* **GIT version control tool**

**Core - Java**

* Java Basics – Architecture, operators, data types, condition statements, loops
* class
* Methods
* Constructors
* Arrays
* Strings
* **OOPs**

1. Inheritance
2. Polymorphism
3. Abstraction
4. Encapsulation
5. Aggregation

* Packages and access modifiers
* Exception Handling

1. Checked and unchecked Exception
2. Try-catch
3. Finally
4. Throw
5. Throws
6. Custom Runtime Exception and compile Time Exception

* MultiThreading

1. Life cycle of Thread
2. Creating a Thread
3. Threads Methods
4. Thread Priority
5. Synchronization and types
6. Deadlock

* Wrapper classes, autoboxing and unboxing
* Collections

1. List
2. ArrayList
3. LinkedList
4. Vector
5. Set
6. HashSet
7. LinkedHashSet
8. TreeSet
9. Map
10. HashMap
11. LinkedHashMap
12. TreeMap
13. HashTable
14. ConcurrentHashMap

* Files 🡪 Reading and writing data

**Java-8:**

* Streams
* Lambda Expressions
* Functional Interface
* Static Method
* Default method
* Predicate & Supplier and Consumer Interface
* Optional class
* Memory changes from Java8
* HashMap changes from java8

**SQL database:**

**Spring**:

* IOC container
* Dependency Injection
* Bean
* Autowiring
* Qualifiers
* Annotations
* Spring boot crud operations - POST, GET, PUT, DELETE Mappings
* Entity designing
* **Hibernate mappings** – oneToOne, manyToOne, OneToMany, ManyToMany
* **Jpa repository** --- Java Persistance API
* **CRUD Repository**
* Jpa anotations
* Jpa queries – inbuilt, customised
* Jpql queries
* SQL -native queries
* Db connections with mysql
* Spring Profiles
* Actuators
* Spring boot Exception’s handlings – custom exceptions and global exceptions
* POST MAN
* Lombok plugin
* Maven build tool
* Tomcat server

**Junit:/ Mockito testing**

* **Microserices**:
* Springcloud API Gateway
* Eureka server-client
* Feign client API
* Resilliance4j
* loggers
* **Student Application**
* **Employee Management**
* **Bank Application**
* **Ecommerce: microservices**