package com.example;  
import org.springframework.context.annotation.Bean;  
import org.springframework.context.annotation.Configuration;  
  
@Configuration  
public class AppConfig {  
  
 @Bean  
 public HelloService helloService() {  
 return new HelloService();  
 }  
}

package com.example;  
  
public class HelloService {  
  
 public void sayHello() {  
 System.*out*.println("Hello from Spring!");  
 }  
  
}.  
  
  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.annotation.AnnotationConfigApplicationContext;  
  
public class Main {  
 public static void main(String[] args) {  
 ApplicationContext context = new AnnotationConfigApplicationContext(AppConfig.class);  
 HelloService hello = context.getBean(HelloService.class);  
 hello.sayHello();  
 }  
}

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>com.example</groupId>

<artifactId>basicSpringApp</artifactId>

<version>1.0-SNAPSHOT</version>

<properties>

<maven.compiler.source>17</maven.compiler.source>

<maven.compiler.target>17</maven.compiler.target>

<project.build.sourceEncoding>UTF-8</project.build.sourceEncoding>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>5.3.33</version>

</dependency>

</dependencies>

<repositories>

<repository>

<id>central</id>

<url>https://repo.maven.apache.org/maven2</url>

</repository>

</repositories>

</project>

2.

public class AppConfig {  
  
 @Bean  
 public CustomerService customerService() {  
 return new CustomerService();  
 }  
  
 @Bean  
 public Customer customer() {  
 Customer customer = new Customer();  
 customer.setCustomerService(customerService());  
 return customer;  
 }  
}

package com.ex1;  
  
public class Customer {  
 private CustomerService customerService;  
  
 // Setter Injection  
 public void setCustomerService(CustomerService customerService) {  
 this.customerService = customerService;  
 }  
  
 public void useService() {  
 customerService.serve();  
 }  
}

package com.ex1;  
public class CustomerService {  
 public void serve() {  
 System.*out*.println("Customer service is being executed.");  
 }  
}

package com.ex1;  
  
  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.annotation.AnnotationConfigApplicationContext;  
  
public class Main {  
 public static void main(String[] args) {  
 ApplicationContext context = new AnnotationConfigApplicationContext(AppConfig.class);  
 Customer customer = context.getBean(Customer.class);  
 customer.useService();  
 }  
}

3.

package com.ex2;  
  
import org.springframework.context.annotation.Bean;  
import org.springframework.context.annotation.Configuration;  
  
@Configuration  
public class AppConfig {  
  
 @Bean  
 public GreetingService greetingService() {  
 return new GreetingService();  
 }  
}

package com.ex2;  
  
public class GreetingService {  
 public void greet() {  
 System.*out*.println("Hello from Exercise 3 - Spring Maven Project!");  
 }  
}

package com.ex2;  
  
  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.annotation.AnnotationConfigApplicationContext;  
  
public class Main {  
 public static void main(String[] args) {  
 ApplicationContext context = new AnnotationConfigApplicationContext(AppConfig.class);  
 GreetingService gs = context.getBean(GreetingService.class);  
 gs.greet();  
 }  
}

4.

package com.ex3;  
  
  
import org.springframework.context.annotation.Bean;  
import org.springframework.context.annotation.Configuration;  
  
@Configuration  
public class AppConfig {  
  
 @Bean  
 public Engine engine() {  
 return new Engine();  
 }  
  
 @Bean  
 public Car car() {  
 // For constructor injection  
 return new Car(engine());  
  
 // For setter injection (uncomment if you're using setter)  
 // Car car = new Car();  
 // car.setEngine(engine());  
 // return car;  
 }  
}

package com.ex3;  
  
public class Car {  
 private Engine engine;  
  
 // Constructor Injection  
 public Car(Engine engine) {  
 this.engine = engine;  
 }  
  
 public void drive() {  
 engine.start();  
 System.*out*.println("Car is driving...");  
 }  
}

package com.ex3;  
public class Engine {  
 public void start() {  
 System.*out*.println("Engine started!");  
 }  
}

package com.ex3;  
  
  
import org.springframework.context.ApplicationContext;  
import org.springframework.context.annotation.AnnotationConfigApplicationContext;  
  
public class Main {  
 public static void main(String[] args) {  
 ApplicationContext context = new AnnotationConfigApplicationContext(AppConfig.class);  
 Car car = context.getBean(Car.class);  
 car.drive();  
 }  
}