NSS SOFTWARE TRAINING PLAN

|  |  |  |  |
| --- | --- | --- | --- |
| WEEK | DATE | TOPIC | TUTOR |
| 1 | Thurs 10th October | 1. Programming Basics 2. What is coding/programming 3. Types of programming language 4. Department Procedures 5. Secure Development Policy | Kwaku |
|  | Fri 11th October | 1. Best/Standard Practices    1. Programming rules    2. Standard Procedures    3. Naming Conventions    4. Logic and Logic Gates    5. Pseudocode    6. Flow Charts | Kwaku |
|  | Sat 12th October | 1. Software Development Life Cycle    1. Introduction to SDLC    2. Documentation    3. Waterfall and Agile | Sesha |
| 2 | Mon 14th October | 1. Programming Flows 2. Flowcharts 3. Conditions 4. Parameters and Arguments | Ebenezer |
|  | Tues 15th October | 1. Software Development Principles 2. DRY 3. CRUD 4. SOLID | Chetan |
|  | Wed 16th October | 1. Database 2. What is a database 3. Types of database 4. Relational Databases 5. Database Schema 6. Tables, Column | Ebenezer |
|  | Thurs 17th October | 1. Software Design Pattern 2. What are design patterns 3. Types of software design pattern 4. Creational, Structural and Behavioral Pattern | Chetan |
|  | Fri 18th October | 1. Database 2. Relations and Joins 3. ER and Conceptual Diagram 4. CRUD with database 5. Indexing 6. Optimizing Database Queries | Ebenezer |
|  | Sat 19th October | 1. Linux Basic 2. Linux Commands Basics 3. Virtualization 4. Permissions and Ownership | Sesha |
| 3 | Mon 21st October | A. Database   1. Relations and Joins 2. ER and Conceptual Diagram 3. CRUD with database 4. Indexing 5. Optimizing Database Queries | Ebenezer |
|  | Tues 22nd October | A. Software Design Pattern   1. What are design pattern 2. Types of software design patterns 3. Creational, Structural, Behavioral Patterns | Chetan |
|  | Wed 23rd October | A. OOP   1. What is OOP 2. OOP Fundamentals 3. Classes and Objects 4. OOP Principles | Ebenezer |
|  | Thurs 24th October | Software Design Pattern   1. What are design pattern 2. Types of software design pattern 3. Creational, Structural, Behavioral Patterns | Chetan |
|  | Fri 25th October | OOP   1. OOP Association 2. OOP Case Study (School Management System) | Ebenezer |
|  | Sat 26th October | A. Testing   1. Types of Testing 2. Gorilla Testing 3. ATP Testing 4. Testing using network 5. Documenting test results | Sesha |
| 4 | Mon 28th October | 1. Introduction    1. History and Features of Java    2. Java Development Kit (JDK) and Java Runtime Environment (JRE)    3. Installation and Configuration 2. Basics of Java Programming    1. Basic Syntax    2. Data Types and Variables    3. Operators | Ebenezer |
|  | Tues 29th October | A. Software Architecture   1. Introduction of software architecture 2. Types of software architecture 3. Logical and Physical Architecture | Chetan |
|  | Wed 30th October | **A.**. Git Essentials   1. What is version control 2. Why version control 3. Repository 4. Git Action (Branches, Checkout, Commits, Fetch, Merge etc.) 5. Git Strategies | Kwaku |
|  | Thurs 31st October | **A.** Coding Technqiues   1. Clean Code 2. CI/CD | Chetan |
|  | Fri 1st November | 1. Control Flow Statements    1. **Conditional Statements**: if, if-else, switch    2. **Loops**: for, while, do-while, enhanced for 2. Arrays    1. One-dimensional Arrays    2. Multi-dimensional Arrays | Ebenezer |
|  | Sat 2nd November | **A.** Coding Practices   1. Logging and Commenting 2. Log levels | Sesha |
| 5 | Mon 4th November | 1. Object-Oriented Programming (OOP)    1. Core OOP Concepts       1. Classes and Objects       2. Methods       3. Constructors       4. Inheritance       5. Polymorphism       6. Abstraction       7. Encapsulation | Ebenezer |
|  | Tues 5th November |  |  |
|  | Wed 6th November | 1. Advanced OOP    * 1. Interfaces      2. Abstract Classes      3. Inner Classes      4. Anonymous Classes      5. Generics | Ebenezer |
|  | Thurs 7th November | Secure Coding Principles | Chetan |
|  | Fri 8th November | 1. Data Structures and Collections Framework    1. Basic Data Structures       1. Arrays       2. Linked Lists       3. Stacks       4. Queues    2. Collections Framework       1. List Interface: ArrayList, LinkedList       2. Set Interface: HashSet, TreeSet       3. Map Interface: HashMap, TreeMap    3. Queue Interface: PriorityQueue, Deque | Ebenezer |
|  | Sat 9th November | **A.**Spring framework   1. Dependency Injection 2. Java Bean 3. Spring MVC 4. Creating a spring application | Sesha |
| 6 | Mon 11th November |  |  |
|  | Tues 12th November | 1. Exception Handling    1. Types of Exceptions    2. Try, Catch, Finally    3. Throw and Throws    4. Custom Exceptions | Ebenezer |
|  | Wed 13th November | 1. Java Input/Output (I/O)    1. File I/O    2. Serialization    3. Buffered I/O | Ebenezer |
|  | Thurs 14th November | 1. Multithreading and Concurrency    1. Introduction to Threads    2. Creating Threads: Extending Thread class, Implementing Runnable interface    3. Thread Lifecycle    4. Synchronization    5. Inter-thread Communication    6. Concurrency Utilities: Executor Framework,Locks, Semaphore, CountDownLatch | Ebenezer |
|  | Fri 15th November | 1. Java 8 Features    1. Lambda Expressions    2. Functional Interfaces    3. Stream API    4. Date and Time API    5. Default Methods | Ebenezer |
|  | Sat 16th November | **A.** Logs and Commenting   1. Log Levels 2. Code Commenting | Sesha |
| 7 | Mon 18th November | 1. Java 9-17 Features    1. Modules (Java 9)    2. Local-Variable Type Inference (Java 10)    3. New String Methods (Java 11)    4. Switch Expressions (Java 12)    5. Text Blocks (Java 13)    6. Records (Java 14)    7. Pattern Matching for instanceof (Java 16)    8. Sealed Classes (Java 17) | Ebenezer |
|  | Tues 19th November | 1. Java Design Patterns    * 1. Creational Patterns: Singleton, Factory, Builder      2. Structural Patterns: Adapter, Composite, Proxy      3. Behavioral Patterns: Strategy, Observer, Command | Chetan |
|  | Wed 20th November | 1. Advanced Topics    1. Java Networking       1. Sockets       2. URL and HTTP       3. Java RMI (Remote Method Invocation)    2. Java Database Connectivity (JDBC)       1. Connecting to Databases       2. CRUD Operations       3. Prepared Statement and Callable Statement | Ebenezer |
|  | Thurs 21st November | 1. Introduction to TDD    1. Principles of TDD    2. Red-Green-Refactor Cycle 2. Writing Tests    1. JUnit Framework       1. Annotations (@Test, @Before, @After, etc.)       2. Assertions       3. Parameterized Tests    2. Mockito for Mocking       1. Creating Mocks       2. Defining Behavior       3. Verifying Interactions | Ebenezer |
|  | Fri 22nd November | **A.** Maven   1. Introduction to maven 2. Maven Build 3. Creating a maven project 4. Jar and War | Kwaku |
|  | Sat 23rd November | **A.** Spring framework   1. Dependency Injection 2. Java Bean | Sesha |
|  | Mon 25th November |  |  |
|  | Tues 26th  November | **A.** Spring Boot   1. Introduction To Spring Boot 2. Creating a spring boot application 3. Connect Spring boot application to database 4. Thyme leaf | Ebenezer |
|  | Wed 27th  November | **A.** Spring boot   1. Create a spring boot API 2. Testing with Postman | Kwaku |
|  | Thurs 28th  November | **A.** Spring Boot   1. Spring Security 2. JWT 3. Authentication | Ebenezer |
|  | Fri 29th  November | **A.** Spring Boot   1. Micro service 2. API Gateway 3. Cloud | Kwaku |
|  | Sat 30th  November | **A.** Spring Framework   1. Spring MVC 2. Creating a spring application   **B.** Spring Framework   1. Spring Dependencies 2. Spring Configuration with Application Context 3. Embedded Tomcat | Sesha |
|  | Mon 2nd  December |  |  |
|  | Tues 3rd  December | **A.** Kafka   1. Introduction to Messaging and Streaming 2. Producer and Consumer | Kwaku |
|  | Wed 4th  December | **A.** Kafka   1. Kafka Broker 2. Kafka Clusters 3. Kafka Topics | Kwaku |
|  | Thurs 5th  December | **A.** Keycloak   1. Introduction to Identity and Access Management 2. JWT 3. Authentication | Ebenezer |
|  | Fri 6th  December | **A.** KeyCloak   1. Realms 2. Clients 3. LDAP | Kwaku |
|  | Sat 7th  December | **A.** Docker   1. Virtualization and containerized 2. Docker Commands 3. Docker Images 4. Docker Run 5. Docker Compose 6. Docker Network 7. Docker Volume | Sesha |