[Bài đọc] Annotation @Service

@Service là một trường hợp đặc biệt của @Component được sử dụng để đánh dấu các class thuộc tầng service. Dưới đây là một ví dụ một class được đánh dấu là @Service

@Service  
public class HibernateCustomerServiceImpl implements CustomerService {  
 private static SessionFactory *sessionFactory*;  
 private static EntityManager *entityManager*;  
  
 static {  
 try {  
 *sessionFactory* = new Configuration()  
 .configure("hibernate.conf.xml")  
 .buildSessionFactory();  
 *entityManager* = *sessionFactory*.createEntityManager();  
 } catch (HibernateException e) {  
 e.printStackTrace();  
 }  
 }  
 @Override  
 public List<Customer> findAll() {  
 String queryStr = "SELECT c FROM Customer AS c";  
 TypedQuery<Customer> query = *entityManager*.createQuery(queryStr, Customer.class);  
 return query.getResultList();  
 }  
}

Spring chỉ tìm kiếm và chọn ra những class được đánh dấu với @Service annotaion để khởi tạo và đăng ký với ApplicationContext do đó để Spring có thể tìm được tất cả các service trong ứng dụng chúng ta cần khai báo.

import org.springframework.context.ApplicationContext;  
import org.springframework.context.ApplicationContextAware;

Và cần phải sử dụng tới @ComponentScan(“com.codegym”) khi đó file AppConfig của chúng ta sẽ như sau

@ComponentScan("com.codegym")  
public class AppConfig implements WebMvcConfigurer, ApplicationContextAware {  
  
 private ApplicationContext applicationContext;  
  
 @Override  
 public void setApplicationContext(ApplicationContext applicationContext) throws BeansException {  
 this.applicationContext = applicationContext;  
 }  
}