Peer to Peer Session

GROUP CHALLENGE

Group 1: Azure VMs and VMSS

Group 2: Azure Storage Account, Files, and Blob Storage

Group 3: Microsoft Entra ID, RBAC, Azure Roles

Group 4: Azure Networking, VNET, IP, Subnet, NSG, ASG

Group 5: Azure Monitor, Cost Management

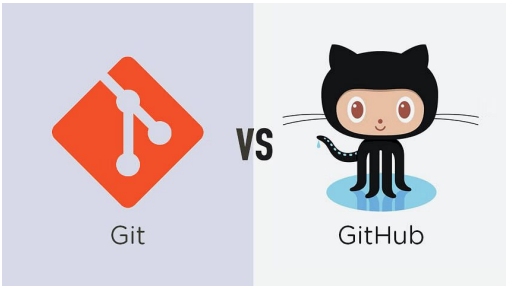
Show and Tell

After the group challenges, we will have a "Show and Tell" session where each group will present their findings and solutions.

Git and Github

Git and Github

- Git:
  - A distributed version control system.
  - Tracks changes in source code during software development.
  - Allows multiple developers to collaborate on a project.
- Github
  - A web-based platform for version control using Git.
  - Provides hosting for software development and version control.
  - Facilitates collaboration among developers.

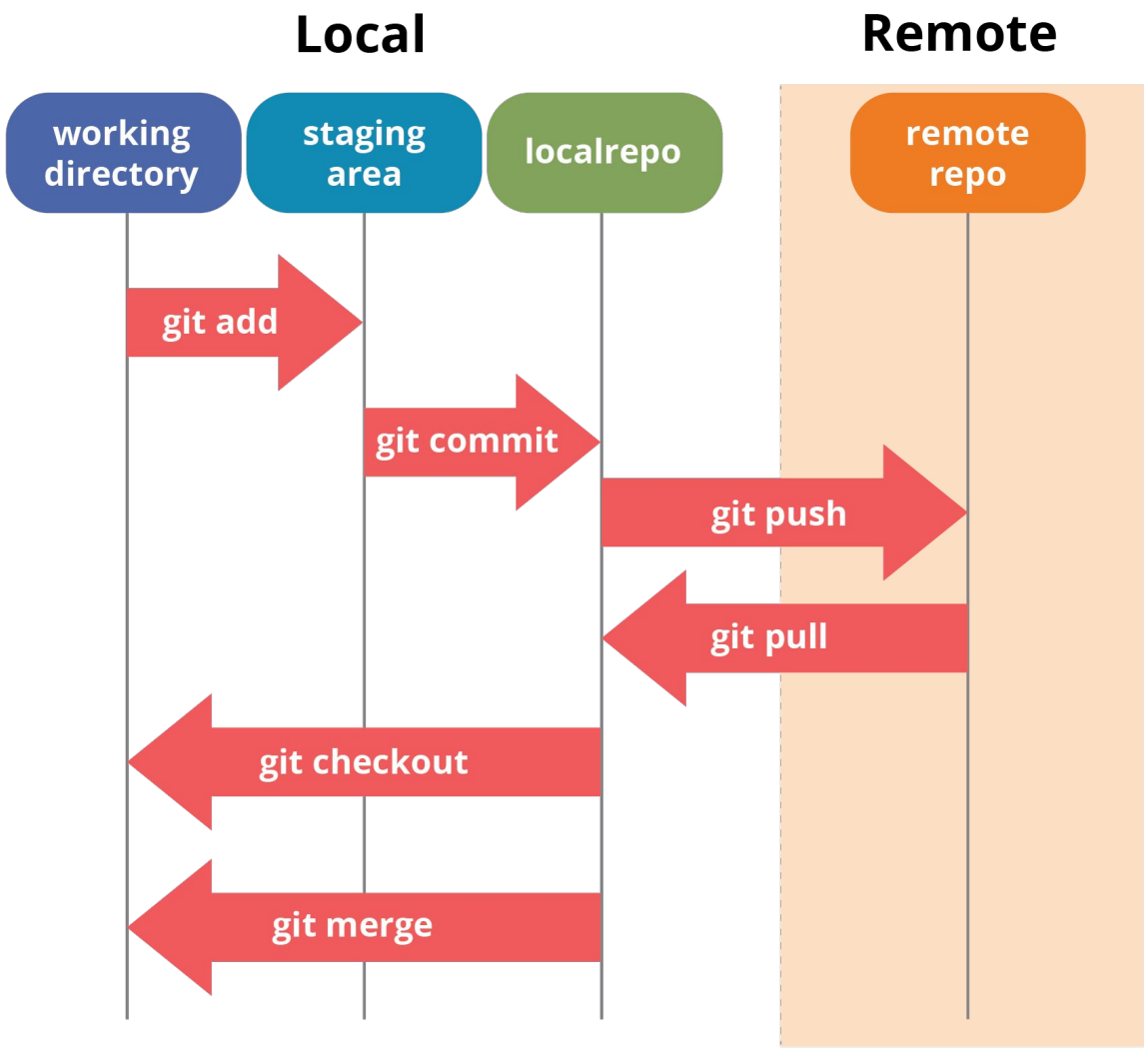


Github Features

- Repository hosting.
- Pull requests for code reviews.
- Issues for bug tracking and feature requests.
- GitHub Actions for CI/CD automation.

Key Concepts in Git

- **Repository (Repo):**
  - A database containing all project files and their revision history.
- **Commit:**
  - A snapshot of changes made to the repository.
  - Each commit has a unique ID (hash).
- **Branch:**
  - A parallel version of the repository.
  - Allows for separate development paths.
- **Merge:**
  - Combining changes from different branches.
- **Clone:**
  - Copying a repository to your local machine.


Local
Remote
working directory
staging area
localrepo
remote repo
git add
git commit
git push
git pull
git checkout
git merge

Setting Up Git

- **Installation:**
  - Download and install Git from git-scm.com.
- **Configuration:**
  - Set your username: git config --global user.name "Your Name"
  - Set your email: git config --global user.email "you@example.com"
- **Creating a Repository:**
  - Initialize a new repository: git init
  - Clone an existing repository: git clone [repository URL]

Basic Git Commands

- **Checking the Status:**
  - git status - Shows the status of changes.
- **Adding Changes:**
  - git add [file] - Stage changes for commit.
  - git add . - Stage all changes.
- **Committing Changes:**
  - git commit -m "commit message" - Commit staged changes with a message.
- **Viewing History:**
  - git log - View commit history.

Branching and Merging

- **Creating a Branch:**
  - git branch [branch-name] - Create a new branch.
- **Switching Branches:**
  - git checkout [branch-name] - Switch to the specified branch.
- **Merging Branches:**
  - git merge [branch-name] - Merge changes from the specified branch into the current branch.
- **Resolving Conflicts:**
  - Manually edit conflicting files.
  - Mark conflicts as resolved: git add [file]
  - Commit the resolution: git commit -m "resolve conflicts"

Working with Remote Repositories

- **Adding a Remote Repo:**
  - git remote add origin [repository URL] - Add a remote repository.
- **Pushing Changes:**
  - git push origin [branch-name] - Push changes to the remote repository.
- **Pulling Changes:**
  - git pull origin [branch-name] - Fetch and merge changes from the remote repository.
- **Fetching Changes:**
  - git fetch origin - Fetch changes from the remote repository without merging.

Git in Cloud Projects

- **Infrastructure as Code:**
  - Store and version control infrastructure scripts (e.g., Terraform).
- **Configuration Management:**
  - Track and manage configurations for cloud environments.
- **Collaboration:**
  - Use Git for team collaboration on cloud projects.
- **CI/CD Integration:**
  - Integrate Git with continuous integration and continuous deployment (CI/CD) pipelines.

Best Practices

- **Commit Often:**
  - Make frequent commits with descriptive messages.
- **Use Branches:**
  - Develop features and fix bugs on separate branches.
- **Regularly Pull Updates:**
  - Keep your local repository up-to-date with remote changes.
- **Review Changes:**
  - Use pull requests and code reviews.
- **Backup Your Work:**
  - Push commits to the remote repository regularly.

Lab

- Learn the basics of Git and GitHub by setting up git, creating a repository, making commits, pushing and pulling.

Assignment

- Write a technical blog on Git and GitHub; include how to set up git, creating a repository, making commits, pushing, pulling and etc 🙂.
- Submission Link: https://docs.google.com/forms/d/e/1FAIpQLSdto61frnJepVK7ep2RyVQ-OyjMJS6lDEQAKUs1GdkCHLRPiA/viewform?usp=sf_link