



CBP	IIB Council 區塊鏈專家認證課程 IIB Council The Certified Blockchain Professional Course 時數：40小時 費用：56,000元 點數：15點 教材：IIB-Council原廠英文電子教材	
適合對象	1. 商業技術主管 2. 政府技術幕僚 3. 全端工程師(Full Stack Developer)	
預備知識	1. 程式語言(Required) 2. 加密貨幣(Cryptocurrencies)基本認識(Preferred) 3. 比特幣(Bitcoin)或以太幣(Ether)使用經驗(Preferred)	
課程內容	1. 掌握區塊鏈技術的理論和技術基礎 2. 了解去中心化(decentralization)的概念，其影響以及與區塊鏈技術的關係 3. 了解區塊鏈開發和實現的基本技術原則 4. 了解比特幣和其他加密貨幣背後的區塊鏈技術運作機制 5. 透過實際範例了解加密技術如何用於數據保護 6. 了解智能合約(Smart Contracts)的理論基礎，以及如何使用Solidity和相關開發框架，在以太坊(Ethereum)區塊鏈上開發去中心化的應用程式(DApp) 7. 研究替代區塊鏈解決方案，包括超級帳本(Hyperledger)、Corda等 8. 了解各行各業中區塊鏈應用的使用案例 9. 探索區塊鏈實現和商業化策略 10. 了解當前業務架構中區塊鏈整合的最佳實務 11. 確定可能的區塊鏈實現對法律方面的影響 12. 預測應用策略採用可能帶來的風險和失敗，並學習採用有效的變更管理和專案管理方法加以避免 13. 探索區塊鏈技術的新研究課題和未來應用範圍	
備註事項	1. 報名本課程贈送一次認證考試 2. 課程結束後，出席率達80%以上者將獲頒發原廠結訓證書 3. 開課日起半年內，同一版本課程可免費重聽一次 4. 白天班之上課時間為09:00~18:00 5. 課程優惠方案： 早鳥優惠價：開課前2周完成報名繳費，享有早鳥優惠價NT\$53,200元	
後續推薦課程	BCIC：區塊鏈與智慧合約實戰演練	

BCIC	區塊鏈與智慧合約實戰演練 Blockchain and Smart Contract in Action 時數：35小時 費用：24,000元 點數：6點 教材：恆逸專用教材	
適合對象	1. 想了解區塊鏈與智慧合約的工程師與專案經理 2. 想了解智慧合約的實作者	
預備知識	1. 任一程式語言的基礎 2. 基本網路知識	
課程內容	1. Blockchain 歷史與簡介 <ul style="list-style-type: none"> • 加解密演算法 • Hash Tree/Merkle Tree 雜湊樹 • 非對稱加解密 2. Blockchain 生態系 <ul style="list-style-type: none"> • Ethereum V.S. Hyperledger 3. 智慧合約與Solidity簡介 <ul style="list-style-type: none"> • Solidity與Solidity相關檔案 • Pragma、註解與Mport • 合約本體 • state變數、struct與修飾子 • 事件、列舉與函數 • 資料型態 • 參照型態 • 陣列與對應 4. Solidity語法 <ul style="list-style-type: none"> • 函數與修飾子 • View、常數與Pure Function • Address Function • Fallback Function • 例外處理、事件和Summary 5. Ethereum <ul style="list-style-type: none"> • EVM(Ethereum Virtual Machine) • Ethereum(Smart Contract) • Truffle框架的設定與使用 6. 合約的偵錯 7. 在雲端的部署	
後續推薦課程	BCFS：區塊鏈與DApp去中心化應用程式Full Stack全棧開發實戰演練	

BCFS

區塊鏈與DApp去中心化應用程式Full Stack全棧開發實戰演練

Full Stack DApps Development Using Block Chain in Action

時數：35小時 | 費用：24,000元 | 點數：6點 | 教材：恆逸專用教材



適合對象	<ol style="list-style-type: none"> 1. 想要開發區塊鏈應用程式的工程師 2. 想學習Ethereum區塊鏈實作的工程師 3. 想要了解區塊鏈的軟體架構師 		
先修課程	<ol style="list-style-type: none"> 1. CBP：IIB Council區塊鏈專家認證課程 2. BCIC：區塊鏈與智慧合約實戰演練 		
預備知識	<ol style="list-style-type: none"> 1. 基礎網路知識 2. 對JavaScript基礎認識 3. 了解Node.JS、React為佳 		
課程內容	<table border="0"> <tr> <td style="vertical-align: top;"> <ol style="list-style-type: none"> 1. Ethereum網路架構 <ul style="list-style-type: none"> • Blockchain的基本架構 • Blockchain的演化 • Ether、GAS和EVM 2. 使用wallet錢包和區塊鏈網路溝通 <ul style="list-style-type: none"> • 網頁錢包的安裝 • 連線與使用 3. Geth的使用與環境安裝和架設 <ul style="list-style-type: none"> • 設置Geth • Geth_CLI的操作 • Geth的JS Api 4. web3.js <ul style="list-style-type: none"> • 帳戶列表和餘額API使用 • 交易與合約API呼叫 • 交易物件傳送 • 事件與日誌 • 過濾監控API </td> <td style="vertical-align: top;"> <ol style="list-style-type: none"> 5. Truffle Framework與Solidity智慧合約 6. 智慧合約設計樣式 7. 前端分散式應用程式的考量 8. ERC20自訂代幣 9. Truffle與React/Redux <ul style="list-style-type: none"> • 建立前端應用程式專案 • 部署合約 • 使用React進行前端開發 </td> </tr> </table>	<ol style="list-style-type: none"> 1. Ethereum網路架構 <ul style="list-style-type: none"> • Blockchain的基本架構 • Blockchain的演化 • Ether、GAS和EVM 2. 使用wallet錢包和區塊鏈網路溝通 <ul style="list-style-type: none"> • 網頁錢包的安裝 • 連線與使用 3. Geth的使用與環境安裝和架設 <ul style="list-style-type: none"> • 設置Geth • Geth_CLI的操作 • Geth的JS Api 4. web3.js <ul style="list-style-type: none"> • 帳戶列表和餘額API使用 • 交易與合約API呼叫 • 交易物件傳送 • 事件與日誌 • 過濾監控API 	<ol style="list-style-type: none"> 5. Truffle Framework與Solidity智慧合約 6. 智慧合約設計樣式 7. 前端分散式應用程式的考量 8. ERC20自訂代幣 9. Truffle與React/Redux <ul style="list-style-type: none"> • 建立前端應用程式專案 • 部署合約 • 使用React進行前端開發
<ol style="list-style-type: none"> 1. Ethereum網路架構 <ul style="list-style-type: none"> • Blockchain的基本架構 • Blockchain的演化 • Ether、GAS和EVM 2. 使用wallet錢包和區塊鏈網路溝通 <ul style="list-style-type: none"> • 網頁錢包的安裝 • 連線與使用 3. Geth的使用與環境安裝和架設 <ul style="list-style-type: none"> • 設置Geth • Geth_CLI的操作 • Geth的JS Api 4. web3.js <ul style="list-style-type: none"> • 帳戶列表和餘額API使用 • 交易與合約API呼叫 • 交易物件傳送 • 事件與日誌 • 過濾監控API 	<ol style="list-style-type: none"> 5. Truffle Framework與Solidity智慧合約 6. 智慧合約設計樣式 7. 前端分散式應用程式的考量 8. ERC20自訂代幣 9. Truffle與React/Redux <ul style="list-style-type: none"> • 建立前端應用程式專案 • 部署合約 • 使用React進行前端開發 		