1. 빌드 : 소프트웨어 프로그램을 실행가능한 형태로 만드는 과정

단순하고 표준화된 빌드가 필요하다면: Maven

성능과 유연성을 중시하고 복잡한 빌드가 필요한 경우: Gradle

2. Jar vs War

둘 다 패키지

jar -> java 클래스,라이브러리,어플리케이션 들어가 있고

war -> html,jsp 관련이 들어가 있음

3. 의존성도구

의존성이란 한 소프트웨어가 다른 소프트웨어나 라이브러리를 필요로 하는 것을 의미

웹 애플리케이션이 웹 프레임워크 Spring을 사용하는 경우가 의존성에 해당함

의존성 예시

JDBC:

Java 애플리케이션에서 데이터베이스와 연결하려면 JDBC 라이브러리가 필요합니다. 이때 JDBC는 애플리케이션의 의존성입니다.

Spring Framework:

Spring Boot 애플리케이션을 만들 때, Spring의 여러 기능을 사용하려면 Spring 라이브러리(예: Spring Web, Spring Data 등)가 필요합니다. 이러한 라이브러리들이 의존성입니다.

JUnit:

테스트를 작성하려면 JUnit 라이브러리가 필요할 수 있습니다. 이 경우 JUnit은 테스트 코드에서 의존하는 라이브러리입니다.

4. controller

쉽게 말해서 조종하는 거임

Model View Controller(MVC) : Model(백)과 view(프론트) 사이 다리 역할

텍스트, 스크린샷, 직사각형, 도표이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

서버 안에 3개 다 있음

Controller 종류 여러가지 있지만 여기서는 2가지만 다룸

1. @Controller -> html코드 반환(return hello.c)

2. @RestController -> 데이터 반환(return “Hello”)

반환한게 화면에 뜨게 됨.(return 뒤 보면 다름)

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

5. controller안 변수 넣기

1) @PathVariable ->url/name 하면 name을 가져와서 씀

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

변수이름을 바꾸고 싶으면스크린샷, 텍스트, 폰트이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

값이 없으면 Error 뜰거임(경로를 찾지 못했다는 404오류)

2) @RequestParam->url/name=bin&age=12 이렇게 변수에 값 넣듯이 해야 함(url쪽 보기)

텍스트, 스크린샷, 소프트웨어, 운영 체제이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

추가로 v3에서 String,int,Integer 등 단순한 타입이면 @RequestParam 생략 가능

Required = false/true(필수 설정), defaultValue = “-1” or “string”

텍스트, 스크린샷, 폰트이(가) 표시된 사진

AI가 생성한 콘텐츠는 부정확할 수 있습니다.

@PathVariable에서는 defaultValue 못 넣음. url 안에 들어가 있기 때문에 필수임