Certificate Program in DevOps and Cloud Engineering

August	Course and (Session) Names	Pre-Reads	Learning Outcomes	Post-Reads
Week 1	Python Programming (Flask and OOPs in Python)	LINK	• Understand the core principles of OOP: encapsulation, inheritance, and polymorphism. • Design modular applications by encapsulating data and functionality within classes. • Implement inheritance to create hierarchies of related classes, fostering code reusability. • Explore polymorphism to enable flexible and dynamic interactions between objects. • Learn to perform data manipulation operations, including insertion, deletion, and retrieval. • Apply array and set operations to solve real-world scenarios, enhancing problem-solving skills • Develop web applications using Flask with proficiency and understanding. • Create dynamic web applications using Flask. • Create a structure for a Flask project, handle HTTP requests and responses, and effectively integrate templates to render dynamic content. • Design and implement RESTful APIs using Flask	LINK
	Python Programming (Introduction to Database and Flask [Contd.])			
Week 2	Git and GitHub (Introduction to Git and Github)	LINK	• Gain proficiency in using Git for version control, including tracking changes, branching, and merging. • Develop skills in utilizing GitHub for collaborative software development, including forking, cloning, and pull requests.	LINK
	Git and GitHub (Advance Git)		• Master advanced Git techniques, including merging, reverting, and rebasing, to effectively manage code history and resolve conflicts in collaborative development environments. • Develop proficiency in utilizing Git's stashing feature to save and apply changes, ensuring a streamlined and efficient development process while working with multiple code branches.	
Week 3	Networking and servers (Understanding Networking)	LINK	• Acquire a foundational understanding of networking fundamentals, protocols, and components, fostering effective communication and data transfer within networked environments, reflecting comprehension and knowledge application. • Analyze and Implement Network Solutions: Utilize analytical skills to diagnose and resolve network problems, proficiently configure devices, and implement basic network setups for optimal performance, demonstrating application and analysis abilities. • Demonstrate TCP/IP and Security Knowledge: Showcase knowledge of TCP/IP, subnetting, and network security principles, ensuring secure and efficient network operations, and illustrating application and understanding. • Apply Networking Concepts to Solutions: Apply networking concepts to design, deploy, and manage small-scale network solutions, considering emerging technologies and adhering to industry best practices, reflecting application and synthesis skills.	LINK
	Networking and servers (Apache2, Nginx)		• Develop an understanding of Apache2 and its configuration for website hosting and HTTP request management, demonstrating comprehension and application skills. • Optimize Web Traffic: Attain proficiency in deploying and optimizing Nginx server, showcasing expertise in handling web traffic efficiently and applying acquired knowledge effectively.	
Week 4	Automated Testing and Linux and Shell Scripting (Testing using Pytest)	LINK	• Develop a foundational understanding of Pytest fundamentals, showcasing comprehension of core testing concepts. • Create resilient and dependable test cases, demonstrating application skills in designing comprehensive tests. • Utilize Pytest for test automation, illustrating application skills in practical automation scenarios. • Evaluate and interpret test outcomes, reflecting analytical skills in assessing testing effectiveness. • Apply Pytest best practices to enhance testing processes	LINK
	Automated Testing and Linux and Shell Scripting (Introduction to Linux and Commands)		• Grasp core Linux fundamentals and command-line navigation to efficiently manage files, perform system operations, and customize the environment, demonstrating comprehension and application skills. • Develop expertise in executing vital Linux commands, scripting, and resolving common issues for effective system administration, showcasing application and analysis abilities.	
Week 5	Automated Testing and Linux and Shell Scripting (File system and permissions)		• Develop a foundational understanding of file system concepts, structures, and organization, showcasing comprehension skills. • Explain various file permission types and their importance in access control, reflecting comprehension and application. • Demonstrate proficiency in using the chmod command to adjust file permissions within a Unix/Linux environment, illustrating application skills in real-world scenarios.	
	Automated Testing and Linux and Shell Scripting (Bash Programming)		• Understand the basic syntax and structure of Bash scripts. • Demonstrate proficiency in using variables, loops, conditionals, and functions in Bash scripts. • Utilize input/output redirection and command substitution to manipulate data in Bash. • Apply effective debugging techniques to identify and resolve errors in Bash scripts. • Develop practical Bash scripts for tasks such as file manipulation, text processing, and system administration. • How Bash automates tasks for efficient DevOps workflows.	

**The learning outcomes are tentative and are subject to change.*