**Core - Java**

* Java Basics – Architecture, operators, data types, condition statements, loops , variables
* OOPs

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

* Arrays and Strings
* 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 IO operations

**Java-8:**

* Streams
* Lambda Expressions
* Functional Interface
* Method reference
* Static Method
* Default method
* Predicate & Supplier and Consumer Interface
* Optional class

**Spring**: Student Application

* IOC container
* Dependency Injection
* Bean
* Autowiring
* Qualifiers - 1
* Annotations
* Spring boot crud operations - POST, GET, PUT, DELETE Mappings
* Entity designing
* **Hibernate mappings** – oneToOne, manyToOne, OneToMany, ManyToMany
* Hibernate joining multiple tables and columns
* **Jpa repository** --- Java Persistance API
* **CRUD Repository**
* Jpa anotations
* Jpa queries – inbuilt, customised
* Jpql queries
* SQL -native queries
* Db connections with mysql
* H2 database- in-memory database – temporary database
* Spring Profiles - 1
* Spring boot Exception’s handlings – custom exceptions and global exceptions
* Swagger ui Doc -1
* POST MAN
* GIT version control tool
* Lombok plugin -1
* Maven build tool
* Tomcat server

**Junit:/ Mockito:**

* **Microserices**:
* Zuul API Gateway
* Eureka server-client
* Rest Template
* Bean creation
* Spring cloud sleuth – loggings
* Zipkin server

**Student**

**Address**

**College**

**Courses**

**Employee:**

**Address**

**Company**

**Projects**