



ไอทีจีเนียส เอ็นจิเนียริ่ง

IT GENIUS ENGINEERING CO., LTD.

COURSE ID

DOP-01

COURSE OUTLINE · รายละเอียดหลักสูตรอบรม

DEVOPS / TERRAFORM

Infrastructure as Code ด้วย Terraform

• 18 ชั่วโมง

• 3 วัน

• ควรมีพื้นฐาน Cloud

• ระดับ Intermediate



ระยะเวลา: 3 วัน

18 ชั่วโมง (3 วัน วันละ 6 ชม.) · 09:00-16:00 น.



รูปแบบ: Classroom / Hybrid

Onsite หรือ Online ผ่าน Microsoft Teams · Public / In-house



รายละเอียดหลักสูตร

www.itgenius.co.th

การสร้างและแก้ไขโครงสร้างพื้นฐานด้วยมือผ่านหน้าเว็บคอนโซลทำให้เกิดความไม่สม่ำเสมอ ตรวจสอบย้อนหลังไม่ได้ และสร้างซ้ำในอีกสภาพแวดล้อมได้ยาก Terraform แก้ปัญหานี้ด้วยการเขียนโครงสร้างพื้นฐานเป็นโค้ด (Infrastructure as Code) ที่เก็บใน Git ตรวจสอบผ่าน Pull Request และนำไปสร้างซ้ำได้เหมือนกันทุกครั้ง อีกทั้งยังใช้ได้กับหลายผู้ให้บริการคลาวด์ด้วยเครื่องมือเดียว

หลักสูตรนี้พาผู้เรียนเขียน Terraform ตั้งแต่ศูนย์จนถึงระดับใช้งานในทีม ครอบคลุมภาษา HCL, Provider และ Resource, การจัดการ State อย่างปลอดภัย, ตัวแปรและ Output, การแยกเป็น Module ที่นำกลับมาใช้ซ้ำได้, การแยกสภาพแวดล้อม Dev และ Prod, Remote State สำหรับทำงานเป็นทีม, การจัดการทรัพยากรที่สร้างไว้แล้วด้วย import, การจัดการ Secret อย่างปลอดภัย ไปจนถึงการวาง CI/CD ให้รับ plan และ apply อัตโนมัติ ทุกหัวข้อลงมือทำจริงบนคลาวด์ (อบรม 3 วัน วันละ 6 ชั่วโมง รวม 18 ชั่วโมง ระดับ Intermediate)

เครื่องมือ

Bash Shell

systemd

OpenSSH

Networking

APT / DNF

ราคาค่าอบรม · PUBLIC / IN-HOUSE

10,500 บาท / ท่าน

ราคายังไม่รวมภาษีมูลค่าเพิ่ม (VAT 7%) · รวมเอกสาร Lab Guide โค้ดตัวอย่าง อาหารกลางวันและอาหารว่าง และ Certificate · ใช้อุปกรณ์ส่วนตัวทดสอบในคลาส · รับจัด In-house ขอใบเสนอราคาได้



เพิ่มเพื่อน
LINE

02-570-8449

088-807-9770

@itgenius

www.itgenius.co.th

01 วัตถุประสงค์ Objectives

- เข้าใจแนวคิด Infrastructure as Code และประโยชน์เกี่ยวกับการสร้างทรัพยากรด้วยมือ
- ติดตั้งและใช้งาน Terraform CLI พร้อมวงจรคำสั่งหลัก init, plan, apply และ destroy
- เขียนภาษา HCL ได้อย่างถูกต้องทั้ง Resource, Data Source, Variable, Output และ Expression
- เข้าใจการทำงานของ State และจัดการอย่างปลอดภัยรวมถึงการล็อกไม่ให้ชนกัน
- แยกโค้ดเป็น Module ที่นำกลับมาใช้ซ้ำได้และจัดโครงสร้างโปรเจกต์อย่างเป็นระบบ
- แยกสภาพแวดล้อม Dev, Staging และ Production ด้วย Workspace หรือโครงสร้างไดเรกทอรี
- ตั้งค่า Remote State และ State Locking สำหรับการทำงานเป็นทีม
- นำทรัพยากรที่สร้างไว้แล้วเข้ามาอยู่ในการจัดการของ Terraform ด้วย import
- จัดการ Secret อย่างปลอดภัยและวาง CI/CD ให้รับ Terraform อัตโนมัติ

02 กลุ่มเป้าหมาย Target Audience

- DevOps Engineer และ Cloud Engineer ที่ต้องดูแลโครงสร้างพื้นฐานบนคลาวด์
- ผู้ดูแลระบบ (System Administrator) ที่ต้องการเปลี่ยนจากงานมือเป็นงานอัตโนมัติ
- นักพัฒนาที่ต้องสร้างและดูแลสภาพแวดล้อมของระบบด้วยตนเอง
- สถาปนิกระบบที่ต้องออกแบบโครงสร้างพื้นฐานให้สร้างซ้ำได้และตรวจสอบได้
- ทีมที่ใช้หลายคลาวด์และต้องการเครื่องมือมาตรฐานเดียวในการจัดการ

03 จุดเด่นของหลักสูตร Highlights

- **ลงมือเขียนจริงทุกหัวข้อ** สร้างทรัพยากรบนคลาวด์ด้วยโค้ดจริง
- **ครอบคลุมงานทีม** Remote State, Locking และ Pull Request Workflow
- **Module ใช้ซ้ำได้** จัดโครงสร้างโปรเจกต์แบบมืออาชีพ
- **จัดการของเดิมที่มีอยู่** import ทรัพยากรที่สร้างด้วยมือเข้าระบบ
- **ต่อยอด CI/CD** รับ plan และ apply อัตโนมัติอย่างปลอดภัย

04 ความรู้พื้นฐาน Prerequisites

- เข้าใจแนวคิดคลาวด์พื้นฐาน (แนะนำผ่าน AWS Cloud Practitioner หรือ Azure AZ-900 มาก่อน)
- ใช้งาน Command Line และ Git ในระดับพื้นฐานได้
- เข้าใจแนวคิดเครือข่ายพื้นฐาน เช่น IP, Subnet และ Security Group
- ไม่จำเป็นต้องเขียนโปรแกรมเป็น แต่ควรอ่านไฟล์ตั้งค่าแบบมีโครงสร้างได้

05 สิ่งที่ต้องเตรียมและเครื่องมือ Setup & Tools

รายการ	รายละเอียด
อุปกรณ์	Notebook (Windows หรือ macOS) เชื่อมต่ออินเทอร์เน็ตได้
เครื่องมือหลัก	Terraform CLI, Visual Studio Code พร้อมส่วนขยาย HashiCorp Terraform
บัญชีคลาวด์	บัญชีทดสอบ AWS หรือ Azure สำหรับสร้างทรัพยากรจริงในคลาส
ระบบจัดเก็บโค้ด	Git และ GitHub หรือ GitLab สำหรับฝึก Workflow แบบทีม
เอกสารอ้างอิง	Terraform Registry, เอกสาร Provider และโค้ดตัวอย่างของหลักสูตร

หมายเหตุ: ผู้เรียนจะสร้างทรัพยากรจริงบนบัญชีคลาวด์ทดสอบระหว่างอบรม ควรใช้บัญชีสำหรับทดสอบเท่านั้นและรัน terraform destroy เมื่อจบแต่ละ Lab เพื่อไม่ให้เกิดค่าใช้จ่ายค้าง สถาบันจะแนะนำการตั้งค่างบประมาณและการแจ้งเตือนค่าใช้จ่ายก่อนเริ่ม Lab และห้ามนำ Credential ของระบบ Production จริงมาใช้ในคลาส

06 ภาพรวมการเรียนรู้ 3 วัน Big Picture

วัน	หัวข้อหลัก
Day 1 Section 1-4	แนวคิด IaC และการติดตั้ง, ภาษา HCL, Provider และ Resource, การจัดการ State
Day 2 Section 5-8	ตัวแปรและ Output, Module, Workspace และหลายสภาพแวดล้อม, Remote State และการทำงานเป็นทีม
Day 3 Section 9-12	การจัดการทรัพยากรที่มีอยู่, ความปลอดภัยและ Secret, CI/CD สำหรับ Terraform และ Capstone



DAY
1

พื้นฐาน Terraform และ State

เนื้อหาการอบรม · Section 1-4

รวม

6 ชั่วโมง

1 แนวคิด Infrastructure as Code และการติดตั้ง

- ปัญหาของการสร้างทรัพยากรด้วยมือ และประโยชน์ของ Infrastructure as Code
- Terraform ทำงานอย่างไร: Declarative, Provider, Plan และ Apply และเทียบกับเครื่องมืออื่น
- การติดตั้ง Terraform CLI และการตั้งค่าการเข้าถึงบัญชีคลาวด์อย่างปลอดภัย
- Lab: ติดตั้ง Terraform และสร้างทรัพยากรแรกบนคลาวด์ด้วยโค้ด

2 ภาษา HCL และโครงสร้างโค้ด

- โครงสร้างไฟล์ .tf และบล็อกหลัก: terraform, provider, resource และ data
- ชนิดข้อมูล นิพจน์ (Expression) เชื้อไข และฟังก์ชันที่ใช้บ่อย
- การอ้างอิงระหว่างทรัพยากรและการกำหนดลำดับการสร้าง (Dependency)
- Lab: เขียนโค้ดสร้างเครือข่ายและเครื่องเซิร์ฟเวอร์ที่อ้างอิงกัน

3 Provider และการจัดการทรัพยากร

- การใช้งาน Provider และการล็อกเวอร์ชันด้วย Version Constraint
- วงจรชีวิตทรัพยากร: การสร้าง แก้ไข แทนที่ และลบ และการอ่านผลลัพธ์ของ plan
- การใช้ Data Source เพื่ออ้างอิงทรัพยากรที่มีอยู่แล้ว
- Lab: แก้ไขโครงสร้างพื้นฐานและอ่าน plan ให้เข้าใจก่อน apply ทุกครั้ง

4 การจัดการ State

- State คืออะไร ทำไมสำคัญ และความเสี่ยงเมื่อ State สูญหายหรือถูกแก้ไขผิด
- คำสั่งจัดการ State: show, list, mv และ rm และเมื่อใดควรใช้
- การป้องกันข้อมูลลับใน State และการไม่นำ State ขึ้น Git
- Lab: สืบค้นไฟล์ State และฝึกลบทรัพยากรภายใน State



DAY
2

ตัวแปร Module และการทำงานเป็นทีม

เนื้อหาการอบรม · Section 5-8

รวม

6 ชั่วโมง

5 ตัวแปร Output และการตั้งค่า

- การประกาศตัวแปร (Variable) ชนิดข้อมูล ค่าเริ่มต้น และการตรวจสอบค่า (Validation)
- การส่งค่าผ่าน tfvars, ตัวแปรสภาพแวดล้อม และ Command Line
- การใช้ Output ส่งค่าออกและการใช้ Local Value ลดการเขียนซ้ำ
- Lab: แปลงโค้ดที่ Hardcode ให้ยืดหยุ่นด้วยตัวแปรและ Output

6 Module ที่นำกลับมาใช้ซ้ำได้

- แนวคิด Module และการออกแบบให้ใช้ซ้ำได้ในหลายโครงการ
- การเขียน Module ของตนเองและการเรียกใช้ Module จาก Terraform Registry
- การจัดโครงสร้างโปรเจกต์ให้อ่านง่ายและดูแลระยะยาวได้
- Lab: แยกโค้ดเป็น Module และเรียกใช้ซ้ำในสองสภาพแวดล้อม

7 หลายสภาพแวดล้อม (Dev, Staging, Prod)

- การแยกสภาพแวดล้อมด้วย Workspace เกี่ยวกับการแยกด้วยโครงสร้างไดเรกทอรี
- การจัดการค่าตั้งต้นที่ต่างกันในแต่ละสภาพแวดล้อมอย่างปลอดภัย
- แนวปฏิบัติการตั้งชื่อและติด Tag ทรัพยากรให้ตรวจสอบต้นกุ่มได้
- Lab: สร้างสภาพแวดล้อม Dev และ Prod จากโค้ดชุดเดียวกัน

8 Remote State และการทำงานเป็นทีม

- การเก็บ State ไว้ส่วนกลาง (Remote Backend) และเหตุผลที่ทีมต้องใช้
- State Locking เพื่อป้องกันการแก้ไขพร้อมกันจนข้อมูลเสียหาย
- Workflow แบบทีม: Branch, Pull Request, การรีวิวผล plan ก่อน apply
- Lab: ย้าย State ขึ้น Remote Backend และทดสอบการล็อกเมื่อมีคนรันพร้อมกัน



DAY
3

ของเดิม ความปลอดภัย และ CI/CD

เนื้อหาการอบรม · Section 9-12

รวม

6 ชั่วโมง

9 การนำทรัพยากรที่มีอยู่เข้าระบบ

- การใช้ terraform import นำทรัพยากรที่สร้างด้วยมือเข้ามาอยู่ในการจัดการ
- การจัดการเมื่อสถานะจริงกับโค้ดไม่ตรงกัน (Configuration Drift)
- แนวทางค่อย ๆ เปลี่ยนระบบเดิมให้มาอยู่ใต้ Terraform โดยไม่กระทบบริการ
- Lab: import ทรัพยากรที่สร้างด้วยมือและปรับโค้ดให้ตรงกับของจริง

10 ความปลอดภัยและการจัดการ Secret

- การจัดการข้อมูลลับ: ไม่เก็บรหัสผ่านและคีย์ไว้ในโค้ดหรือใน Git
- การใช้บริการจัดการความลับของคลาวด์ร่วมกับ Terraform
- การกำหนดสิทธิ์ให้บัญชีที่รัน Terraform ตามหลัก Least Privilege
- การตรวจสอบโค้ดโครงสร้างพื้นฐานเพื่อหาการตั้งค่าที่ไม่ปลอดภัย

11 CI/CD สำหรับ Terraform

- การวาง Pipeline ให้รัน fmt, validate และ plan อัตโนมัติเมื่อเปิด Pull Request
- การควบคุมขั้นตอน apply ให้ต้องมีการอนุมัติก่อนแตะ Production
- การเก็บผล plan เป็นหลักฐานและการย้อนกลับเมื่อเกิดปัญหา
- Lab: สร้าง Pipeline ตรวจสอบและรัน plan อัตโนมัติบน Git Repository

12 Capstone และแนวปฏิบัติที่ดี

- โจทย์: ออกแบบและสร้างโครงสร้างพื้นฐานครบชุดด้วย Terraform ตั้งแต่เครือข่ายถึงแอปพลิเคชัน
- แนวปฏิบัติที่ดีในการเขียนและดูแลโค้ดโครงสร้างพื้นฐานระยะยาว
- การต่อยอดสู่ Ansible, Kubernetes และเครื่องมือ DevOps อื่นในสายงาน
- Workshop: นำเสนอโครงสร้างที่สร้างขึ้นและทบทวนโค้ดร่วมกัน



07 เรียนจบแล้วได้อะไร Outcomes

- เขียน Terraform ได้จริง สร้างและแก้ไขโครงสร้างพื้นฐานด้วยโค้ด
- จัดการ State อย่างปลอดภัย รวมถึง Remote State และการล็อก
- ออกแบบ Module ใช้ซ้ำได้ โครงสร้างโปรเจกต์แบบมืออาชีพ
- แยกสภาพแวดล้อมได้ Dev, Staging, Prod จากโค้ดชุดเดียว
- นำของเดิมเข้าระบบ import ทรัพยากรที่สร้างด้วยมือ
- วาง CI/CD ได้ plan อัตโนมัติและ apply แบบอัตโนมัติ

08 สิ่งที่คุณเรียนจะได้รับ Deliverables

- เอกสารประกอบการอบรมฉบับสมบูรณ์ และ Lab Guide การเขียน Terraform ทีละขั้นตอน
- โค้ดตัวอย่างครบทุก Lab พร้อม Module ที่นำกลับไปใช้ซ้ำได้
- Checklist แนวปฏิบัติที่ดีด้านโครงสร้างโปรเจกต์ ความปลอดภัย และการทำงานเป็นทีม
- ประกาศนียบัตรจบหลักสูตร (Certificate of Completion) และช่องทางสอบถามวิทยากรหลังจบหลักสูตร

09 ข้อมูลผู้สอน Instructor

อ.สามิตร ไทยม ผู้สอนและผู้เชี่ยวชาญด้านโครงสร้างพื้นฐานและ DevOps ที่มีประสบการณ์กว่า 15 ปี วางระบบอัตโนมัติและโครงสร้างพื้นฐานบนคลาวด์ให้องค์กรจำนวนมาก จุดเด่นคือการอธิบายให้เข้าใจง่าย เชื่อมโยงกับงานจริง และพาผู้เรียนเขียนโค้ดจนสร้างโครงสร้างพื้นฐานได้ด้วยตัวเอง

ราคาค่าอบรม 10,500 บาท/ท่าน · สนใจ IN-HOUSE ขอใบเสนอราคา

10,500 บาท / ท่าน (ยังไม่รวม VAT 7%)

☎ 02-570-8449

LINE @itgenius

☎ 088-807-9770

✉ contact@itgenius.co.th



เพิ่มเพื่อน
LINE