สถาบันไอทีจีเนียส
IT GENIUS INSTITUTE

COURSE ID
OTC-08

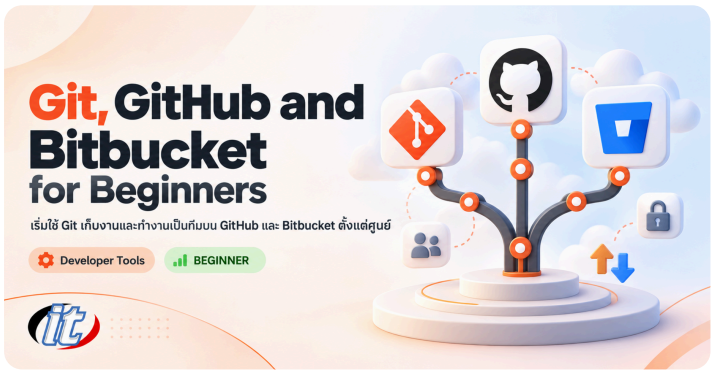
DEVOPS / VERSION CONTROL

COURSE OUTLINE · รายละเอียดหลักสูตรอบรม

Git, GitHub and Bitbucket for Beginners

เริ่มใช้ Git เก็บงานและทำงานเป็นทีมบน GitHub และ Bitbucket ตั้งแต่ศูนย์

- 12 ชั่วโมง
- 2 วัน
- พื้นฐาน ใช้ Windows หรือ macOS ได้
- ระดับ Beginner



ระยะเวลา: 2 วัน
12 ชั่วโมง (2 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

งานพัฒนาซอฟต์แวร์ เว็บไซต์ สคริปต์ดูแลระบบ หรือแม้แต่เอกสารที่แก้ร่วมกันหลายคน ล้วนต้องการวิธีเก็บประวัติการแก้ไข ย้อนกลับไปเวอร์ชันก่อน และรวมงานของหลายคนเข้าด้วยกันโดยไม่ทับกัน Git คือเครื่องมือ Version Control ที่เป็นมาตรฐานของวงการ ส่วน GitHub และ Bitbucket คือบริการที่ทีมใช้เก็บงานและรีวิวงานร่วมกัน หากใช้ไม่ถูกวิธีก็มักเจอปัญหางานหาย Conflict แก้ไม่เป็น หรือรหัสผ่านและข้อมูลลับหลุดขึ้น Repository

หลักสูตรนี้พาผู้เริ่มต้นใช้ Git รุ่นปัจจุบันตั้งแต่ศูนย์ ติดตั้งและตั้งค่าบน Windows หรือ macOS ใช้ VS Code เป็นทั้ง Editor และเครื่องมือ Diff และ Merge บันทึกงานด้วย add และ commit ดูประวัติ ย้อนงานอย่างปลอดภัยด้วย restore และ revert แตก Branch และแก้ Conflict จากนั้นเชื่อมต่อ GitHub ด้วย HTTPS และ SSH ทำงานเป็นทีมผ่าน Pull Request และใช้ Bitbucket Cloud ด้วย API Token ที่มาแทน App Password ก่อนปิดท้ายด้วย Workshop โปรเจกต์ทีมตั้งแต่สร้าง Repository จนรวมงานเสร็จ (อบรม 2 วัน วันละ 6 ชั่วโมง รวม 12 ชั่วโมง ระดับ Beginner)

เครื่องมือ
- Git 2.5x
- GitHub
- Bitbucket Cloud
- VS Code
- Pull Request

ราคาค่าอบรม · PUBLIC / IN-HOUSE

6,500 บาท / ท่าน

สมัครผ่านเว็บไซต์ ลด 10%

ราคายังไม่รวมภาษีมูลค่าเพิ่ม (VAT 7%) · รวมเอกสารประกอบการอบรม Lab Guide, Git Cheat Sheet, Repository ตัวอย่างสำหรับฝึก และ Certificate · รับจัด In-house ขอใบเสนอราคาได้

เพิ่มเพื่อน LINE

02-570-8449 · 088-807-9770 · @itgenius · www.itgenius.co.th

01 วัตถุประสงค์ Objectives

- เข้าใจแนวคิด Version Control และความแตกต่างของ Git, GitHub และ Bitbucket
- ติดตั้งและตั้งค่า Git พร้อมใช้ VS Code เป็น Editor และเครื่องมือ Diff และ Merge ได้
- บันทึก ดูประวัติ เปรียบเทียบ และย้อนงานด้วยคำสั่ง Git รุ่นปัจจุบันได้อย่างปลอดภัย
- แตก Branch รวมงาน และแก้ Merge Conflict ได้เอง
- เชื่อมต่อ GitHub และ Bitbucket ด้วย HTTPS และ SSH แล้ว push และ pull งานร่วมกับทีมได้
- ทำงานเป็นทีมด้วย Pull Request และ Code Review ได้

02 กลุ่มเป้าหมาย Target Audience

- นักเรียน นักศึกษา และผู้เริ่มต้นเขียนโปรแกรมที่ต้องการเก็บงานอย่างเป็นระบบ
- ครู อาจารย์ และวิทยากรที่ต้องใช้ GitHub ในการสอนหรือส่งงาน
- นักไอทีและผู้ดูแลระบบที่ต้องเก็บสคริปต์และไฟล์ตั้งค่าในระบบ Version Control
- ทีมงานที่เริ่มใช้ GitHub หรือ Bitbucket ร่วมกันเป็นครั้งแรก

03 จุดเด่นของหลักสูตร Highlights

- **เริ่มจากศูนย์** ไม่ต้องเคยใช้ Git หรือ Command Line มาก่อน
- **คำสั่งรุ่นปัจจุบัน** switch, restore และ Branch หลักชื่อ main
- **เครื่องมือเดียวจบ** VS Code เป็นทั้ง Editor, Diff และ Merge
- **ยืนยันตัวตนแบบปัจจุบัน** SSH Key, Token และ Credential Manager
- **ทำงานเป็นทีมจริง** Pull Request บน GitHub และ Bitbucket

04 ความรู้พื้นฐาน Prerequisites

- ใช้งานคอมพิวเตอร์และระบบปฏิบัติการ Windows หรือ macOS ได้คล่อง
- ติดตั้งโปรแกรมและจัดการไฟล์และโฟลเดอร์ได้
- ค้นหาข้อมูลบนอินเทอร์เน็ตได้ ไม่จำเป็นต้องเคยเขียนโปรแกรมมาก่อน
- มีอีเมลสำหรับสมัครบัญชี GitHub และ Atlassian (Bitbucket) แบบฟรี

05 สิ่งที่ต้องเตรียมและเครื่องมือ Setup & Tools

รายการ	รายละเอียด
อุปกรณ์	Notebook Windows หรือ macOS เชื่อมต่ออินเทอร์เน็ต พร้อมสิทธิ์ติดตั้งโปรแกรม
Git	Git รุ่นล่าสุด (Git for Windows พร้อม Git Bash หรือ Git บน macOS)
Editor	VS Code พร้อม Source Control, Diff และ Merge Editor ในตัว
บริการออนไลน์	บัญชี GitHub และ Bitbucket Cloud แบบฟรีของผู้เรียน และ GitHub CLI
ไฟล์ฝึก	Repository ตัวอย่าง โจทย์ Conflict และ Git Cheat Sheet

หมายเหตุ: ปรับปรุงใหม่ปี 2026 จากหลักสูตรเดิม: ใช้ Git รุ่นปัจจุบันกับคำสั่ง git switch และ git restore แทนการใช้ checkout ทุกงาน ตั้ง Branch หลักเป็น main ซึ่ง Git 3.0 จะใช้เป็นค่าเริ่มต้น เปลี่ยนจาก Notepad++, TextMate 2 และ P4Merge มาใช้ VS Code เป็นทั้ง Editor และเครื่องมือ Diff และ Merge เปลี่ยนการยืนยันตัวตนจากรหัสผ่านซึ่ง GitHub เลิกรองรับแล้ว มาเป็น Git Credential Manager, Personal Access Token และ SSH Key แบบ Ed25519 และใช้ API Token กับ Bitbucket Cloud แทน App Password ที่ Atlassian ปิดการใช้งานแล้ว เพิ่ม Pull Request, Code Review และ Branch Protection บริการที่ใช้มีแบบฟรีเพียงพอสำหรับทุก Lab โดยผู้เรียนสมัครบัญชีของตนเอง หน้าจอของ GitHub และ Bitbucket เปลี่ยนบ่อย เนื้อหาจะยึดหน้าจอล่าสุด ณ วันอบรม

06 ภาพรวมการเรียน 2 วัน Big Picture

วัน	หัวข้อหลัก
Day 1 Section 1-5	รู้จัก Version Control, ติดตั้งและตั้งค่า Git, บันทึกงานด้วย add และ commit, ดูประวัติและเปรียบเทียบด้วย diff, ย้อนงานอย่างปลอดภัยด้วย restore และ revert, แตก Branch และแก้ Merge Conflict
Day 2 Section 6-10	ยืนยันตัวตนกับ GitHub ด้วย HTTPS และ SSH, clone, push และ pull, ทำงานเป็นทีมด้วย Pull Request, ใช้ Bitbucket Cloud ด้วย API Token และ Workshop แนวปฏิบัติที่ดีกับโปรเจกต์ทีม

DAY 1

Git บนเครื่องของเรา

เนื้อหาการอบรม · Section 1-5

รวม
6 ชั่วโมง

1 Lab: รู้จัก Git และติดตั้งเครื่องมือ

- Version Control คืออะไร และ Git, GitHub, Bitbucket และ GitLab ต่างกันอย่างไร
- คำศัพท์สำคัญ: Repository, Commit, Branch, Remote และ Merge
- ติดตั้ง Git for Windows พร้อม Git Bash หรือ Git บน macOS และคำสั่ง Terminal เบื้องต้น
- ตั้งค่า user.name, user.email, init.defaultBranch เป็น main และใช้ VS Code เป็น Editor
- ภาพรวม Git 3.0 ที่กำลังจะมา เช่น Branch main เป็นค่าเริ่มต้นและ SHA-256

2 Lab: บันทึกงานด้วย init, add และ commit

- Working Tree, Staging Area และ Repository
- เริ่มโปรเจกต์ใหม่และเพิ่ม Git ให้โปรเจกต์เดิมด้วย git init
- git status, git add และ git commit พร้อมข้อความ Commit ที่ดี
- ยกเว้นไฟล์และโฟลเดอร์ด้วย .gitignore รวมถึงไฟล์ลับอย่าง .env
- เปลี่ยนชื่อ ย้าย และลบไฟล์ด้วย git mv และ git rm

3 Lab: ดูประวัติและเปรียบเทียบการแก้ไข

- git log แบบ --oneline และ --graph และ git show
- git diff ระหว่าง Working Tree, Staging Area และ Commit
- เปรียบเทียบระหว่าง Commit และเฉพาะไฟล์ที่สนใจ
- ดู Diff แบบภาพใน VS Code และหาว่าใครแก้บรรทัดใดด้วย git blame
- สร้างคำสั่งย่อด้วย Git Alias

4 Lab: ย้อนและแก้ไขงานอย่างปลอดภัย

- ยกเลิกการแก้ไขด้วย git restore และ git restore --staged
- แก้ Commit ล่าสุดด้วย git commit --amend
- ย้อน Commit ที่ส่งไปแล้วด้วย git revert
- git reset แบบ soft, mixed และ hard พร้อมข้อควรระวัง
- พักงานชั่วคราวด้วย git stash และกู้ Commit ด้วย git reflog

5 Lab: Branch และ Merge

- สร้างและสลับ Branch ด้วย git switch และ git switch -c
- Fast-forward Merge และการใช้ --no-ff
- Three-way Merge และ Merge Conflict เกิดขึ้นได้อย่างไร
- แก้ Conflict ด้วย Merge Editor ของ VS Code
- Lab: จำลอง Conflict สองแบบแล้วแก้จนรวมงานสำเร็จ

DAY 2

GitHub, Bitbucket และการทำงานเป็นทีม

เนื้อหาการอบรม · Section 6-10

รวม
6 ชั่วโมง

6 Lab: เริ่มต้น GitHub และการยืนยันตัวตน

- สมัครบัญชี GitHub, ตั้งค่าโปรไฟล์ และเปิด Two-factor Authentication
- เหตุที่ใช้รหัสผ่านกับ Git ไม่ได้แล้ว และทางเลือก HTTPS กับ SSH
- HTTPS ด้วย Git Credential Manager หรือ Personal Access Token
- สร้าง SSH Key แบบ Ed25519 แล้วเพิ่มเข้า GitHub
- Lab: ทดสอบการเชื่อมต่อด้วย ssh -T และ gh auth login

7 Lab: Remote Repository: clone, push, fetch และ pull

- สร้าง Repository บน GitHub และเชื่อมกับ Repository บนเครื่อง
- git remote, origin และ Upstream Branch
- git push, git fetch และ git pull ต่างกันอย่างไร
- pull แบบ Rebase และการแก้ Conflict ระหว่าง pull
- Lab: ทำงานสองเครื่องกับ Repository เดียวกัน

8 Lab: ทำงานเป็นทีมด้วย Pull Request

- GitHub Flow: แตก Branch, Commit, เปิด Pull Request แล้ว Merge
- Code Review: Comment, Suggest Changes และ Approve
- ป้องกัน Branch main ด้วย Branch Protection หรือ Ruleset
- Issue, README แบบ Markdown และการ Fork เพื่อร่วมโปรเจกต์โอเพนซอร์ส
- Lab: เปิดและรีวิว Pull Request กับเพื่อนในห้อง

9 Lab: Bitbucket Cloud

- Workspace, Project และ Repository ใน Bitbucket Cloud
- Clone ผ่าน HTTPS ด้วย API Token ซึ่งมาแทน App Password
- ตั้งค่า SSH Key สำหรับ Bitbucket
- Commit, push, pull และ Pull Request บน Bitbucket
- เมื่อไรองค์กรเลือก Bitbucket และการเชื่อมกับ Jira

10 Workshop: แนวปฏิบัติที่ดีและโปรเจกต์ทีม

- Source Control ใน VS Code, GitHub Desktop และใช้ AI ช่วยร่าง Commit Message โดยตรวจก่อนใช้
- ข้อตกลงของทีม: ชื่อ Branch, ข้อความ Commit และป้องกันรหัสผ่านหรือ Key หลุดขึ้น Repository
- Workshop: แบ่งทีมสร้าง Repository กำหนดกติกา Branch main แล้วพัฒนาบน Feature Branch
- เปิดและรีวิว Pull Request แก้ Conflict และ Merge จนได้งานที่สมบูรณ์
- ติด Tag และ Release รุ่นแรก ทบทวนปัญหาที่พบ และแนวทางเรียนต่อ

07 เรียนจบแล้วได้อะไร Outcomes

- **เข้าใจ Version Control** Git, GitHub และ Bitbucket
- **ใช้ Git คล่อง** commit, log, diff และ restore
- **แก้ Conflict ได้เอง** Branch, Merge และ VS Code
- **เชื่อมต่อ Remote ได้** HTTPS, SSH และ Token
- **ทำงานเป็นทีม** Pull Request และ Code Review
- **ใช้ได้ทั้งสองแพลตฟอร์ม** GitHub และ Bitbucket Cloud

08 สิ่งที่ผู้เรียนจะได้รับ Deliverables

- เอกสารประกอบการอบรมครบทั้ง 2 วัน และ Lab Guide ทีละขั้นตอน
- Git Cheat Sheet ภาษาไทยของคำสั่งที่ใช้บ่อย
- Repository ตัวอย่างและโจทย์ Conflict สำหรับฝึกต่อหลังอบรม
- ประกาศนียบัตรจบหลักสูตร (Certificate of Completion) และช่องทางสอบถามวิทยากรหลังจบหลักสูตร

09 ข้อมูลผู้สอน Instructor

อ.สามิตร โกยม ผู้สอนและนักพัฒนาที่ใช้ Git, GitHub และ Bitbucket ในงานพัฒนาซอฟต์แวร์และงานทีมเป็นประจำ เน้นอธิบายสิ่งที่เกิดขึ้นเบื้องหลังแต่ละคำสั่งด้วยภาพ ให้ผู้เรียนลงมือทำและจำลองปัญหาจริง เช่น Conflict หรือ Commit ผิด แล้วแก้เองทีละขั้น เพื่อให้กลับไปใช้ Git กับงานของตนเองและทีมได้อย่างมั่นใจ

ราคาค่าอบรม 6,500 บาท/ท่าน · สนใจ IN-HOUSE ขอใบเสนอราคา

6,500 บาท / ท่าน (ยังไม่รวม VAT 7%)

สมัครผ่านเว็บไซต์ ลด 10%

02-570-8449 · 088-807-9770

@itgenius · contact@itgenius.co.th

เพิ่มเพื่อน LINE