웹어플리케이션 시작하기

|  |  |
| --- | --- |
| 1 | 다음 설명 중 틀린 것을 고르면?  (1) Java 언어로 개발할 수 있는 분야는 크게 3가지 플랫폼이 있으며, 이 플랫폼에는 JavaSE, JavaEE,  JavaME 가 있다.  (2) 위 3가지 플랫폼 중 가장 기본이 되는 플랫폼은 Java SE이므로 JavaEE나 JavaME 기반의 어플리케이션  을 개발하려면, 반드시 JavaSE는 기본적으로 설치가 되어 있어야 한다.  (3) JavaSE로 제작된 프로그램은, 이 프로그램을 실행하려는 사용자의 PC에 프로그램 코드가 배포되어 있  어야 한다.  (4) JavaEE 기반으로 제작되어진 웹프로그램을 서버 컴퓨터에 배포해 놓기만 하면, ~~이 웹 프로그램을사용하~~  ~~려는 자는 자신의 개인 PC에 Java를 설치하고서, 서버로 부터 해당 프로그램을 다운로드 받은 후 실행할 수 있다~~ |
| 2 | 웹어플리케이션에 대한 설명 중 틀린 것을 고르면 ?  (1) JavaEE 기반 프로그램으로 제작된 프로그램은 서버에 배포되며, 이 프로그램을 이용하려는 자는 오직  웹브라우저를 이용하여 서버에 접속하기만 하면 된다.  .  (2) JavaEE기반으로 작성된 웹어플리케이션을 구성하는 파일 중 HTML,CSS, 이미지, JS 등 클라이언트에  디자인 적으로 보여지는 기술들은 Front-End 영역에 속한다.  **(3) 사용자가 Front-End 영역에 속하는 파일을 웹브라우저로 요청할 경우, 서버는 해당 자원을 클라이언트에**  **그대로 전송해 버린다.**  **(4) 서버에 배포된 웹어플리케이션을 구성하는 파일 중 jsp 파일 등은 클라이언트가 웹브라우저로 직접 해석**  **할 수 없는 기술이므로, 클라이언트 웹브라우저에 직접 전송되지 않고 서버에서 실행을 해버리며 그 결**  **과를 HTML형식으로 웹브라우저에 전송한다**  (5) 위(4)번과 같이 클라이언트 측에서 실행되지 않고, 서버에서 실행되어 버리는 기술을 Back-End 라 한다 |
| 3 | 다음 설명 중 틀린 것을 고르면?  (1) 웹어플리케이션은 중앙의 서버가 실행을 담당하므로 서버 컴퓨터의 사양과 성능 등이 중요한 반면, 사용  자는 어떠한 추가적 설치나 환경을 구축해야 할 부담 없이, 오직 웹브라우저 하나만 준비하면 된다.  ~~(2) 유지 보수성이나 배포의 편의성을 고려해본다면, JavaSE가 JavaEE에 비해 훨씬 유리하다.~~  (3) Front-End 영역의 파일들은 서버에서 실행되지 않으며, 클라이언트의 웹브라우저가 서버로 부터 해당  파일을 전송받고, 웹브라우저 스스로가 다운로드 받은 파일을 직접 실행하는 방식이다.  **~~(4) 위(3)번의 논리대로 라면, 클라이언트 사용자 PC에는 웹서버로부터 다운로드 한 파일이 계속 쌓이게 되~~**  **~~므로, 이 다운로드된 파일을 사용자가 주기적으로 지워 줘야 한다.~~**  ~~(5) HTML은 그냥 문서이지만, JS 파일은 프로그래밍 언어이므로 중요한 데이터베이스 접속 정보라던가~~  ~~보안적으로 중대한 정보 등을 작성해놓기에 적절하다~~  **~~(6) 웹어플리케이션 개발 시 어떤 언어를 사용하느냐에 따라, 데이터베이스 제품의 종류도 달라져야 한다~~** |
| 4 | 가장 최적화된 개발 및 운영 환경이 될 수 있도록 아래의 표를 채우세요   |  |  |  |  | | --- | --- | --- | --- | | 웹프로그래밍 언어 | 서버용 OS | 웹서버 프로그램 WAS | DBMS | | ASP, ASP.NET (MS) | Windows Server | IIS (Internet Information Service) | MSSQL | | PHP | Linux | Apache | MySQL | | JSP | Linux, Unix | WebLogic, JBoss, JEUS | Oracle | |
| 5 | 개발환경 구축과 관련한 설명 중 틀린 것을 고르면?  웹 어플리케이션 서버 (WAS) – HTTP, DNS, Telnet, 스트리밍, FPT 서버 등  웹 서버: 위 기능을 모두 포함  (1) 웹서버는 HTTP 서버 , FTP 서버, DNS 서버 등을 지원하는 웹용 프로그램 이다.  (2) 웹서버의 종류는 윈도우 계열의 IIS, Linux 계약의 Apache 등이 있으며, 이들 웹서버는 모두 JavaEE 어플  케이션을 직접 단독으로 실행할 수 있는 능력이 있다.    (3) JavaEE 기반 웹어플리케이션을 실행하기 위해서는 Java 기술을 이해하고, JavaEE API를 이해할 수 있  는 WAS 프로그램이 필요한데, 이러한 프로그램을 웹어플리케이션 서버라고 한다.  (4) WAS와 웹서버는 동일한 개념이다  (5) WAS는 일반적인 웹서버의 기능 중 HTTP 서버 로써의 능력만 있으며, 이에 더해 서버 측에서 실행되는  어플리케이션을 해석 및 관리하는 컨테이너 로써의 능력이 포함된 프로그램이다. |
| 6 | Tomcat 설치 및 실행 실습을 마친 후, 다음 중 틀린 것을 고르면?  (1) 자바 기반의 WAS에는 Apache의 Tomcat, 오라클의 WebLogic, 레드햇의 JBoss, 티맥스소프트의 JEUS  등이 있다.  (2) Tomcat도 WAS에 해당하기는 하지만, Tomcat에는 일반적인 WAS가 지원하는 클러스터링, 부하 분산,  보안 등의 기능이 포함되어 있지 않으므로 비교적 작은 규모의 웹어플리케이션 구축에 사용된다.  (3) JavaSE 개발 시엔 JDK를 개발자의 PC에 설치해야 하고, JavaEE 개발 시엔 JavaEE 개발용 SDK인  JavaEEJK를 설치해야 한다.  (4) JavaEE 기반 웹어플리케이션 개발시엔 별도의 SDK (소프트웨어 개발도구)를 설치할 필요가 없다.  (5) 위(4)번의 이유는 Sun에서는 JavaEE의 기술 스팩 만을 제시하고, 그 기준에 따라 각 벤더사들이 WAS  를 제작하기 때문이다. 이이 따라 JavaEE의 스팩을 따르는 구현체 파일들은 해당 WAS 제품 각각에 포함  되기 때문이다.  (6)위(5)번의 이유로, 자바 웹어플리케이션 개발을 위해서는 오직 WAS만 설치하면 되며 별도로 추가해야할  개발도구는 없다.    (7) Tomcat에서 개발 및 테스트된 웹어플리케이션이 있다고 가정할 때, 추후 WAS의 종류가 WebLogic으로  교체 될 경우, 개발자는 WebLogic 용으로 어플리케이션을 코드를 다시 작성해야 한다.  (8) WAS의 개발 회사는 서로 틀리지만 모두 Sun의 기준 스팩에 따라 제품을 제작하였으므로 JavaEE API  객체들의 명칭, 메서드 등 도 동일하게 유지된다. 따라서 개발자는 어떤 WAS를 선택하더라도 Sun의  공식 API 문서로 개발할 수 있으며 프로그램 코드 또한 바꾸지 않아도 된다. |
| 7 | 다음 설명 중 틀린 것을 모두 고르면?  (1) JavaEE 기반의 웹서비스를 구축하려면 JavaEE 기반의 코드가 실행될 수 있는 전용 서버 프로그램인  WAS 를 이용해야 하는데, 이 WAS 프로그램을 개발하는 업체는 바로 Sun 사이다    (2) JavaEE 전용 서버의 제조사마다 작동 방식이 틀리므로, 제품이 바뀌면 자바 개발자들은 웹프로그램  소스 코드도 변경해야 한다.  (3) HTML/CSS같은 정적 자원들을 서비스하기 위한 용도의 서버를 일반적으로 HTTP 서버라 하는데,Tomcat  은 HTTP 서버 로써의 능력을 가지고 있다.  (4) 만일 FTP서버, DNS 서버, SSH 서버 등의 추가 기능이 필요할 경우 WAS 만으로는 한계가 있으며, 이때  는 웹서버의 지원이 필요하고, 단지 Tomcat는 웹컨테이너로 써의 역할만 수행하면 된다. |
| 8 | JavaEE(Enterprise Edition)의 기업 업무용 어플리케이션 시장의 독점을 견제하기 위한 마이크로 소프트의  개발 플랫폼은 무엇이며 JavaEE와의 차이점을 설명하시오. |