

من أول كود لبناء تطبيق كامل FullStack BootCamp



تخيل أنك خلال 14 أسبوعًا – 160 ساعة تدريبية، تنتقل من الصفر لتصبح مطور ويب محترف جاهز لسوق العمل مخيم كودرز للبرمجة يمنحك **تجربة عملية** متكاملة: مشاريع حقيقية، تدريب مكثف مع مدربين خبرة أكثر من 16 عام، **شهادة معتمدة**، ودعم كامل حتى تبدأ مسارك الوظيفي أو عمك الحر

هنا، لا نتحدث عن دروس نظرية مملة، بل عن رحلة واقعية تصقل مهاراتك وتبني مستقبلك خطوة بخطوة

- من كتابة سيرتك الذاتية وإجتياز المقابلات
- إلى تصميم وبرمجة ونشر تطبيقات ويب احترافية
- وصولاً إلى هوية رقمية قوية وفرص عمل فعلية

رسالتنا

نحن نؤمن أن كل شخص يملك القدرة ليصبح مطورًا محترفًا، ونضع مهمتنا في تحويل الشغف إلى إنجازات نحن هنا لنمكن كل مهتم بالبرمجة من إطلاق كامل طاقاته عبر أدوات تعليمية متقدمة، تدريب عملي مكثف، وتجارب حقيقية تبني الثقة قبل المهارة. نعمل بروح الفريق ونشجع على التعاون وتبادل الخبرات لصناعة مجتمع تقني متماسك، قادر على مواجهة التغيرات التكنولوجية، والتفوق في كل بيئة عمل. غايتنا أن نضع جيلًا من المبرمجين الذين لا يواكبون المستقبل فحسب، بل يصنعونه بأنفسهم.

خليك معنا لتتعرف على المخيم:

المخيم هو برنامج تدريبي متكامل يعلّمك برمجة وتصميم مواقع الويب باستخدام React, و ASP.NET Core Mvc, مع إدارة قواعد البيانات وتطوير مهاراتك التقنية من الصفر للاحتراف. SQL , C# , API , JWT , GITHUB

الفئات المستهدفة

يستهدف المخيم فئات معينة من:

- الخريجين الجدد لطلاب تكنولوجيا المعلومات Fresh Graduate.
- طالب الجامعات بمختلف التخصصات (شو ماكان تخصصك).
- كل شخص يريد ان يدخل سوق البرمجة.
- موظفين شركات , مشرفين مختبرات الجامعات (شو ماكانت وظيفتك).
- اي شخص يريد ان يزيد دخله بشكل كبير.
- الراغبين في تغيير مسارهم المهني او تطوير مسارهم للوصول للاحتراف في عالم البرمجة.
- رواد الاعمال في تأسيس اعمالهم و مشروعهم.
- اي شركة حابة تعمل مسار للموظفين وتطويرهم.

مخرجات التدريب

- 1- تخريج طالب يمتلك خبرة عملية لا تقل عن 8 أشهر في مجال تطوير الويب، مع مهارات متقدمة في Problem Solving، والعمل ك Full Stack Developer، إضافة إلى خبرة في التعامل مع قواعد البيانات وتطوير واجهات المستخدم باستخدام React.
- 2- سوف يكون الطالب قادر على برمجة و تصميم اي موقع ويب كامل Front End - Back End ونشره على الإنترنت وحمايته وتسويقه وصيانتته وبيعه حسب القيمة السوقية، بتأهل معنا للتقنيات بشكل متقدم.
- 3- كل خريج من مخيم كودرز سوف يكون قادر على الالتحاق مع اي فريق برمجة في اي شركة وكأنه خبرة 8 شهور على الأقل.
- 4- كل طالب قادر على عمل تحليل لأي نظام وتصميم النظام وتطويره من الـ A - Z.
- 5- سوف تكون القدرة للطلاب للدخول في نقاشات بكل الـ Technology Stack بشكل نظري وعملي، و هذا اهم شيء في مقابلات العمل او شرح للعميل عن التكنولوجيا التي سوف تستخدم في بناء التطبيق الخاص به.



مميزات التدريب

سوف يتم اكتساب الخبرة على يد مدربين و مبرمجين محترفين ومعتمدين دوليا خبرات 16 عام

تقييم مستوى الطالب قبل وبعد التدريب من خلال امتحانات فعلية

تم تصميم البرنامج على مراحل حتى يتم نقل الطالب من المرحلة الابتدائية إلى مرحلة الاحتراف بالبرمجة ك Fullstack

سوف يتم تعليم الطالب تحت اشراف فريق كامل لضمان التعلم من الأخطاء

يوجد تقسيط لدينا و يكون على شكل دفعات ميسرة بدون كفيل ولابنوك.

شهادات معتمدة من كودرز باعتمادات عالمية من اكسفورد و طلال ابوغزاله و كامبردج و ICL

يجب ان يحضر الطالب 90% من ساعات التعليم والتدريب حتى يستطيع الحصول على الشهادة

محاضراتنا

وجاهي 

اونلاين 

مسجلة 

مراحل التدريب ضمن مخيم البرمجة

2 التعرف على البرمجة وطرق حل المشاكل وتحليلها
Problem Solving

4 تعلم تصميم صفحات الويب
HTML5-CSS3-JAVASCRIPT

6 بناء تطبيق الويب
ASP.NET CORE MVC Using C#

8 بناء الخدمات باستخدام
Restful Web API Services
Reposting Design Pattern

10 نشر الموقع على الويب
Web Hosting Azure + Any
Web Host Environment IIS

1 التعرف على مصطلحات
البرمجة وال IT

3 تعلم لغة C# بشكل مبتدأ و
متقدم
Basic to Advance

5 تعلم بناء قواعد البيانات
باستخدام Microsoft Sql Server

7 تعلم LINQ Query والتعامل
مع الـ Data Sources

9 تصميم الواجهات باستخدام
React

11 تطوير المسار المهني
(HR مقابلات & LinkedIn)

الاقوات



متابعة يومية من خلال قنوات تواصل فعالة ومتابعة حل
الواجبات اليومية والمشاريع الاسبوعية
GitHub - Zoom - Whatsapp

محاضرات اسبوعية حضوري + افتراضي مع مدرب محترف
Online - Offline - Recorded



14 اسبوع من التدريب المتواصل المكثف
بما يقارب 160 ساعة تدريبية

3 ايام في الاسبوع



4 ساعات يومياً

المرحلة الاولى: التعرف على مصطلحات البرمجة والـ IT

مصطلحات هامة وأساسية في عالم البرمجة مهمة لكل مبرمج هي دورة سريعة وهامة لكل مبرمج مبتدئ أو يعمل في مجال البرمجة

- أهم المصطلحات والتعريفات في عالم البرمجة.
- أساسيات ومصطلحات لا غنى عنها في عالم البرمجة.
- 45 مصطلح برمجي مهمين وأساسيين الى المبرمج.
- مصطلحات هامة وأساسية في عالم البرمجة.
- شرح معنى SQL ببساطة؟
- شرح الفرق بين Compiled Language و Interpreted Languages في البرمجة.

◀ الخطة للمرحلة الاولى

شرح عام للمصطلحات البرمجية

- مقدمة ممتازة تحفيزية
- مصطلحات هامة واساسية في عالم البرمجة مهمة لكل مبرمج
- 1- الخوارزمية Algorithm
- 2- برنامج Program
- 3- واجهة برمجة التطبيقات API
- 4- الوسيطة Argument
- 5- الكود القياسي الامريكي لتبادل المعلومات ASCII
- 6- التعبير المنطقي Boolean
- 7- الخطأ Bug
- 8- الحرف CHAR
- 9- الشيء Object
- 10- البرمجة الشيئية OOP
- 11- الفئة CLASS
- 12- الكود Code
- 13- واجهة سطر الاوامر (CLI)
- 14- التجميع Compliation
- 15- العبارات الشرطية Conditionals
- 16- الثوابت Constants
- 17- نوع البيانات Data Types
- 18- المصفوفة Array
- 19- اعلان Declaration

- 20- استثناء Exception
- 21- التعبير Expression
- 22- إطار العمل Framework
- 23- كود صلب Hardcode
- 24- الحلقة loop
- 25- حلقة لانهائية Endless loop
- 26- التكرار Iteration
- 27- الكلمات الاساسية Keywords
- 28- لاشيء Null
- 29- المعامل Operand
- 30- عامل التشغيل Operator
- 31- المتغير Variable
- 32- المؤشر Pointer
- 33- اللغة منخفضة المستوى Low-level programming language
- 34- اللغة عالية المستوى High-level programming language
- 35- لغة الآلة Mchine Learning
- 36- لغة التمييز Markup Language
- 37- الحزمة Package
- 38- وقت التشغيل Runtime
- 39- الخلفية Backend
- 40- الواجهة الامامية Front End
- 41- جانب الخادم Server Side
- 42- مصدر البيانات
- 43- البيان Statement
- 44- بيان الجملة Syntax
- 45- الرمز المميز Token

شرح تفصيلي لبعض المواضيع البرمجية

- شرح SQL ببساطة؟ شرح معنى SQL ببساطة.
- شرح الفرق بين Compiled Languages و Interpreted Languages في البرمجة.
- شرح مبادئ واساسيات الـ OOP بطريقة مبسطة بدون تعقيد.
- شرح الفرق بين High Level Languages و Low Level Languages في البرمجة.
- أفضل فريم وورك لعمل موقع وموبايل أبليكيشن في أسرع وقت وبدون مجهود.
- شرح API وما هي طرق استخدامه في البرمجة.

المرحلة الثانية الخوارزميات وحل المشكلات في البرمجة Problem Solving & Algorithm

مهارة البرمجة التنافسية (Programming Competitive) وحل المشكلات Problem Solving هي واحدة من أهم المهارات التي تميزك عن الآخرين في مجال صناعة البرمجيات إذا أتقنتها، وستضيف المزيد من القوة إلى سيرتك الذاتية، كما أن إتقانك لهذه المهارة سيعزز التفكير المنطقي لديك وسيحسن من طريقة تفكيرك ونظرتك للأمور بشكل عام. ومن المعروف أن العديد من عمالقة الشركات التكنولوجية المشهورين، بما في ذلك فيسبوك وجوجل وأمازون وما إلى ذلك يهتمون بشكل كبير بمهارات البرمجة التنافسية وحل المشكلات لدى المرشحين لديهم، لذا إذا كنت ترغب في الالتحاق بتلك الشركات فيجب عليك الاهتمام بتعلم هذه المهارة والعمل بجد لإتقانها وتعزيزها.

ما هي الخوارزمية؟

الخوارزمية Algorithm هي مجموعة من التعليمات المترتبة لحل مشكلة ما في الرياضيات أو أي مشكلة تواجهك في الحياة اليومية خلال زمن محدد وعدد خطوات محدود

◀ المواضيع الخاصة بالمرحلة الثانية

حل المشكلات Problem Solving

- ماهو الـ Problem Solving
- كيف تتم حل المشكلات؟
- استخدام الـ Problem Solving في البرمجة
- كيف يستفيد المبرمج من تعلمه للـ Problem Solving

الخوارزميات Algorithms

- ماهي الخوارزميات Algorithm
- ما هو الهدف من تعلم الخوارزميات؟
- طرق / خطوات كتابة الخوارزمية
- تعريف المشكلة Problem Defintion
- تصميم / كتابة الخوارزمية Writing an Algorithim
- تحويل الخوارزمية إلى برنامج للحاسوب Writing Code
- عوامل جودة الخوارزمية
- Flowchart ماهي المخططات التدفقية
- تطبيقات على الخوارزميات و المخططات التدفقية

المرحلة الثالثة: تعلم لغة برمجة ال C# بشكل مبتدأ ومتقدم

C# Programming Fundamental

The C# Programming Fundamentals is designed to equip learners with a solid foundation in the syntax, concepts, and principles of C# programming language.

The module covers various topics, starting with an introduction to the C# programming language, including its history and significance in modern software development.

It then progresses to cover essential aspects of programming in C# such as data types and variables, including integer, floating-point, Boolean, and string types. The part also covers the use of arithmetic, logical, and comparison operators in C#, as well as loops, and branching statements in C# to control program flow, as well as the basics of creating and calling methods and functions, including passing arguments and returning values

► What you will learn

- Introduction to C#.
- History of C# Version
- C# Code Execution
- Installing and Configuring C# in Visual Studio
- .NET Framework
 - What is .NET?
 - CLR – Common Language Runtime.
 - Standard Class Libraries
- Variable and Datatype (Call by value and call by reference)
- Operators and Operators Precedence.
- Increment and Decrement Operators.
- Conditional Statements.
 - If, else if, else.
 - Switch statements.
- Loops
 - For Loops in C#.
 - Foreach loops.
 - While loops.
 - Do-While loops.
 - Infinite Loops.
 - Jump Statements.

- Arrays
 - Arrays and Types of Arrays.
 - Manipulating Arrays.
- Lists.
- String
 - Strings and String Methods
 - Slicing strings into substrings & Formatting strings
- Predefined data types.
- Working with DateTime
 - DateTime class.
 - Timespan class.
 - Formatting dates and times.
- User-defined data types (Classes)
- OOP Object Oriented Programming
 - Object-Oriented Programming Concepts.
 - Classes and Objects.
 - Encapsulation and Abstraction.
 - Constructors.
 - Inheritance.
- Git & GitHub
 - How to use Git & GitHub to manage your code and upload projects

Advanced C# Programming

The Advanced C# Programming Fundamentals is designed to provide learners with an in-depth understanding of advanced C# programming concepts and techniques. The module covers various topics, starting with an overview of advanced programming concepts such as LINQ, delegates, and events. It then progresses to cover advanced object-oriented programming principles such as inheritance, polymorphism, and encapsulation, as well as advanced techniques for working with classes and objects, such as extension methods, interfaces, and abstract classes.

► What you will learn

- C# Namespace.
 - What is a namespace?
 - Creating a namespace.
 - C# System namespace.
- C# Generic
 - What is a generic?
 - Generic Type Parameters.
 - Constraints on Type Parameters.
 - Generic Classes and Interfaces.
 - Generic Methods.
- Polymorphism.
- Interfaces.
 - What is an interface?
 - Creating an interface.
 - Implementing an interface.
- Static Class and Static Methods.
- Extension Methods.
- C# Debugging, Exception Handling.
 - Debugging, Exception Handling.
 - What is debugging.
 - Types of Errors – Syntax, Runtime, and Logical.
 - Compile Time vs. Runtime Errors.
 - Null Reference Exception.

- LINQ C#
 - What is LINQ?
 - Write your first LINQ query!
 - Select and Select Many.
 - OfType and Where.
 - OrderBy, and Reverse.
 - GroupBy.
 - Distinct, Except.
 - Any, and Contains.
 - Skip and Take.
 - Inner Join.
 - First, Last, Single, and ElementAt
 - Aggregate and Count.
- Lambda Expression.
- What is Synchronous and Asynchronous?
- How to write good code.
 - What is good code?
 - Writing reusable code.
 - Writing clean code.
 - Writing efficient code.
 - Writing scalable code.

المرحلة الرابعة: تعلم تصميم صفحات الويب HTML5 CSS3 JavaScript

Fundamentals of HTML Basic HTML

► What you will learn

- Introduction to Web Development
- What is HTML5?
- Doctype & layout
- HTML Tag
- Formatting HTML Tag
- Forms and Input Tag
- IFrame Tag
- links and images
- Video Tag
- div , id & class
- List Tag
- Table Tag
- Semantic and Non- Semantic Tags

Fundamentals of CSS

► What you will learn

- What is CSS?
- CSS Syntax
- CSS Simple Selector
- CSS combinator selectors
- Pseudo-class selectors
- Pseudo Elements Selectors
- Attribute Selectors
- CSS Fonts property
- CSS Positions
- CSS Float.
- CSS Display.
- CSS Colors
- CSS Background Colors.
- CSS Style Table.
- CSS Styling Images.
-
-

- CSS Forms.
- CSS Text Effects.

Introduction to JavaScript

► What you will learn

- Introduction to JavaScript.
- Setting up your development environment.
- Writing your first JavaScript program.
- Understanding variables.
- Understanding data types.
- Using arithmetic and comparison operators.
- Arrays in JavaScript.
- Working with strings.
- Working with string manipulation.
- Conditional statements (if/else, switch).
- Looping structures (for, while).
- Functions and scope.
- Understanding the Document Object Model (DOM).
- Accessing DOM elements with JavaScript.
- Creating and removing DOM elements with JavaScript.
- Handling errors and exceptions

المرحلة الخامسة: بناء قواعد البيانات باستخدام Microsoft SQL Server

Course Overview

technical skills required to write basic Transact-SQL queries for Microsoft SQL Server. This course is the foundation for all SQL Server-related disciplines; namely, , Database Development.

Writing Queries Using Microsoft® SQL Server® 2021 Transact-SQL.

► What you will learn

- Write SELECT Queries.
- Query Multiple Tables.
- Sort and Filter Data.
- Describe the Use of Data Types in SQL Server.
- Modify Data Using Transact-SQL.
- Use Built In Functions.
- Group and Aggregate Data.
- Use Subqueries.
- Use Window Ranking,& Aggregate Functions.
- Implement Pivoting.
- Execute Stored Procedures.
- Program with T-SQL.
- Implement Error Handling.
- Implement Transactions.

Outlines

- Introduction to Microsoft SQL Server.
- Introduction to T-SQL Querying
- Data definition language
 - Create
 - Drop
 - Alter
- Writing SELECT Queries
 - Select Statement
 - Select (Where)

- Comparison Operators.
- AND OR Operator.
- NOT Condition.
- BETWEEN Condition.
- LIKE Condition.
- Select (SQL Functions -Single Row).
- Select (Multi Row).
- Select (Group By).
- Select (HAVING).
- Sorting and Filtering Data.
- Working with SQL Server 2014 Data Types.
- Using DML to Modify Data.
- Using Built-In Functions.
- Grouping and Aggregating Data.
- Querying data with Stored Procedures.

المرحلة السادسة: بناء تطبيق الويب ASP.NET CORE MVC

Course Overview

.NET Core is a standard-term release and focuses on being a more unified, simple, and fast development framework compared to .NET. This ASP.NET Core training course teaches attendees how to develop Web UIs and Web APIs with Microsoft's ASP.NET Core framework.

► Course Objectives

The overall objective of this course.

- Build mobile apps for multiple platforms with a single codebase.
- Understand the goals and benefits of ASP.NET Core compared to previous versions and other Web development frameworks.
- Learn to make successful decisions regarding application architecture and data access technology.
- Use ASP.NET's routing system to achieve a REST-style architecture.
- Learn how to build a compelling and maintainable HTML user interface using the Razor view engine and client-side JavaScript.
- Gain experience building a service that makes data available via a modern web API.
- Learn best practices for employing unit testing, logging, and error handling.

Prerequisites

- Previous experience developing web-based applications with C#.
- Some familiarity with HTML, CSS, and JavaScript.

ASP.NET Core C# MVC Web Development

Part 1: Introduction to ASP.NET Core MVC.

- Evolution of .NET and .NET Core.
- Asp.Net Web Form vs. Asp.Net MVC vs. Asp.Net Core.
- What is ASP.NET Core?.
- Advantages of ASP.NET Core.

Part 2: ASP.NET Core Application Architecture

- NuGet Packages
- Application Startup
- Hosting Environments

Part 3: Application Configuration

- Configuration Providers and Sources.

Part 4: ASP.NET Core First Application

- ASP.NET Core Environment Setup
- ASP .NET Core First Application
- Project Layout
- Understanding Life Cycle of ASP.Net Core Request

Part 5: Developing Controllers & Action Methods

- Controllers Overview
- Dependencies
- Action Methods and IActionResult object
- Passing data from Controller to View
- Action Filters
- Middleware

Part 6: Developing Views

- Responsibilities
- Conventions
- Razor Syntax
- Layouts
- ViewData and ViewBag
- Strongly-Typed Views
- Partial Views
- HTML and URL Helpers
- Tag Helpers
- View Components
- Razor Pages
- View Models
- Tag Helpers
- Model Binding

Part 7: Developing Model

- What is Entity Framework Core?
- What is ORM?
- Why Do We Need to Use An ORM?
- Why Entity Framework Core in .NET Applications?
- EF Core Development Approaches.
- EF Core Code First Approach.
- EF Core Database First Approach.
- EF Core Database Providers.
- Why We Need to Use Entity Framework Core.
- Creating MVC Core Models.
- Asynchronous Data Access.
- Working with Forms.
- Validate MVC Core Application.
- Differences Between EF Core and EF.

Part 8: Error Handling

- Best Practices.
- HTTP Error Status Codes.
- Developer Exception Page.

Part 9: Validations & Data Annotations.

- Overview Data Annotations and Validations.
- Validations with Data Annotation.
- Server Side and Client-Side Validation.

Part 10: Security

- Authentication.
- ASP.NET Identity.
- Authorization.

Part 11: Using Entity Framework Core in ASP.NET Core

- Basic CRUD Operations using Entity Framework.

Part 12: Using Layouts, CSS and JavaScript in ASP.NET Core

- Create Layout.
- View Start file.

Part 13: Module Development

- Understanding Areas
- Adding Areas
- Defining Area Routes
- Linking between Areas

Part 14: Routing – End-Points

- Create End Point.
- Create Route methods and Custom this method.

Part 15: Restful Web API Core C#.

HTTP

- Introduction to HTTP.
- HTTP Response.
- HTTP Response Headers.
- HTTP Request.
- HTTP Request Headers.
- Postman.
- HTTP Get vs Post.

Middleware

- Intro to Middleware.
- Middleware Chain.
- Custom Middleware Class.
- Custom Conventional Middleware Class.
- The Right Order of Middleware.

Part 16: Deployment

- Dotnet publish.

Real-time Project involving most of the above concepts Part 19 : Azure Cloud Service

- Overview about azure services.
- Create app service (Azure Portal + .NET).
- Create Azure Database.
- Publish App to Azure Server.

المرحلة السابعة: تعلم LINQ Query والتعامل Data Sources

Part 17: LINQ Basics

- Architecture of LINQ.
- Different Ways to Write LINQ Queries in C#.
- IEnumerable and IQueryable in C#.
- Differences between IEnumerable and IQueryable.
- LINQ Extension Methods in C#.
- LINQ Operators.
 - LINQ Select Operator in C#.
- LINQ Where Method in C#.
 - LINQ Distinct Method in C#.
 - LINQ Concat Method in C#.
- LINQ Aggregate Operators in C#.
- LINQ Any Method in C#.
- LINQ Contains Method in C#.
- LINQ Joins in C#.
- Element Operators in LINQ.

المرحلة الثامنة: بناء الخدمات باستخدام Restful Web API Services

Create New ASP.NET Core Web API

- Understand ASP.NET Core Web API and File Structure.
- Understand REST and HTTP Verbs.
- Routing In ASP.NET Core Web API.
- Running and Calling ASP.NET Core Web API.
- Controller & IActionResult.
 - Creating Controllers.
 - Multiple Action Methods.
 - ActionResult.
 - JsonResult.
 - IActionResult.
 - Status Code Results.
 - Redirect Results.
- Understanding Our Domain (Domain Driven Development).
- Create Domain Models.
- Adding Entity Framework Core Packages.
- Creating the DbContext Class.
- Adding ConnectionString To The Database In Appsettings.Json.
- Understanding Dependency Injection & Injecting DbContext Into Our Application.
- Run EF Core Migrations.

Authentication using JWT

Goal: Secure your API using token-based authentication.

Topics:

- What is JWT (JSON Web Token)?
- Create user registration and login endpoints
- Generate and validate JWT tokens
- Configure authentication middleware
- Use Postman to test JWT flow

المرحلة التاسعة: نشر الموقع على الويب + Web Hosting Azure Any Web Host Environment IIS

How to Start and Run a Web Hosting Business

Learn how to create a web hosting business or add web hosting to your existing services.

Outline

- Domain Names.
 - Introduction to domain names.
 - Domain Name registers.
 - Control Panel signin.
 - Domain Forwarding.
 - What is DNS and how does it work?
- Types of web hosting.
 - Webhosting Packages.
 - SharedHosting.
 - Dedicated Hosting.
 - VPS Hosting.
 - IIS Hosting.
 - Cloud Hosting.
- Publishing
 - How to Publish Your Website using FTP.
 - How to Edit Your Website using File Manager in cPanel.
 - VS Deploy.

Module 01: Core JavaScript & ES6

This module is crucial for building a strong foundation. Without a solid understanding of JavaScript, React will be much harder to grasp.

- **Deep Dive into JavaScript Fundamentals:** Go beyond the basics of variables and functions. Explore concepts like scope, hoisting, and the difference between var, let, and const.
- **Asynchronous JavaScript:** Cover promises and the async/await syntax in more detail, as this is essential for fetching data from APIs.
- **Modern JS Features:** Practice using array methods like map(), filter(), and reduce(), which are used extensively in React for data manipulation.

Module 02: Introduction to React Fundamentals

This is the heart of the course, where students learn to build UIs.

- **JSX and Components:** Focus on building a variety of components, from simple stateless components to more complex ones that manage their own state.
- **State and Props in Depth:** Explain the one-way data flow in React. Use analogies to make the concepts of props (data passed down from parent to child) and state (data managed within a component) easier to understand.
- **React Hooks Mastery:** Dedicate significant time to useState and useEffect. Show how useEffect can be

Module 03: API Integration and Frontend-Backend Communication

This module moves beyond static UIs to dynamic, data-driven applications.

- **RESTful APIs:** Explain what a REST API is and the relationship between HTTP methods (GET, POST, PUT, DELETE) and CRUD operations (Create, Read, Update, Delete).
- **Asynchronous API Calls:** Practice making API calls with both the native Fetch API and the popular Axios library. Teach students how to handle loading states, success responses, and error handling.
- **Building a CRUD Application:** Spend time building a full-featured application that demonstrates all four CRUD operations. This provides a practical, hands-on experience that solidifies the concepts.

Module 04: Advanced State Management and Routing

For larger applications, managing state across many components becomes a challenge. This module solves that problem.

- **Context API vs. Redux Toolkit:** Compare the React Context API with Redux Toolkit, explaining when to use each. The Context API is great for smaller-scale state management, while Redux is ideal for complex, global state.
- **React Router:** Introduce React Router to manage different pages and URLs in a single-page application. Cover creating routes, nesting routes, and navigating programmatically.

Module 05: Styling, Best Practices, and Deployment

This module brings the course to a close by focusing on presentation and the final steps of a project.

- **Styling in React:** Explore different styling methods, including CSS Modules, Styled Components, and using pre-built component libraries like Material-UI or Chakra UI.
- **Performance Optimization:** Introduce basic performance concepts like code splitting with `React.lazy()` and `Suspense`, and the importance of using keys in lists.
- **Project Deployment:** Conclude the course with a practical session on how to deploy a finished React application to a service like Vercel or Netlify, making their work live for the world to see.

Module 06: Advanced React & Performance Optimization

This module moves beyond the basics to cover more complex topics and best practices for building scalable applications.

- **Custom Hooks:** Dive deeper into creating custom hooks to abstract and reuse complex logic across multiple components.
- **Performance Optimization Techniques:**
 - Memoization with `React.memo` and `useCallback`: Explain how these hooks can prevent unnecessary re-renders of components, improving performance.
 - Lazy Loading and Code Splitting: Teach how to split the application's code into smaller chunks to improve the initial loading time.
- **Higher-Order Components (HOCs) and Render Props:** Introduce these patterns as alternative ways to share code and logic between components.

Module 07: Advanced React Patterns and Hooks

This module will help students write cleaner, more maintainable code by introducing them to advanced patterns and less common but powerful hooks.

- **Custom Hooks:** Advanced Examples: Go beyond simple data fetching. Show how to build complex custom hooks for tasks like form validation, managing local storage, or handling window resize events.
- **The useReducer Hook:** Introduce the useReducer hook as an alternative to useState for managing complex state logic. Explain when it's better to use useReducer (e.g., when state transitions depend on previous state or involve multiple sub-values).
- **Context with useReducer:** Combine the Context API with useReducer to create a lightweight, Redux-like state management solution without the full Redux toolkit. This is a powerful pattern for medium-sized applications.
- **Understanding useMemo and useCallback:** Provide detailed, practical examples of how useMemo and useCallback can be used to optimize component performance by preventing unnecessary re-renders of expensive calculations and functions.

المرحلة الحادية عشر: LinkedIn Career Preparation Module

As part of our commitment to ensure that students not only gain technical skills but also succeed in entering the job market, we propose LinkedIn Career Preparation module to be delivered at the end of each training camp (Mobile Development, Web Development, and others).

This module is designed to bridge the gap between technical knowledge and employability by equipping students with the tools to build strong professional identities, expand their networks, and actively pursue career opportunities. The outcome will be students graduating with not only coding skills but also a market-ready professional presence.

Module Outline

Session 1 – Building a Powerful LinkedIn Profile

- Importance of LinkedIn for developers in today's job market.
- Crafting an impactful headline and summary.
- Optimizing experience, education, and project sections.
- Adding bootcamp projects as case studies with GitHub/portfolio links.
- Selecting professional profile and banner images.

Each student completes a polished, recruiter-ready LinkedIn profile

Session 2 – Content & Visibility Strategy

- What to post as a junior developer (projects, learning journey, industry updates).
- Writing engaging posts and using hashtags/tags effectively.
- Engagement tactics: commenting, sharing, and starting conversations.
- Understanding LinkedIn's content visibility algorithm.

Students draft 2–3 posts ready for publishing to kickstart their visibility

Session 3 – Networking & Job Hunting

- Identifying and connecting with recruiters, HR managers, and industry professionals.
- Crafting personalized connection requests and follow-ups.
- Setting up job alerts and applying through LinkedIn Jobs.
- Using endorsements and recommendations to strengthen credibility.

Students establish a networking plan and set up targeted job alerts.

Session 4 – Advanced Career Hacks

- Building credibility through LinkedIn groups and discussions.
- Leveraging the “Featured” section for resumes, portfolios, and projects.
- Practicing mock interviews using LinkedIn’s tools and peer role-play.
- Exploring freelance and remote opportunities via LinkedIn.

Students create a personal “LinkedIn Market Entry Plan” integrating profile, content, networking, and job/freelance applications.

This LinkedIn Career Preparation module ensures our graduates are not only technically competent but also strategically positioned to secure opportunities in the job market. It enhances the overall value of our training camps by preparing students for real-world and market success.

Session 5 - Mock Interview Sessions

We will hold practice sessions to give participants an overview of how to approach job interviews. Attendees will learn the basics of presenting themselves, answering common questions, and handling interview pressure, with practical tips to improve readiness for the job market.