## REST APPLICATION DEVELOPMENT USING SPRING BOOT AND JPA

**Overview:**

This presentation discusses various annotations provided by Spring Boot and Spring for simplifying REST application development using Java. These annotations cover core application components, data access, testing, aspect-oriented programming (AOP), and web development.

**Core Annotations:**

* **@SpringBootApplication:**
  + This meta-annotation is the main entry point for any Spring Boot application.
  + It combines three other annotations: @Configuration, @EnableAutoConfiguration, and @ComponentScan.
* **@Controller:**
  + Defines a class as a controller in Spring MVC architecture.
  + Handles HTTP requests and typically returns a view name to be resolved by a view resolver.
  + Suitable for applications that render HTML views.
* **@RestController:**
  + A specialized version of @Controller designed for RESTful web services.
  + Returns data directly (e.g., JSON or XML) instead of relying on a view resolver.
  + Suitable for applications exposing APIs.
* **@Service, @Repository, @Component:**
  + Stereotype annotations to define and indicate the roles of different Spring components.
  + **@Service:** Indicates a service class in the business logic layer.
  + **@Repository:** Indicates a Data Access Object (DAO) class in the persistence layer.
  + **@Component:** A generic stereotype when a more specific one is not applicable.

**Dependency Injection Annotations:**

* **@Autowired:**
  + Automatically injects dependencies into a Spring bean.
  + Can be applied to fields, constructors, setter methods, or configuration methods.
* **@Qualifier:**
  + Used with @Autowired to specify which bean should be injected when multiple candidates of the same type exist.

**Data Access Annotations:**

* **@Entity:**
  + Defines a data entity managed by the Object-Relational Mapping (ORM) system.
  + Maps class fields to database tables/columns.
  + Provides CRUD operations.
* **@Repository:**
  + Indicates a Spring Data repository, typically a Data Access Object (DAO).
  + Used to perform CRUD operations on entities.
  + Extends CrudRepository or JpaRepository.
* **@Transactional:**
  + Defines the scope of a single database transaction.
  + Applied to methods or classes to indicate that the annotated method or all methods in the class should be wrapped with a transaction.
* **@Query:**
  + Defines a query method using JPQL (Java Persistence Query Language) or native SQL in a Spring Data repository.
* **@Column:**
  + Specifies the details of a column in a database table.
  + Used to customize the mapping of a field to a column.

**Testing Annotations:**

* **@RunWith(SpringRunner.class):**
  + Provides a bridge between JUnit and the Spring TestContext Framework.
* **@SpringBootTest:**
  + Specifies the configuration of the ApplicationContext in a Spring Boot test.
* **@Configuration:**
  + Indicates that a class declares one or more @Bean methods and may be processed by the Spring container.
* **@Value:**
  + Injects values from properties files or other sources into fields.
* **@Conditional:**
  + Conditionally activates a component or configuration class based on the evaluation of a Condition.
* **@Scheduled:**
  + Defines when a method should be invoked based on a fixed rate or cron expression.
* **@EnableAutoConfiguration:**
  + Enables Spring Boot’s auto-configuration mechanism.

**AOP (Aspect-Oriented Programming) Annotations:**

* **@Aspect:**
  + Identifies a class as an aspect.
  + Contains advice methods and pointcuts.
* **@Before, @After, @Around:**
  + Define advice methods to run before, after, or around a method execution.
* **@Pointcut:**
  + Defines a reusable pointcut expression.

**Spring Web Annotations:**

* **@RequestBody:**
  + Binds the method parameter to the body of the HTTP request.
* **@ResponseStatus:**
  + Sets the HTTP response status code.
* **@ExceptionHandler:**
  + Handles exceptions in a specific controller or globally.
* **@ModelAttribute:**
  + Adds attributes to the model before the handler method is invoked.