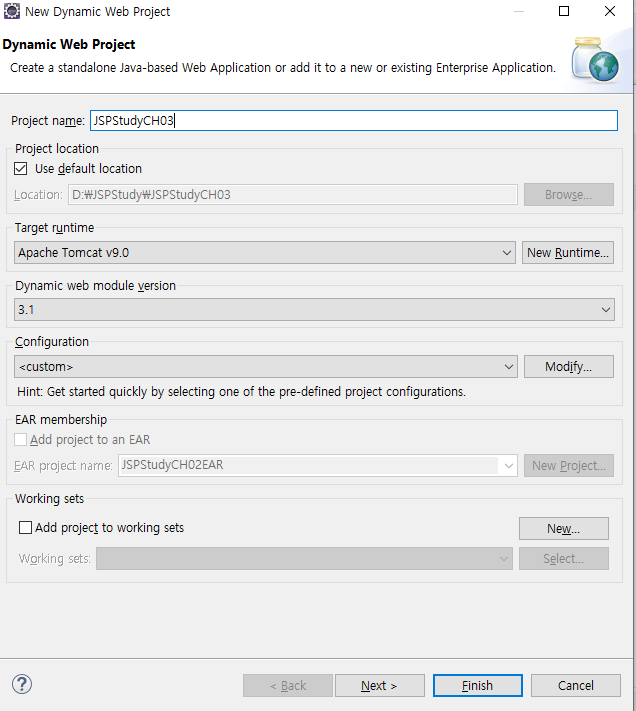
2024-02-01

  
동적인파일

Servlet 3.1 (위로 올릴 수 있음 아래로 내릴 수 없음)

Webapp \_ JSPClassCh03과 같다

WEB\_INF :

도메인 가지고 있다고 생각

패키지는 다 소문자

**package** com.jspstudy.ch03.classes;

**public** **class** ClasesTest {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

// 오브젝트

String msg = "안녕하세요";

java.lang name = **new** String("홍길동");

System.***out***.println(name + "님" + msg);

//util 패키지에 있다

java.util.Calendar cal = java.util.Calendar.*getInstance*();

}

}

Equals : 문자열의 내용이 같은지 비교할 때

Final 붙어있으면 더 이상 상속이 불가능

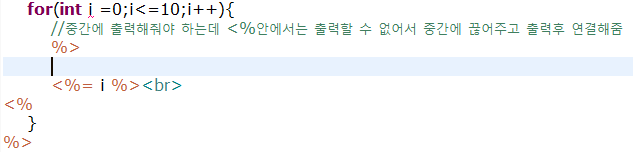
서블릿으로 관리해주는 WAS => 톰캣 서버가 필요하다  
서블릿 컨테이너

500번 서버가 잘못한 것

400번대 사용자 잘못  
- 404번 없는 곳으로 들어간 것  
- 403번 들어가면 안될 곳 들어간 것

페이지에 대한 설정함  
j네는 <% 로 시작해서 %> 로 끝남 -> 무엇이 들어가냐에 따라 용도가 달라짐

  
=이 들어가면 그 변수를 여기에 출력하라는 것



  
pageEncoding을 utf-8로 저장해라

* Include 지시자

컴파일 타입에 다른 JSP 페이지의 내용을 현재 JSP 페이지에 포함할 수 있다.  
현재 JSP페이지(부모페이지) 아래에 include 지시자로 지정한 JSP페이지(자식페이지)가 하나로 합쳐져서 Servlet 클래스로 변환되고 컴파일 되어 하나의 서블릿으로 동작하게 된다.

value 오브젝트는 vo를 사용한다.

자동으로 생성자를 만들어준다

오브젝트를 상속받는다 class Student는 오브젝트를 상속받는다

Class 다중상속 허락하지 않는다

기본생성자를 내가 만들어 준 다음에 그 다음에 쓰면 오류가 나지 않는다.

컴파일 하는 방법을 배웠다

오브젝트를 직접적으로 간접적으로 상속받을 수밖에 없는 시스템이다

캡슐화

출력은 다른 class에서 처리함

Request 사용자가 보낸 데이터 아이피 정보 서버정보 요청 처리하기 위한 톰캣 객체

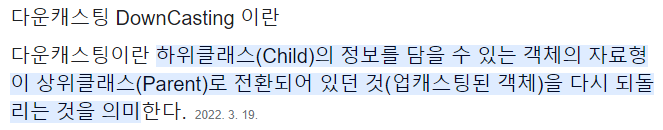
그것을 넣어줘서 사용할 수 있음

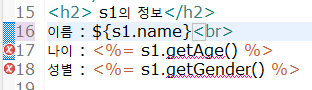
그것을 저장하기 위한 객체이다.

톰캣 안에

업캐스팅과 다운캐스팅

Request는 응답을 하고 사라지고 다시 응답을 받는다

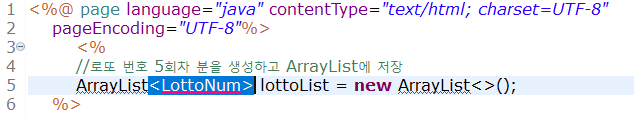
다운캐스팅하고 출력할 수 있다  




${s1.name} 이렇게 사용할 수 있는데, 그러면 attribute 할 필요도 없이 간단하게 사용이 가능

lottoNum은 로또 (실행할 때는 이 j네를 실행해야 됨

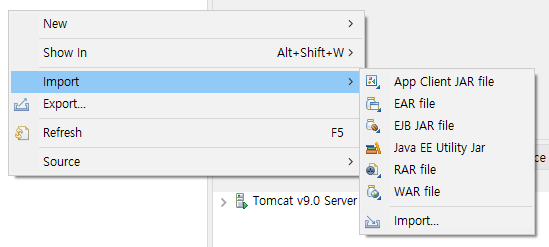
lottoNumResult는 화면출력

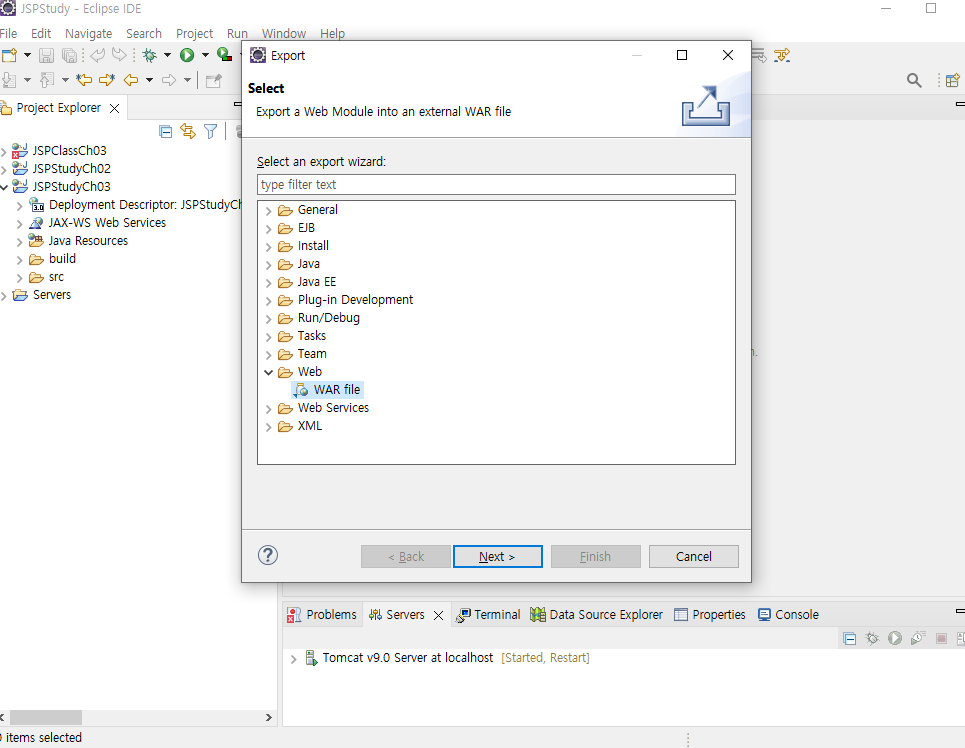
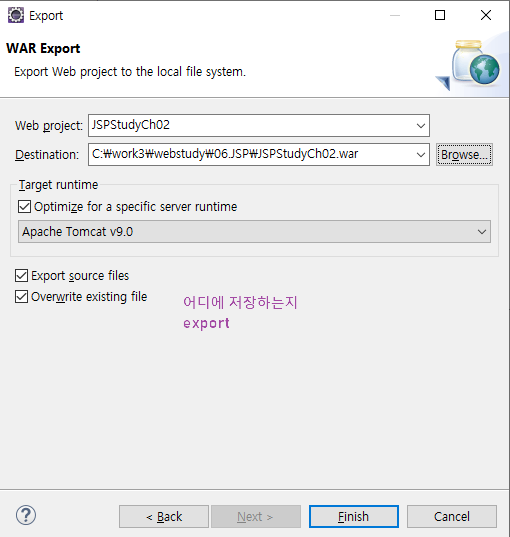
<LottoNum> 제네릭 형의 안정성을 위해 사용한다

Ctrl+space 눌러서 사용하는 것 지시자 선언해준다

  
태그 라이브러리 지시자 -> 접두어를 쓰자 / 태그랑 충돌날수 있으니까 uri쓴다

<c:if > </c:if> 톰캣이라서 열고닫는 것 분명하게 확인해줘야 함

파일 import하는 방법  


파일 export하는 방법  
   
서버에 배포하는 것이기 때문에 소스가 필요 없다  


for문쓰는 두가지 방법  


  
다른 화면에서는 student를 사용하여 써주겠다는 것 그 student는 student1이다!!