamic System Stats. By the end of this course, you will be able to build desktop apps with Electron and even establish communication between processes with IPCMain and IPCRenderer.

Explore Qt Creator, Qt Quick, and QML to design and develop applications that work on desktop, mobile, embedded, and IoT platforms. Key Features: Build a solid foundation in Qt by learning about its core classes, multithreading, File I/O, and networking. Learn GUI programming and build custom interfaces using Qt Widgets, Qt Designer, and QML. Use the latest features of C++17 for improving the performance of your Qt applications. Book Description: Qt is a powerful development framework that serves as a complete toolset for building cross-platform applications, helping you reduce development time and improve productivity. Completely revised and updated to cover C++17 and the latest developments in Qt 5.12, this comprehensive guide is the third edition of Application Development with Qt Creator. You'll start by designing a user interface using Qt Designer and learn how to instantiate custom messages, forms, and dialogues. You'll then understand Qt's support for multithreading, a key tool for making applications responsive, and the use of Qt's Model-View-Controller (MVC) to display data and content. As you advance, you'll learn to draw images on screen using Graphics View Framework and create custom widgets that interoperate with Qt Widgets. This Qt programming book takes you through Qt Creator's latest features, such as Qt Quick Controls 2, enhanced CMake support, a new graphical editor for SCXML, and a model editor. You'll even work with multimedia and sensors using Qt Quick, and finally develop applications for mobile, IoT, and embedded devices using Qt Creator. By the end of this Qt book, you'll be able to create your own cross-platform applications from scratch using Qt Creator and the C++ programming language. What you will learn: Create programs from scratch using the Qt framework and C++ language. Compile and debug your Qt Quick and C++ applications using Qt Creator. Implement map view with your Qt application and display device location on the map. Understand how to call Android and iOS native functions from Qt C++ code. Localize your application with Qt Linguist. Explore various Qt Quick components that provide access to audio and video playbacks. Develop GUI applications using both Qt and Qt Quick. Who this book is for: If you are a beginner looking to harness the power of Qt and the Qt Creator framework for cross-platform development, this book is for you. Although no prior knowledge of Qt and Qt Creator is required, basic knowledge of C++ programming is assumed.

Create powerful cross-platform applications using C# 6, .NET Core 1.0, ASP.NET Core 1.0, and Visual Studio 2015. About This Book: Build modern, cross-platform applications with .NET Core 1.0. Get up-to-speed with C#, and up-to-date with all the latest features of C# 6. Start creating professional web applications with ASP.NET Core 1.0. Who This Book Is For: Are you struggling to get started with C#? Or maybe you're interested in the potential of the new cross-platform features that .NET Core can offer? If so, C# 6 and .NET Core 1.0 is the book for you. While you don't need to know any of the latest features of C# or .NET to get started, it would be beneficial if you have some programming experience. What You Will Learn: Build cross-platform applications using C# 6 and .NET Core 1.0. Explore ASP.NET Core 1.0 and learn how to create professional web applications. Improve your application's performance using multitasking. Use Entity Framework Core 1.0 and learn how to build Code-First databases. Master object-oriented programming with C# to increase code reuse and efficiency. Familiarize yourself with cross-device app development using the Universal Windows Platform and XAML. Query and manipulate data using LINQ. Protect your data by using encryption and hashing. In Detail: With the release of .NET Core 1.0, you can now create applications for Mac OS X and Linux, as well as Windows, using the development tools you know and love. C# 6 and .NET Core 1.0 has been divided into three high-impact sections to help start putting these new features to work. First, we'll run you through the basics of C#, as well as object-orient programming, before taking a quick tour through the latest features of C# 6 such as string interpolation for easier variable value output, exception filtering, and how to perform static class imports. We'll also cover both the full-feature, mature .NET Framework and the new, cross-platform .NET Core. After quickly taking you through C# and how .NET works, we'll dive into the internals of the .NET class libraries, covering topics such as performance, monitoring, debugging, internationalization, serialization, and encryption. We'll look at Entity Framework Core 1.0 and how to develop Code-First entity data models, as well as how to use LINQ to query and manipulate that data. The final section will demonstrate the major types of applications that you can build and deploy cross-device and cross-

platform. In this section, we'll cover Universal Windows Platform (UWP) apps, web applications, and web services. Lastly, we'll help you build a complete application that can be hosted on all of today's most popular platforms, including Linux and Docker. By the end of the book, you'll be armed with all the knowledge you need to build modern, cross-platform applications using C# and .NET Core. Style and approach This book takes a step-by-step approach and is filled with exciting projects and fascinating theory. It uses three high-impact sections to equip you with all the tools you'll need to build modern, cross-platform applications using C# and .NET Core.

Learning Delphi by example! Delphi has been established as one of the best software development tools for Windows desktop applications. Since its inception in 1995, it has constantly grown with the challenges of the IT industry. Today, applications often must be deployable to multiple platforms or mobile devices and offer non-proprietary interfaces for interaction with other software or hardware. TMS Software has been offering software components and products specifically designed for Delphi developers since the first version of Delphi. From recent seminars, conferences, and events TMS realized that practical examples how to make use of their components and products have the greatest learning effect. This idea led to the Hands-on series. In this first book of the Hands-on series, several examples for a reproducible usage scenario are built. Write a professional web service for a database with over 8 million records. Design VCL, FireMonkey and TMS WEB Core web clients for multiple platforms and devices. Implement a reporting service with XData. Generate signed PDF documents with a customizable template. Add a token-based login to your services. Build user interfaces with vector images to be ready for any display resolution. Encrypt sensitive data using modern algorithms. Use parallel programming to build responsive client applications. Write better code with static code analysis. Examples use TMS FixInsight, FlexCel, FNC Core, FNC UI Pack, VCL UI Pack, WEB Core, and XData. Embarcadero Delphi Rio 10.3 or newer suggested. Trial versions suffice to run the example projects. MySQL is used as database server. Any other database can be used that is compatible with XData. Web/XData basics explained in "TMS WEB Core: Web Application Development with Delphi" are not included again but referenced.

Cross-platform Desktop Applications With Node, Electron, and Nw.js Manning Publications

Learn to rapidly build and deploy cross-platform applications from a single codebase with practical, real-world solutions using the mature Delphi 10.4 programming environment Key Features Implement Delphi's modern features to build professional-grade Windows, web, mobile, and IoT applications and powerful servers Become a Delphi code and project guru by learning best practices and techniques for cross-platform development Deploy your complete end-to-end application suite anywhere Book Description Delphi is a strongly typed, event-driven programming language with a rich ecosystem of frameworks and support tools. It comes with an extensive set of web and database libraries for rapid application development on desktop, mobile, and internet-enabled devices. This book will help you keep up with the latest IDE features and provide a sound foundation of project management and recent language enhancements to take your productivity to the next level. You'll discover how simple it is to support popular mobile device features such as sensors, cameras, and GPS. The book will help you feel comfortable working with FireMonkey and styles and incorporating 3D user interfaces in new ways. As you advance, you'll be able to build cross-platform solutions that not only look native but also take advantage of a wide array of device capabilities. You'll also learn how to use embedded databases, such as SQLite and InterBase ToGo, synchronizing them with your own custom backend servers or modules using the powerful RAD Server engine. The book concludes by sharing tips for testing and deploying your end-to-end application suite for a smooth user experience. By the end of this book, you'll be able to deliver modern enterprise applications using Delphi confidently. What you will learn Discover the latest enhancements in the Delphi IDE Overcome the barriers that hold you back from embracing cross-platform development Become fluent with FireMonkey controls, styles, LiveBindings, and 3D objects Build Delphi packages to extend RAD Server or modularize your applications Use FireDAC to get quick and direct access to any data Leverage IoT technologies such as Bluetooth and Beacons and learn how to put your app on a Raspberry Pi Enable remote apps with backend servers on Windows and Linux through REST APIs Develop modules for IIS and Apache web servers Who this book is for This book is for Delphi developers interested in expanding their skillset beyond Windows programming by creating professional-grade applications on multiple platforms, including Windows, Mac, iOS, Android, and back-office servers. You'll also find this book useful if you're a developer looking to upgrade your knowledge of Delphi to keep up with the latest changes and enhancements in this powerful toolset. Some Delphi programming experience is necessary to make the most out of this book.

JavaScript is the little scripting language that could. Once used chiefly to add interactivity to web browser windows, JavaScript is now a primary building block of powerful and robust applications. In this practical book, new and experienced JavaScript developers will learn how to use this language to create APIs as well as web, mobile, and desktop applications. Author and engineering leader Adam D. Scott covers technologies such as Node.js, GraphQL, React, React Native, and Electron. Ideal for developers who want to build full stack applications and ambitious web development beginners looking to bootstrap a startup, this book shows you how to create a single CRUD-style application that will work across several platforms. Explore GraphQL's simple process for querying data Learn about shared authentication for APIs, web apps, and native applications Build performant web applications with React and Styled Components Use React Native to write cross-platform applications for iOS and Android that compile to native code Learn how to write desktop applications with Electron

Electron is currently the most popular framework and runtime for creating cross-platform desktop applications with web technologies. Maintained by GitHub, Electron combines Chromium's rendering library with Node.js and C++ to power apps such as Slack and Visual Studio Code. This short, hands-on guide introduces each aspect of developing and distributing your own apps with this framework. Author Felix Rieseberg, a staff engineer at Slack, begins by explaining how Electron's modules and APIs generally work. You'll then learn how to build an app from ground up and how to integrate it with your operating system for a native user experience. Learn Electron's fundamental patterns and most commonly used modules and APIs Build flexible user interfaces that can display windows in any size, shape, or opacity Understand the process for shipping an Electron app: packaging, creating installers, and enabling automatic updates Explore core considerations for developing apps with Electron: performance, community support, and framework shortcomings

Enhance your cross-platform programming abilities with the powerful features and capabilities of Qt 6 Key Features Leverage Qt and C++ capabilities to create modern, cross-platform applications that can run on a wide variety of software applications Explore what's new in Qt 6 and understand core concepts in depth Build professional customized GUI applications with the help of Qt Creator Book Description Qt is a cross-platform application development framework widely used for developing applications that can run on a wide range of hardware platforms with little to no change in the underlying codebase. If you have basic knowledge of C++ and want to build desktop or mobile applications with a modern graphical user interface (GUI), Qt is the right choice for you. Cross-Platform Development with Qt 6 and Modern C++ helps you understand why Qt is one of the favorite GUI frameworks adopted by industries worldwide, covering the essentials of programming GUI apps across a multitude of platforms using the standard C++17 and Qt 6 features. Starting with the fundamentals of the Qt framework, including the features offered by Qt Creator, this practical guide will show you how to create classic user interfaces using Qt Widgets and touch-friendly user interfaces using Qt Quick. As you advance, you'll explore the Qt Creator IDE for developing applications for multiple desktops as well as for embedded and mobile platforms. You will also learn advanced concepts about signals and slots. Finally, the book takes you through debugging and testing your app with Qt Creator IDE. By the end of this book, you'll be able to build cross-

platform applications with a modern GUI along with the speed and power of native apps. What you will learn Write cross-platform code using the Qt framework to create interactive applications Build a desktop application using Qt Widgets Create a touch-friendly user interface with Qt Quick Develop a mobile application using Qt and deploy it on different platforms Get to grips with Model/View programming with Qt Widgets and Qt Quick Discover Qt's graphics framework and add animations to your user interface Write test cases using the Qt Test framework and debug code Build a translation-aware application Follow best practices in Qt to write high-performance code Who this book is for This book is for application developers who want to use C++ and Qt to create modern, responsive applications that can be deployed to multiple operating systems such as Microsoft Windows, Apple macOS, and Linux desktop platforms. Although no prior knowledge of Qt is expected, beginner-level knowledge of the C++ programming language and object-oriented programming system (OOPs) concepts will be helpful.

This book constitutes revised selected papers from the 13th International Conference on Web Information Systems and Technologies, WEBIST 2017, held in Porto, Portugal, in April 2017. The purpose of the WEBIST series of conferences is to bring together researchers, engineers and practitioners interested in technological advances and business applications of web-based information systems. The 12 full papers presented in this volume were carefully reviewed and selected from originally 77 paper submissions. They contribute to the understanding of relevant trends of current research on Web information systems and technologies, comprising unified interfaces, Progressive Web Apps (PWAs) as well as a mobile device taxonomy, XML and open data processing, the history of Web engineering, web development for end-users, access control, Web platform assessment, rule engines, and scientific blogging.

Mashups are the integration of data from different sources to create one unified experience. Adobe's AIR (Adobe Integrated Runtime) platform is revolutionary in that it allows web developers to create applications using familiar technologies (such as Flex, Flash and HTML) but now they can deploy them on the desktop rather than having them locked to a web browser. This book covers mashups from a web designer and developer point of view, rather than a programmer. It covers everything from the basics and background of mashups to advanced functionality and integrating mashups with the desktop.

Build an awesome production-ready cross-platform desktop application using Electron and React.js About This Video Use the core functionalities of the latest Electron 5.0 to develop a doctor appointments application Integrate React with Electron to create cross-platform user interfaces Utilize the APIs to increase the efficiency and performance of your applications In Detail Are you building an application across multiple OS like Windows, macOS, and Linux? Electron framework allows you to build user interfaces for all the platforms at the same time. This course is designed to help you integrate Electron with React and create advanced UI. In this course, you will learn everything you need to know about Electron. You will start by setting up your environment and create the basic layouts to render the application window using HTML windows and IPC. All the concepts will be implemented in a real-world application to give you a build-as-you-learn experience. Utilize the Electron APIs to build your application functionalities. Finally, integrate Electron with React to create powerful user interfaces. You will also learn to deploy your application for both Windows and iOS systems. By the end of the course, you will have the skills to build powerful desktop apps with GUIs that wow your clients using Electron.js.

Build powerful cross-platform desktop applications with web technologies such as Node, NW.JS, Electron, and React About This Book Build different cross-platform HTML5 desktop applications right from planning, designing, and deployment to enhancement, testing, and delivery Forget the pain of cross-platform compatibility and build efficient apps that can be easily deployed on different platforms. Build simple to advanced HTML5 desktop apps, by integrating them with other popular frameworks and libraries such as Electron, Node.JS, Nw.js, React, Redux, and TypeScript Who This Book Is For This book has been written for developers interested in creating desktop applications with HTML5. The first part requires essential web-master skills (HTML, CSS, and JavaScript). The second demands minimal experience with React. And finally for the third it would be helpful to have a basic knowledge of React, Redux, and TypeScript. What You Will Learn Plan, design, and develop different cross-platform desktop apps Application architecture with React and local state Application architecture with React and Redux store Code design with TypeScript interfaces and specialized types CSS and component libraries such as Photonkit, Material UI, and React MDL HTML5 APIs such as desktop notifications, WebSockets, WebRTC, and others Desktop environment integration APIs of NW.js and Electron Package and distribute for NW.JS and Electron In Detail Building and maintaining cross-platform desktop applications with native languages isn't a trivial task. Since it's hard to simulate on a foreign platform, packaging and distribution can be quite platform-specific and testing cross-platform apps is pretty complicated. In such scenarios, web technologies such as HTML5 and JavaScript can be your lifesaver. HTML5 desktop applications can be distributed across different platforms (Window, MacOS, and Linux) without any modifications to the code. The book starts with a walk-through on building a simple file explorer from scratch powered by NW.JS. So you will practice the most exciting features of bleeding edge CSS and JavaScript. In addition you will learn to use the desktop environment integration API, source code protection, packaging, and auto-updating with NW.JS. As the second application you will build a chat-system example implemented with Electron and React. While developing the chat app, you will get Photonkit. Next, you will create a screen capturer with NW.JS, React, and Redux. Finally, you will examine an RSS-reader built with TypeScript, React, Redux, and Electron. Generic UI components will be reused from the React MDL library. By the end of the book, you will have built four desktop apps. You will have covered everything from planning, designing, and development to the enhancement, testing, and delivery of these apps. Style and approach Filled with real world examples, this book teaches you to build cross-platform desktop apps right from scratch using a step-by-step approach.

The Fyne GUI toolkit solves many of the challenges relating to traditional technologies and older programming languages. This book introduces the key APIs and techniques behind Fyne applications that make them easy to build. From the basics through to building five completed applications, you'll get up to speed with every stage of app development.

This book provides a practical guide to planning and building a professional Flex 2 application from start to finish. Written by a team of expert developers, the book details, step-by-step, how the authors built the popular Rich Media X (RMX) application for Adobe user groups, step by step. Coverage provides all the information needed to implement several killer web 2.0 features in Flex, including a blog, wiki, events calendar, and more. It is packed with essential practical techniques in use at a professional production environment, which you can learn from and use in your own Flex applications.

Electron is an open-source framework for creating desktop applications with your favorite web technologies: JavaScript, HTML, and CSS. It makes it easy to create a simple app with a few lines of code, using languages you already know. Want to learn more? Join Ray Villalobos as he walks through the techniques you need to master cross-platform desktop development with

Electron. Learn how to configure and communicate between app windows (web pages) via the main and renderer processes; use Vue.js-the progressive JavaScript framework-with Electron; and improve your apps by modifying components and adding modals. Plus, see how to finish up an app by customizing menus and adding an icon for your dock.

Electron is a framework for building cross-platform desktop applications with web technologies. It combines Google Chrome's content module with Node.js, letting users use their web development skill set to build applications that run natively on all major platforms. Electron in Action guides the reader, step-by-step, as they learn to build desktop applications that run on Windows, OSX, and Linux. By the end, readers will be ready to build their own professional desktop applications using the web tools and technologies they already know. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.

Summary Electron in Action guides you, step-by-step, as you learn to build cross-platform desktop applications that run on Windows, OSX, and Linux. By the end of the book, you'll be ready to build simple, snappy applications using JavaScript, Node, and the Electron framework. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications. About the Technology Wouldn't it be great to build desktop applications using just your web dev skills? Electron is a framework designed for exactly that! Fully cross-platform, Electron lets you use JavaScript and Node to create simple, snappy desktop apps. Spinning up tools, games, and utilities with Electron is fast, practical, and fun! About the Book Electron in Action teaches you to build cross-platform applications using JavaScript, Node, and the Electron framework. You'll learn how to think like a desktop developer as you build a text tool that reads and renders Markdown. You'll add OS-specific features like the file system, menus, and clipboards, and use Chromium's tools to distribute the finished product. You'll even round off your learning with data storage, performance optimization, and testing. What's inside Building for macOS, Windows, and Linux Native operating system APIs Using third-party frameworks like React Deploying to the Mac App Store About the Reader Requires intermediate JavaScript and Node skills. No experience building desktop apps required. About the Author Steven Kinney is a principal engineer at SendGrid, an instructor with Frontend Masters, and the organizer of the DinosaurJS conference in Denver, Colorado. Table of Contents PART 1 - GETTING STARTED WITH ELECTRON Introducing Electron Your first Electron application PART 2 - BUILDING CROSS-PLATFORM APPLICATIONS WITH ELECTRON Building a notes application Using native file dialog boxes and facilitating interprocess communication Working with multiple windows Working with files Building application and context menus Further operating system integration and dynamically enabling menu items Introducing the tray module Building applications with the menubar library Using transpilers and frameworks Persisting use data and using native Node.js modules Testing applications with Spectron PART 3 - DEPLOYING ELECTRON APPLICATIONS Building applications for deployment Releasing and updating applications Distributing your application through the Mac App Store

Discover how to take your existing web development skills and learn how to create desktop applications for macOS, Windows, and Linux, using GitHub's Electron. Learn how to combine the power of Node.js and Chromium to provide a powerful development platform for creating web applications that break free from the browser. Electron: From Beginner to Pro guides you through the capabilities that you have available to create desktop applications. Learn to use features like file system access, create native menus, OS-specific dialogs and more. The authors will show you how to package your application for distribution for multiple platforms and enable auto-updating. What You Will Learn Leverage your knowledge of HTML, CSS and JavaScript Use current web applications for the desktop Create and use Electron's main process and render process to create effective desktop applications Communicate between processes and between windows Build desktop applications that can be updated and distributed Who This Book Is For Web developers looking to leverage their HTML, CSS and JavaScript skills to create desktop widgets and applications. Developers wanting to leverage existing a Web application to extend functionality with a desktop application.

Learn how to develop cross-platform desktop app from scratch with Electron and NodeAbout This Book * Build a solid foundation with Electron for an easier development experience* Use modern JavaScript frameworks and tools along with Electron to take your desktop applications to the next level* Extend the functionality of Electron through modulesWho This Book Is ForIf you are a developer with prior experience of building front-end applications and you are keen on developing a cross-platform desktop application, then this book is for you. This book is also ideal for experienced JavaScript developers with a basic understanding of front-end development and Node.js development.What You Will Learn * Explore various tools and libraries to build and debug an Electron application* Use popular JavaScript frameworks such as Angular and Typescript along with Electron to enhance your app* Work with the desktop UI development for Electron using Photon* Find out how to use various Electron APIs like Clipboard, Process, Shell, Image, File, Session, and Cookie* Integrate your application into different desktop environments with Electron API* Cache your network resources using service worker* Test the Electron application using Mocha and Spectron* See how to package and distribute an Electron applicationIn Detail Though web applications are becoming increasingly popular, desktop apps are still important. The Electron framework lets you write cross-platform desktop applications using JavaScript, HTML, and CSS, and this book will teach you how to create your first desktop application with Electron. It will guide you on how to build desktop applications that run on Windows, Mac, and Linux platforms.You will begin your journey with an overview of Electron, and then move on to explore the various stages of creating a simple social media application. Along the way, you will learn how to use advanced Electron APIs, debug an Electron application, and make performance improvements using the Chrome developer tools. You'll also find out how to package and distribute an application, and more.By the end of the book, you will be able to build a complete desktop application using Electron and web technologies. You will have a solid understanding of the common challenges that desktop app developers face, and you'll know how to solve them.Style and approachCovers everything you need to know about Electron with full examples and explanations to get you building desktop apps with Electron as quickly as possible.

Learn how to use Electron to create compelling cross-platform desktop apps with a few lines of JavaScript, HTML, and CSS code.

It's tedious and time-consuming to create OS-specific versions of every desktop application, especially with different set of tools for each platform. NW.js radically simplifies desktop development, providing a true cross-platform development stack built on HTML, CSS and JavaScript, Node.js modules, and the Chrome Blink engine. NW.js applications interact with the host operating system just like any other platform-native project, so developers have full access to all local files and resources. The performance is great, and best of all, it saves developers time because they only have to write one version of their application. "Cross-Platform Desktop Applications" guides readers step-by-step as they learn to develop NW.js desktop applications that run on Windows, OSX and Linux. They begin by getting the big picture of what NW.js can (and can't) do. Readers test drive NW.js as they build their first desktop application. Next, they find out how to take advantage of OS- specific features like menus, system-tray apps, clipboards, and the file system. Along the way, this book teaches how to debug errors and diagnose performance bottlenecks with NW.js's browser developer tools, package an application as a standalone executable for each OS, and even distribute it to various app stores. The book gives developers the inspiration and skills they need to code pro quality desktop applications using the web languages they

already know. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.

Mastering Java and Objective-C to write high-quality Android and iOS apps isn't necessary with the Titanium mobile development toolkit, which can be used to create native mobile applications using standard web technologies like XML, TSS (similar to CSS and HTML), and JavaScript. Titanium compiles code against native platform SDKs to produce apps with the same features made possible using Objective-C and Java. Alloy, the open source MVC framework designed for Titanium, enforces good application design helping keep app's UI, business logic, and data separate so code is easier to read and maintain. Titanium Alloy in Action introduces Titanium and the Alloy MVC framework to web developers and graphic designers new to mobile app development. It begins with easy steps, laying out the tools and orienting readers to the world of mobile development, allowing them to quickly turn their designs into fully working apps. Using real-life examples, the book explains how to use XML, TSS, and JavaScript to build fully-native iOS and Android apps from a single code base, connect them to a cloud server, and deploy them to app stores. Along the way, it points out the main differences between iOS and Android, their visual language and UX paradigms, and the best approach to cross-platform mobile development. Purchase of the print book includes a free eBook in PDF, Kindle, and ePub formats from Manning Publications.

A comprehensive guide for beginners to learn the key concepts, real-world applications, and latest features of C# 10 and .NET 6 with hands-on exercises using Visual Studio 2022 and Visual Studio Code

Key Features Explore the newest additions to C# 10, the .NET 6 class library, and Entity Framework Core 6 Create professional websites and services with ASP.NET Core 6 and Blazor Build cross-platform apps for Windows, macOS, Linux, iOS, and Android

Book Description This latest edition is extensively revised to accommodate all the latest features that come with C# 10 and .NET 6. You will learn object-oriented programming, writing, testing, and debugging functions, implementing interfaces, and inheriting classes. The book covers the .NET APIs for performing tasks like managing and querying data, monitoring and improving performance, and working with the filesystem, async streams, serialization, and encryption. It provides examples of cross-platform apps you can build and deploy, such as websites and services using ASP.NET Core. The best application for learning the C# language constructs and many of the .NET libraries does not distract with unnecessary application code. Hence, the C# and .NET topics covered in Chapters 1 to 12 feature console applications. In Chapters 13 to 17, having mastered the basics, you will build practical applications and services using ASP.NET Core, the Model-View-Controller (MVC) pattern, and Blazor. There are also two new online chapters on using .NET MAUI to build cross-platform apps and building services using a variety of technologies, including Web API, OData, gRPC, GraphQL, SignalR, and Azure Functions. What you will learn Build rich web experiences using Blazor, Razor Pages, the Model-View-Controller (MVC) pattern, and other features of ASP.NET Core Build your own types with object-oriented programming Write, test, and debug functions Query and manipulate data using LINQ Integrate and update databases in your apps using Entity Framework Core, Microsoft SQL Server, and SQLite Build and consume powerful services using the latest technologies, including gRPC and GraphQL Build cross-platform apps using .NET MAUI and XAML

Who this book is for This book is for beginners to C# and .NET or programmers who have worked with C# in the past but feel left behind by the changes in the past few years. This book doesn't expect you to have any C# or .NET experience; however, you should have a general understanding of programming. Students and professionals with a science, technology, engineering, or mathematics (STEM) background can benefit from this book.

Develop powerful cross-platform applications with Xamari

About This Book* Write native cross-platform applications with Xamarin* Design user interfaces that can be shared across Android, iOS, and Windows Phone using Xamarin.Forms* This step-by-step practical guide will teach you various strategies for cross-platform development

Who This Book Is For If you are a developer with experience in C# and are just getting into mobile development, this is the book for you. If you have experience with desktop applications or the web, this book will give you a head-start with cross-platform development.

What you will learn* Understand Apple's MVC design pattern* Get to grips with the Android activity lifecycle* Share C# code across platforms and call native Objective-C or Java libraries from C#* Create a real web service back end in Windows Azure using SQL Azure as database storage* Set up third-party libraries such as NuGet and Objective Sharpie in many different ways, and port a desktop .NET library to Xamarin* Use Xamarin.Mobile for camera, contacts, and location

In Detail Xamarin is a leading cross-platform application development tool used by top companies such as Coca-Cola, Honeywell, and Alaska Airlines to build apps. In version 4, there are significant updates to the platform including the release of Xamarin.Forms 2.0, and improvements have been made to the iOS and Android designers. With Xamarin being acquired by Microsoft, it is now a part of Visual Studio family.

This book takes a step-by-step approach, teaching you how to build applications for iOS, Android, and Windows. We will walk you through a popular application, complete with a back-end web service and native features such as GPS location, camera, push notifications, and other core features. Additionally, you'll learn how to use external libraries with Xamarin and Xamarin.Forms to create user interfaces. We also provide instructions for Visual Studio and Windows. This edition has been updated with new screenshots and detailed steps to provide you with a holistic overview of the new features incorporated in Xamarin 4.

Learn the fundamentals of QT 5 framework to develop interactive cross-platform applications

Key Features A practical guide on the fundamentals of application development with QT 5 Learn to write scalable, robust and adaptable C++ code with QT Deploy your application on different platforms such as Windows, Mac OS, and Linux

Book Description Qt is a mature and powerful framework for delivering sophisticated applications across a multitude of platforms. It has a rich history in the Linux world, is widely used in embedded devices, and has made great strides in the Mobile arena over the past few years. However, in the Microsoft Windows and Apple Mac OS X worlds, the dominance of C#/.NET and Objective-C/Cocoa means that Qt is often overlooked. This book demonstrates the power and flexibility of the Qt framework for desktop application development and shows how you can write your application once and deploy it to multiple operating systems. Build a complete real-world line of business (LOB) solution from scratch, with distinct C++ library, QML user interface, and QTest-driven unit-test projects. This is a suite of essential techniques that cover the core requirements for most LOB applications and will empower you to progress from a blank page to shipped application. What you will learn

- Install and configure the Qt Framework and Qt Creator IDE
- Create a new multi-project solution from scratch and control every aspect of it with QMake
- Implement a rich user interface with QML
- Learn the fundamentals of QTest and how to integrate unit testing
- Build self-aware data entities that can serialize themselves to and from JSON
- Manage data persistence with SQLite and CRUD operations
- Reach out to the internet and consume an RSS feed
- Produce application packages for distribution to other users

Who this book is for This book is for application developers who want a powerful and flexible framework to create modern, responsive applications on Microsoft Windows, Apple Mac OS X, and Linux desktop platforms. You should be comfortable with C++ but no prior knowledge of Qt or QML is required.

A comprehensive and in-depth guide to writing functional programs using F#

About This Book Learn how to manage, run, and automate your servers using Puppet Explore how to use F# to develop large-scale applications quickly and simply, and become more productive in today's age of cloud computing and multi-core programming

This easy-to-follow guide is packed with real-world examples that will jump-start you with F# development on the .NET platform

Who This Book Is For If you are a C# developer with a basic knowledge of F# and want to explore the functional programming paradigm further to master your F# skills, then this book is for you.

What You Will Learn Understand the basics of F# and organize F# source code with Visual Studio Work with F# data structures and create functional data structures in F# interoperate with C# Build and use asynchronous programming patterns with F# Create and use type providers that

help perform data analysis from within Visual Studio Develop applications with pure F# code in WPF or ASP.NET MVC Find out how to perform distributed programming with ServiceBus or ZeroMQ Visualize data with charts, and work with Excel and R language Type providers In Detail F# is a multi-paradigm programming language that encompasses object-oriented, imperative, and functional programming language properties. Now adopted in a wide range of application areas and is supported both by industry-leading companies who provide professional tools and by an active open community, F# is rapidly gaining popularity as it emerges in digital music advertising, creating music-focused ads for Spotify, Pandora, Shazam, and anywhere on the web. This book will guide you through the basics and will then help you master F#. The book starts by explaining how to use F# with Visual Studio, file ordering, and the differences between F# and C# in terms of usage. It moves on to explain the functional core of F# such as data types, type declarations, immutability, strong type interference, pattern matching, records, F# data structures, sequence expressions, and lazy evaluation. Next, the book takes you through imperative and asynchronous programming, F# type providers, applications, and testing in F#. Finally, we look into using F# with distributed programming and using F# as a suitable language for data science. In short, this book will help you learn F# for real-world applications and increase your productivity with functional programming. Style and approach This easy-to-follow guide with syntaxes will help you master the concepts of F#. Packed with in-depth examples of real-world uses, this book covers each topic in detail with a reference to C#, so you will understand the difference between the languages.

If you are a developer with experience in C# and are just getting into mobile development, this is the book for you. If you have experience with desktop applications or the Web, this book will give you a head start on cross-platform development.

This book contains selected papers from the 9th International Conference on Information Science and Applications (ICISA 2018) and provides a snapshot of the latest issues encountered in technical convergence and convergences of security technology. It explores how information science is core to most current research, industrial and commercial activities and consists of contributions covering topics including Ubiquitous Computing, Networks and Information Systems, Multimedia and Visualization, Middleware and Operating Systems, Security and Privacy, Data Mining and Artificial Intelligence, Software Engineering, and Web Technology. The proceedings introduce the most recent information technology and ideas, applications and problems related to technology convergence, illustrated through case studies, and reviews converging existing security techniques. Through this volume, readers will gain an understanding of the current state-of-the-art information strategies and technologies of convergence security. The intended readership includes researchers in academia, industry and other research institutes focusing on information science and technology.

This book exposes innovative technics for developing native macOS desktop applications by using C# and the .NET Core 3.1. You will discover that the implementation of a macOS native application can be done with other tools than the classical tools proposed by Apple: SwiftUI, Objective-C ... Before reserved to C++ programmers, the macOS application arena is now open to the C# developer's community. What you will learn in this book? - Essentials macOS commands (for rookie macOS user). - Essentials C# coding technics (for rookie C# developer). - Setup an efficient and professional development environment for .NET Core 3.1 on your Mac. - Review a panel of technical solutions for the GUI implementation. - Choose the adapted UI for your application specific needs. - Code your desktop application (boilerplates furnished). - Produce macOS executable from your C# project. - Package and distribute your application for the macOS ecosystem. Who is it for? - macOS C++, Java or Swift developers. - ASP.NET C# developers. - Windows C# developers. Accelerate your project start. This book includes project templates (boilerplates) useful for starting quickly and easily the coding of your macOS desktop application. This book avoids you a long and tedious phase of research for finding the most relevant technical solution for your app. Thus, you can focus on the functional features of the application rather than the technical constraints of the Mac OS X system.

A project-based guide to help you create, package, and deploy desktop applications on multiple platforms using modern JavaScript frameworks Key Features Use your web development skills with JavaScript and Node.js to build desktop applications for macOS and Windows Develop desktop versions of popular mobile applications that are similar to Slack, Spotify, and more Design desktop apps with automatic updates and real-time analytics capabilities Book Description The Electron framework allows you to use modern web technologies to build applications that share the same code across all operating systems and platforms. This also helps designers to easily transition from the web to the desktop. Electron Projects guides you through building cross-platform Electron apps with modern web technologies and JavaScript frameworks such as Angular, React.js, and Vue.js. You'll explore the process of configuring modern JavaScript frameworks and UI libraries, real-time analytics and automatic updates, and interactions with the operating system. You'll get hands-on with building a basic Electron app, before moving on to implement a Markdown Editor. In addition to this, you'll be able to experiment with major JavaScript frameworks such as Angular and Vue.js, discovering ways to integrate them with Electron apps for building cross-platform desktop apps. Later, you'll learn to build a screenshot snipping tool, a mini-game, and a music player, while also gaining insights into analytics, bug tracking, and licensing. You'll then get to grips with building a chat app, an eBook generator and finally a simple digital wallet app. By the end of this book, you'll have experience in building a variety of projects and project templates that will help you to apply your knowledge when creating your own cross-platform applications. What you will learn Initialize Node.js, Node Package Manager (NPM), and JavaScript to set up your app Integrate Phaser with Electron to build a simple 2D game Improve app quality by adding an error tracking system and crash reports Implement group chat features and event handling capabilities using Firebase Integrate a WordPress-like rich-text editor into your app Build Electron applications using a single codebase Who this book is for This book is for JavaScript developers who want to explore the Electron framework for building desktop apps. Working knowledge of modern frontend JavaScript frameworks and Node.js is assumed. No prior knowledge of desktop development is required.

[Copyright: 0e20cabb461886b8f4d54049db602847](https://www.pdfdrive.com/ebooks/0e20cabb461886b8f4d54049db602847)