

웹 기반의 자바 프로그래밍 강의 교안 개발

이충기

명지대 컴퓨터소프트웨어학과

(2002. 4. 28. 접수)

The Development of a Web-based Java Programming Course Contents

Chungki Lee

Department of Computer Software Myongji University

(Received April 28, 2002)

국문요약

최근에 인터넷과 웹의 이용이 보편화됨에 따라 이를 교육에도 활용하고자 하는 많은 노력이 전 세계적으로 교육자들에 의해 이루어지고 있다. 웹 기반 강의의 장점은 언제 어디서나 저렴한 비용으로 교육받을 수 있다는 것이다. 웹을 이용하여 자바 프로그래밍을 배우고자 하는 수요도 늘어나고 있다. 이에 따라 자바 프로그래밍을 웹 기반 강의로 가르치기 위한 교안이 많이 개발되어 사용되고 있다. 이러한 교안들은 프로그래밍 기법과 관련된 내용이 부족하고 흔히 실습 환경과 분리되어 있다. 본 논문은 이러한 문제를 해결하기 위한 교안 개발 방법과 개발 사례를 제시한다. 또한 교안의 학습자들에 의한 교안 평가 결과를 보고한다.

Abstract

Recently, as the use of the Internet and the Web becomes universal, there has been a great deal of efforts to use it for education among educators throughout the world. The advantage of a Web-based lecture is that people can take it anywhere at a low cost anytime. The demand for learning java programming using the Web is increasing. Accordingly, a number of Web-based Java programming course contents have developed and used. Problems with existing contents are that materials about how to program are not sufficient and the presentation of the lecture is most often separated from the programming environment. In this paper, we present a methodology for developing course contents of Java programming to solve these problems and develop them using it. Then we report the evaluation result of course contents by students who took a Web-based lecture using them.

1. 서론

최근에 인터넷과 웹의 이용이 보편화됨에 따라 이를 교육에도 활용하고자 하는 많은 노력이 전 세계적으로 교육자들에 의해 이루어지고 있다. 컴퓨터, 방송과 통신망을 이용한 원격 교육은 지난 30여 년 동안 이루어져 왔다. 그러한 교육은 일부 성과를 거두기도 하였으나 교육 효과가 전통적인 면 대 면 강의에 비해 낮았기 때문에 보편화되지 못하고 제한된 여건과 환경에서만 활용되었다.

그러나 인터넷과 웹이 생활필수품과 같은 존재로 우리 생활의 일부로 자리잡음에 따라 이를 교육에도 활용할 가능성은 더 많은 관심을 불러 일으켰다. 그 결과로 많은 관련 연구가 수행되고 관련 기술이 개발되었다(Ausserhofer, 1999a).

따라서 사이버 교육을 효율적으로 수행할 수 있는 사이버 교육 솔루션은 기술적 관점에서는 안정화 단계에 들어섰다고 볼 수 있다.

최근에 정식 학사 학위를 주는 국내외 사이버 대학들이 개교하여 성공적으로 운영되고 있는 것이 좋은 예이다. 대학뿐만 아니라 기업에서도 가상 교육을 통한 재교육이 이미 시행되고 있으며 이를 일반인들에게 공개하는 과정에 있다(이중연, 1999). 또한 실제 사이버 교육/학습 사례도 많이 발표되고 있다(강숙희, 1999; 이충기, 2001; Ross, 2000).

웹 기반 교육을 위한 콘텐츠는 수많은 잠재 고객을 가지고 있다. 웹 기반 강의의 가장 큰 장점은 언제 어디서나 많은 수의 학습자에게 각자의 능력에 맞게 학습을 할 수 있다는 것이다(McCormack, 1998). 빠른 기술 발전에 따른 직장인들의 재교육 필요성이 늘어나고 언제 어디서나 저렴한 비용으로 교육을 받고자 하는 수요가 증대되고 있으므로 웹 기반 강의의 전망은 매우 밝고 잠재 고객 계층도 매우 많다고 볼 수 있다.

웹 기반 강의를 위한 소프트웨어 개발 및 활용 사례 등은 많이 발표되고 있다(김동식 등, 2000; 허원, 2001; Ausserhofer, 1999b).

인터넷과 웹의 이용이 폭발적으로 늘어남에 따라 이를 위한 소프트웨어의 개발이 국가 경쟁력을

향상시키는 데 필수적인 요소가 되고 있다. 이에 따라 인터넷과 웹을 위한 프로그래밍 언어로 개발된 자바 언어로 프로그래밍 교육을 할 필요성이 날로 증대되고 있다.

반면에 적지 않은 직장인들과 대학생들이 언제 어디서나 적은 비용으로 자바 프로그래밍을 배우고자 하는 수요가 늘어나고 있다. 이러한 수요에 따라 자바 프로그래밍을 웹 기반 강의로 가르치기 위한 교안이 콘텐츠 개발 회사, 국내외 사이버대학과 정보기술 전문 교육기관 등에 의해 많이 개발되어 운영되고 있다.

개발된 교안은 교수자가 강의 노트를 설명하는 것을 녹화한 유형이거나 텍스트 중심의 자율학습용 유형이다. 교안의 주 내용은 자바 언어의 주요 개념과 이를 반영한 문법적 요소를 설명하고 간단한 프로그램 예제와 이해 점검용 연습 문제 등으로 구성되어 있다.

이러한 교안들은 주어진 문제를 자바 프로그래밍으로 해결하는 방법에 대한 설명이 거의 없거나 매우 부족하다. 또한 교안이 프로그래밍 환경과 분리되어 있어 제시된 프로그램 예제들을 그대로 혹은 수정하여 컴파일하고 실행할 수 없다. 그러므로 학습자가 교안을 학습한 후 자바 언어에 대해서는 배우지만 주어진 문제를 자바로 프로그램 하기가 쉽지 않다.

본 논문은 이러한 문제를 해결하기 위하여 정형화된 강의 내용 외에 프로그래밍 기법과 실습을 강조한 교안을 개발한다. 먼저 자바 프로그래밍 교과목의 특성을 분석하고 교수자와 학습자의 요구사항을 만족시키기 위한 교안 개발 방법을 제시한다. 다음으로 이러한 개발 방법에 따라 개발된 교안 개발 사례를 보고한다. 마지막으로 교안의 학습자들과 콘텐츠 전문가들에 의한 교안 평가 결과를 제시한다.

2. 교안 개발 목표 및 방법

2.1 교안 개발 목표

인터넷을 통한 웹 기반 강의는 시공간을 초월하여 교육을 받을 수 있을 뿐만 아니라 양방향의

대화형 통신이 가능하고 브라우저가 핵심 응용으로 자리잡음에 따라 간단하면서도 익숙한 인터페이스를 제공하는 장점이 있어 교안 개발을 잘 하면 대 면 강의에 근접한 교육 효과를 기대할 수 있다. 따라서 대 면 강의에 근접한 교육 효과를 얻는 것이 첫 번째 개발목표이다. 또한 개발하는 교과목이 프로그래밍 과목이므로 학습자가 교안을 학습한 후 자바 언어의 주요 개념과 이를 반영한 문법적 요소뿐만이 아니라 주어진 문제를 프로그래밍 할 수 있는 능력을 갖도록 하는 것이 두 번째 개발 목표이다.

2.2 교안 개발 방법

웹 기반 교안을 개발하는 것은 기술적 관점에서 쉽다. 그러나 교과목의 내용을 분석하여 그에 따라 기획하고 설계하고 개발하는 것은 매우 어렵다.

특히 웹 기반 강의는 전통적인 대 면 강의와는 다른 매체, 사용자 인터페이스, 방법론을 사용하기 때문에 교과목의 특성 분석과 병행하여 교안의 사용자인 학습자와 교수자의 요구와 환경을 가능한 한 충분히 수용해야 한다. 자바 프로그래밍 과목의 특성은 다음과 같다:

- (1) 이론보다 프로그래밍 실습과 연습이 더 중요하다.

대부분의 프로그래밍 과목의 강의 교안은 문법적인 요소에 대한 설명과 프로그래밍 예제 중심으로 이루어져 있다. 자바 언어의 문법적인 요소는 배우기가 상대적으로 쉽다. 또한 프로그래밍 예제도 학습자가 이해할 수 있다. 그러나 이러한 이론적인 지식을 활용하여 주어진 문제를 해결하는 프로그램을 작성하는 것은 배우기가 쉽지 않다. 특히 강의를 듣고 교안을 학습한 후에 프로그래밍 과제를 스스로 하기가 매우 어렵다. 이에 따라 대학의 전통적인 대 면 강의에서는 이론 강의와 함께 대응하는 프로그래밍 실습 과목을 통해 이 문제를 부분적으로 해결하고 있다. 정보기술 전문 교육기관의 프로그래밍 과정에서는 잘 설계된 실무형 프로젝트를 교육 기간 동안 계속 수행하게

함으로서 이 문제를 해결하고 있다.

- (2) 프로그래밍을 배우는 것이 처음에 어렵고 지나면서 쉬워진다.

프로그래밍 기술을 배우는 것이 그렇게 어렵지는 않다. 그러나 국내 학생들과 같이 암기 위주의 학습에 익숙하고 논리적 사고 훈련이 부족한 학습자는 프로그래밍을 시작하기가 어렵다. 이러한 학습자가 초급 프로그래밍 과목을 수강하는 경우가 대부분이다. 초보자가 프로그래밍을 시작하도록 도와주는 보편적인 방법중의 하나는 주어진 프로그램을 그대로 작성하여 실행시키게 하거나 조금 수정하여 실행시키게 하는 것이다. 이와 같은 실습을 통하여 프로그래밍을 할 수 있다는 자신감을 갖도록 한 다음에 실제로 프로그래밍을 많이 해 보도록 하는 것이다. 학습자는 반복적인 프로그래밍 연습을 통해 프로그래밍에 대한 흥미도 생기고 익숙해져 점점 배우기가 쉬워진다.

- (3) 프로그래밍을 하기 위한 플랫폼과 소프트웨어가 필수적이다.

프로그래밍 실습과 연습을 하기 위해서는 플랫폼과 컴파일러와 같은 소프트웨어가 반드시 구비되어야 한다. 학습자가 학교에서는 물론이고 집에서 같은 환경에서 프로그래밍을 할 수 있어야 한다. 특히 초보자가 쉽게 프로그래밍을 시작할 수 있도록 소프트웨어가 간단하고 배우기가 쉬워야 한다. 또한 소프트웨어를 구입하는 비용이 전혀 부담스럽지 않아야 한다. 이러한 요구사항을 만족시키기 위해 대학의 프로그래밍 과목에서는 프로그램의 컴파일과 실행 기능이 추가된 공개 소프트웨어인 텍스트 편집기를 사용하고 있다.

자바 프로그래밍에 관한 웹 기반의 기존 교안들은 위와 같은 특성을 반영하고 있으나 여전히 다음과 같은 문제점들을 가지고 있다:

- (1) 프로그래밍의 이론과 실습의 분리

프로그래밍 강의의 웹 기반 교안은 보통 프로그래밍 예제들의 컴파일과 실행을 하기 위해 사용

되는 프로그래밍 환경과 분리되어 있다. 그 결과로 교수자와 학습자는 적어도 두 가지의 다른 환경들을 다루어야 한다. 이는 초보 학습자가 프로그래밍 실습을 시작하기가 어려워 계속 배우는 데 장애가 될 수도 있다.

(2) 정적인 프로그래밍 예제들

웹 기반 교안 내의 프로그램 예제들은 관련 주제만을 포함하는 완전한 실행 가능한 프로그램의 일부이다. 학습자는 제시된 프로그램 코드를 수정하여 컴파일하고 실행할 수 없다. 따라서 학습자는 제시된 프로그램을 이해할 수는 있으나 직접 실행하여 결과를 확인할 수 없다.

(3) 하드웨어/소프트웨어 호환성 문제

프로그래밍 교과목에서 많은 문제들은 학습자가 집에서 프로그래밍 실습과 연습을 하는 것을 선호함에 따라 일어난다. 그 결과로 학습자들은 다른 종류의 컴파일러를 가진 다양한 플랫폼 위에서 프로그램 개발도구들을 다룬다. 집에 있는 컴퓨터에서 작성하고 시험한 프로그램들과 결과들이 여러 가지의 이유로 학교의 시험 환경에서 실행되지 않을 수 있다.

기존 교안들의 문제점을 해결하고 웹 기반 자바 프로그래밍 강의의 교육 효과를 극대화하기 위한 교안 개발 방법은 다음과 같다:

(1) 주어진 문제를 프로그래밍으로 해결하는 방법을 포함시킨다.

해당 강의 주제에 적합한 문제를 프로그래밍으로 해결하는 과정을 단계적으로 자세하게 설명함으로써 문제 해결 능력을 배우게 한다. 단순히 구현 결과만을 제시하는 대신 주어진 문제의 요구 사항 분석과 이에 따른 대안들을 검토한 후에 선택한 설계에 대해 상세히 기술한다.

(2) 프로그래밍 실습 관련 내용을 포함시키고 실습 환경과 연동되게 한다.

프로그래밍을 배우는 것은 실습과 프로그래밍

연습을 통해 완성된다. 웹 기반 강의에서는 실습도 인터넷과 웹을 통해 이루어져야 하므로 교안에 실습 관련 내용이 포함되어야 한다. 특히 주어진 프로그래밍 예제 프로그램을 바로 실행시켜 결과를 확인하거나 수정할 수 있는 실습 환경과 바로 연동될 수 있어야 한다. 실습 환경은 비싼 소프트웨어를 사서 설치하는 대신에 무료로 공개된 소프트웨어를 내려 받아 설치하도록 하여 학교와 집에서 동일해야 한다.

(3) 프로그래밍 과제를 스스로 하기 쉽게 개발하여 포함시킨다.

프로그래밍을 잘 하기 위해서는 실제로 프로그래밍을 많이 해 보아야 한다. 그러나 초보자가 교안을 학습한 후 프로그래밍을 스스로의 힘으로 하기가 쉽지 않다. 무엇보다 중요한 것이 초보자가 프로그래밍을 할 수 있다는 자신감을 가지게 하는 것이다. 이를 위해 과제를 초보자가 프로그래밍을 처음부터 시작하도록 도와주기 위해 과제물과 관련된 내용에 대한 자세한 설명을 담고 있는 파일과 대부분이 채워져 있는 골격 프로그램의 일부 미완성 부분을 완성시키도록 하는 유형으로 개발하여 포함시킨다. 이렇게 함으로서 초보자가 프로그램의 전체적인 구조를 알게 하고 구체적으로 해야 될 작업들이 프로그램 내에서 부분적으로 어떻게 구현되지를 배울 수 있다.

(4) 단순하고 사용하기 쉽게 개발한다.

수강생이 웹 기반 강의 교안을 학습할 때 프로그램을 많이 내려 받아 설치해야 되는 등 복잡한 일을 수행해야 한다면 수강생의 학습 의욕을 떨어뜨릴 수도 있고 최악의 경우에는 학습 진행이 이루어지지 않을 수도 있다. 또한 교안을 볼 때 도움말을 수시로 참조해야 하는 등 사용법이 복잡하다면 이 또한 학습 효과를 반감시킬 수 있다. 따라서 교안을 단순하고 사용하기가 쉽게 개발하는 것이 바람직하다.

(5) 좋은 디자인을 가지게 개발한다.

수강생이 장시간 교안을 보아야 하므로 교안이

좋은 디자인을 갖도록 해야 한다. 화면 구성이 잘 정돈되어야 하고 화면마다 적절한 양의 정보가 있어야 한다. 또한 느낌이 좋고 편안한 색을 써서 눈의 피로감을 줄여 주어야 한다.

3. 교안의 개발

웹 기반 교안의 개발은 교과목의 특성 및 사용자의 요구와 환경을 고려하여 세심하게 이루어져야 한다. 자바 프로그래밍 교안은 전장에서 언급한 이러한 고려 사항들을 반영하고 개발 방법에 따라 개발한다. 또한 개발하는 교안이 국내 대학교과목으로 사용되므로 한 학기 강의라는 요건을 갖추어야 한다. 보통 국내 대학의 한 학기는 16주이며 학점은 3학점이다. 따라서 본 교안은 3학점 16주 강의를 기본으로 설계한다. 이에 따라 교안은 자바 프로그래밍의 초급 과정에서 다루는 기본 내용들을 포함한다.

3.1 교안의 구조

교안의 구조는 가능한 단순하게 설계한다. 교안은 우리에게 익숙하고 네비게이션(navigation)이 쉬운 계층적 구조를 갖는다. 시작 페이지는 환영 메시지와 함께 학습자가 교안의 효과적인 학습을 위해 필요한 사항들인 과목 개요, 금주 교안, 숙제, 도움말, 참조 자료, 설치 정보 등을 포함한다. <그림 1>은 이러한 내용들을 포함하는 시작 페이지를 보여 준다. 과목 개요는 과목에 대해 전반적으로 소개하는 내용으로서 학습자가 언제든지 쉽게 참조할 수 있도록 한다. 교안의 주 내용은 주 별로 나누어 금주 교안에 포함시킨다. 숙제는 해당 주의 강의 내용에 대한 프로그래밍 과제를 포함한다.

도움말은 교안을 보는 것을 도와 주기 위해 교안의 메뉴와 아이콘에 대해 설명한다. 참조 자료는 프로그래밍에 필수적인 자바 클래스 라이브러리의 API를 필요시 참조할 수 있도록 해당 웹 페이지에 연결되는 연결점(link)을 포함한다. 설치 정보는 학습자가 프로그래밍 실습 환경을 설치하는 과정을 자세히 설명한다. 먼저 자바 클래스 라이

브러리인 JDK를 설치하는 방법을 설명하고 다음으로 공개된 간단한 자바 프로그래밍 소프트웨어인 TextPad 설치 방법을 설명한다. <그림 2>는 실습 소프트웨어인 TextPad를 설치하는 방법을 설명하는 내용을 캡처한 화면이다.

교안의 주 내용인 금주 교안은 면 대 면 강의의 전형적인 순서를 따르고 프로그래밍 과목의 특성을 반영하여 예제 프로그램을 포함시킨다. 그밖에 전주 강의 내용에 대한 퀴즈와 교안 학습 후의 교안 평가를 위한 설문 조사는 웹 기반 강의 운영 플랫폼에서 제공하는 기능을 활용하므로 이를 알리는 공지 사항을 포함한다. 금주 교안에는 전주 강의 요약, 퀴즈 공지 사항, 금주 강의 내용, 예제 프로그램, 과로 나뉘어진 소주제들, 강의 내용 요약, 설문 조사 공지 사항, FAQ와 참고 문헌

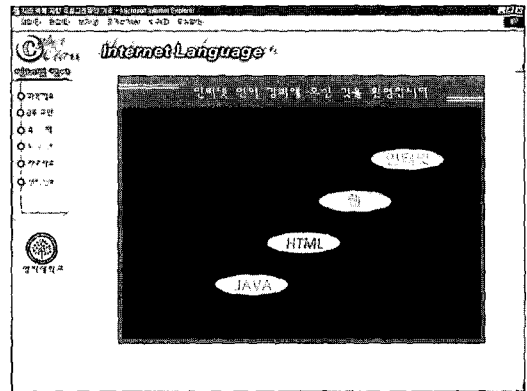


그림 1. 교안의 시작 화면

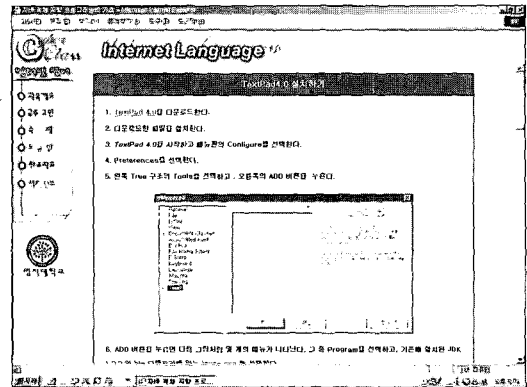


그림 2. 실습 소프트웨어 설치 방법 설명 화면

등이 있다. FAQ는 강의 내용 중 예상되는 질문들을 모아 놓은 것이다. 참고 문헌은 강의 내용과 관련된 자료 등을 포함한다. <그림 3>는 이러한 내용을 보여 주는 한 주 교안의 초기 화면으로서 왼쪽 메뉴의 금주 교안을 누르면 나온다.

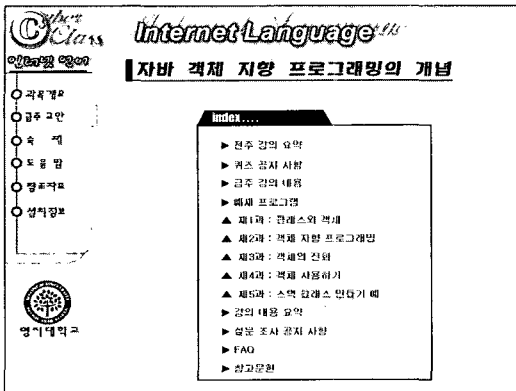


그림 3. 한 주 교안의 초기 화면

금주 강의 내용은 해당 주의 강의 내용을 인터넷을 통해 접속 시에 내려받는 시간이 긴 경우에 대비하여 텍스트, 음성, 비디오 중 선택할 수 있게 한다. 음성과 비디오는 교안의 단조로움을 피하고 학습자의 관심을 유도하고 주의를 환기시키기 위해 사용한다. <그림 4>은 세 가지의 다른 매체로 제공되는 금주 강의 내용 화면을 보여 준다.

연필 모양의 아이콘을 누르면 강의 내용을 텍스트로 보여 준다. TV 모양의 큰 아이콘을 누르면 강의 내용을 비디오로 보여 준다. 마이크 모

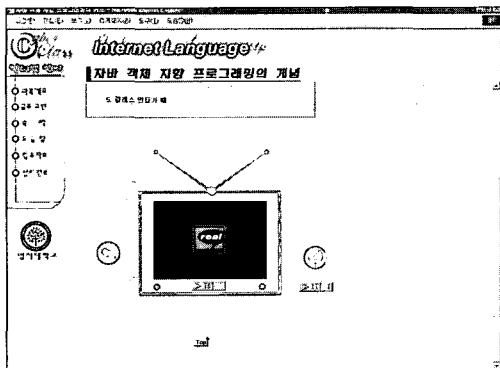


그림 4. 금주 강의 내용 화면

양의 아이콘을 누르면 강의 내용을 음성으로 들려준다.

예제 프로그램은 해당 주의 교안 내용을 이해하면 읽을 수 있고 작성할 수 있는 대표적인 프로그램의 코드와 실행 결과를 보여 준다. 제시된 프로그램에 대해 보기(view) 기능과 내려 받기(down) 기능이 있다. 내려 받기 기능은 프로그램을 내려 받아 현재 위치에서 연동된 자바 개발 소프트웨어인 TextPad를 통해 열어 그대로 혹은 수정하여 컴파일하고 실행해 볼 수 있도록 한다. <그림 5>은 한 주 교안의 예제 프로그램을 내려 받아 현재 위치에서 TextPad를 통해 열린 화면을 보여 준다. 따라서 학습자는 교안을 학습하는 중에 주어진 프로그램을 가지고 실습을 할 수 있다.

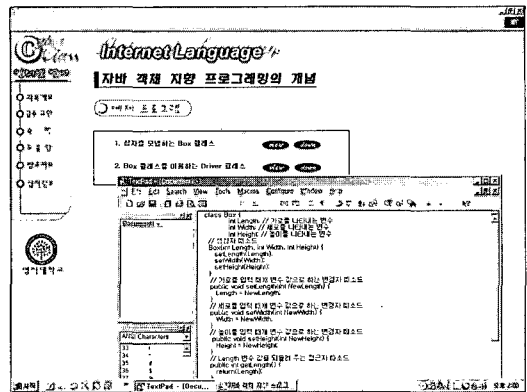


그림 5. 예제 프로그램의 내려 받기 화면

금주 교안의 주 내용인 각과에 들어가는 내용은 면 대 면 강의를 듣는 것과 유사한 교육 효과를 볼 수 있도록 설계해야 한다. 이를 위한 구체적인 방안은 다음과 같다:

- (1) 면 대 면 강의에서는 강의 노트의 내용을 부연해서 설명할 수 있으나 웹 기반 강의에서는 그렇게 할 수 없으므로 교안에 설명까지 포함해야 한다. 따라서 강의 노트보다 자세히 작성해야 한다. 그러나 웹을 통해 교안을 오랫동안 집중해서 보기가 쉽지 않음으로 교안의 양이 너무 많은 것은 적절하

지 않다. 이에 따라 과의 수를 5~7개로 하고 각과에 들어가는 내용은 가능한 다섯 페이지를 넘지 않도록 한다. <그림 6>는 다섯 페이지로 이루어진 한과의 첫 페이지 화면을 보여 준다.

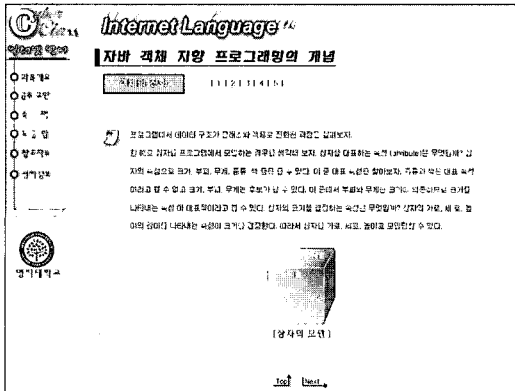


그림 6. 교안의 내용 화면

(2) 주요 개념을 설명할 때 신세대들에 익숙한 만화 같은 애니메이션을 첨가하여 학습자의 흥미를 유발하고 개념의 정확한 이해를 돕는다. 애니메이션에 사용한 메타포는 교안의 학습자와 같은 연령대의 사이버 남녀 대학생들이 교안을 학습한 후 의문 나는 사항들을 대화를 통해 서로 묻고 대답하면서 이해하는 형태를 취한다. <그림 7>는 이러한 애니메이션의 한 화면을 캡처한 것이다. 또한 데모가 적절한 경우 데모시 화

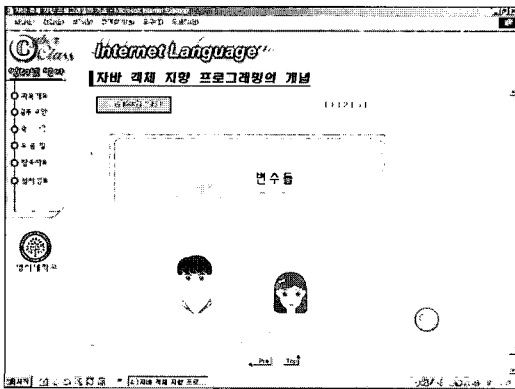


그림 7. 교안의 애니메이션 캡처 화면

면을 캡처하여 교안에 포함시킨다.

(3) 주어진 문제를 어떻게 프로그램하는지를 단계별로 자세하게 설명한다. 특히 프로그램 설계 부분에 중점을 두어 여러 대안들을 제시하고 그 중 하나를 선택하는 이유를 제시한다. 다음으로 선택된 프로그램 설계에 따라 어떻게 구현하는지를 상세하게 설명한다. <그림 8>은 대표적인 데이터 구조중의 하나인 스택을 모델하는 클래스를 프로그램하는 단계 중 문제 분석과 설계 대안들을 고려하는 과정을 보여준다. <그림 9>은 대안들 중 선택된 프로그램 설계를 보여 준다. <그림 10>는 설계에 따라 구현된 프로그램을 보여 준다.

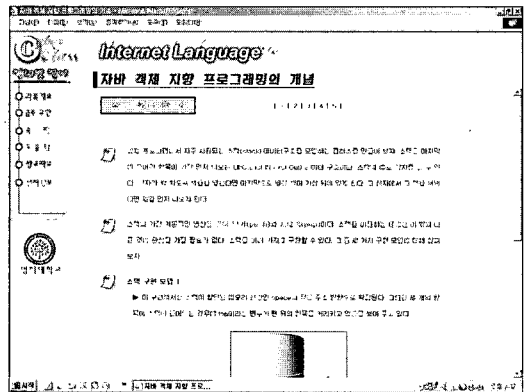


그림 8. 문제 분석 및 설계 대안 제시 화면

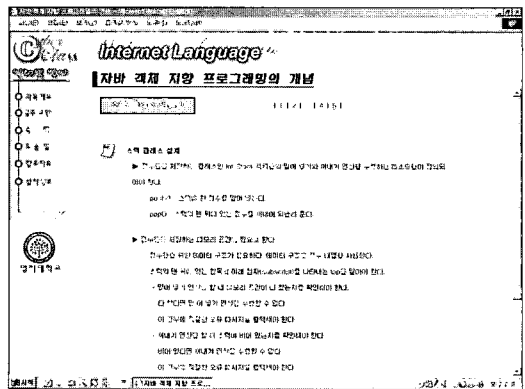


그림 9. 선택된 프로그램 설계 화면

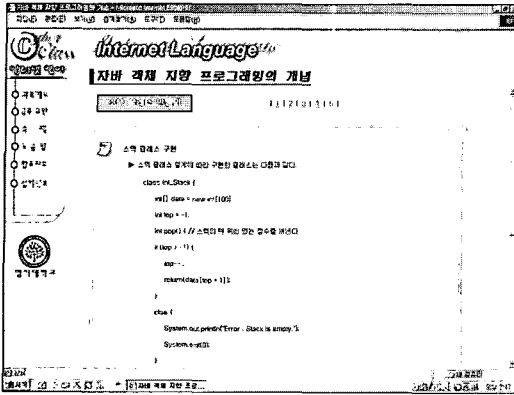


그림 10. 설계에 따라 구현된 프로그램 화면

(4) 실제 강의 시 학생들의 주의를 환기시키기 위해 강사가 수시로 질문을 던지는 것이 효과적이다. 수강생들이 수업을 들을 때 소홀하다가도 강사가 질문을 던지면 자기가 대답할 수도 있으므로 모두 긴장하게 되고 답을 찾으려고 나름대로 노력하게 된다. 이런 효과를 웹 기반 강의에서도 얻기 위해 교안의 각 페이지를 가능하면 지금까지 강의 내용을 제대로 이해했는지 확인하는 질문으로 끝나게 하는 것도 좋은 방법이다. 정답을 통해 재설명하는 효과를 거두게 한다. <그림 11>은 교안 내에서 질문에 대한 답을 누른 경우의 화면을 보여 준다.

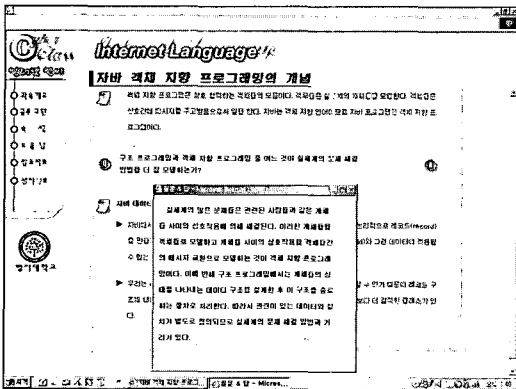


그림 11. 교안의 질문과 답 화면

프로그램 과제는 초보자가 쉽게 할 수 있도록 설계한다. 해당 주의 강의 내용을 이해하면 프로그램 할 수 있는 문제를 선택한다. 문제에 대한 프로그램을 일부 미완성 부분을 가진 형태로 작성하여 포함시킨다. 미완성 부분을 채우는 것을 도와 주기 위해 문제와 미완성 프로그램 코드의 각 부분에 대한 자세한 설명을 담고 있는 정보 파일을 만든다. 학습자가 이 정보 파일을 읽으면 교안에서 배운 강의 내용이 실제 문제를 프로그램 하는 데 어떻게 적용되는지를 알 수 있고 미완성 부분도 채워 넣을 수 있도록 한다. 또한 학습자가 강의 내용 관련 부분에 집중하도록 프로그램내의 입출력 관련 부분은 완성시킨다. 이렇게 함으로서 학습자가 프로그램을 정확하게 완성하면 출력 결과는 문제의 정답과 일치한다. 따라서 학습자가 완성한 프로그램의 정확성을 스스로 확인할 수 있고 자동 채점 프로그램을 개발하여 채점할 수 있다. 과제물에 포함된 파일은 다음과 같다:

- 정보 파일: 문제와 관련된 내용과 제공된 프로그램에 대한 설명
- 프로그램 파일: 문제에 대한 미완성 프로그램
- 정답 파일: 프로그램의 실행 결과
- 예제 프로그램 파일: 완전한 예제 프로그램
- 예제 프로그램 설명 파일: 예제 프로그램의 설명

한 주의 프로그램 과제물과 관련된 정보 파일과 프로그램 파일의 일부가 각각 부록 1과 부록 2에 제시된다. 프로그램 파일에 있는 프로그램은 바로 컴파일하고 실행할 수 있다. 제공된 프로그램의 일부 내용은 완성되어 있고 일부 내용은 미완성 상태로 있어 채워 넣어야 한다. 완성되어 있는 부분을 이해하면 어렵지 않게 미완성 부분을 채워 넣을 수 있게 설계되어 있다.

시험은 퀴즈, 중간 시험과 기말 시험으로 구성된다. 퀴즈는 전 주 강의 내용에 대해 매 주 보게 하여 복습을 하게 하고 교안을 매 주 습관적으로 학습하도록 유도한다. 시험 문제는 이론적인 내용뿐만이 아니라 프로그래밍 능력을 평가할 수 있도록

록 설계한다. 예를 들면 1부터 100까지의 정수들의 합을 구하는 프로그램들 중 틀린 것을 찾는 것과 같은 유형의 문제이다. 시험은 사이버 강의 운영 플랫폼을 사용하여 실시되므로 플랫폼의 기능 제약에 따라 문제는 객관식으로 만든다.

설문 조사는 학습자가 교안을 학습한 후의 교안 평가 의견을 수렴하여 교안을 개선하기 위한 목적이다. 설문 내용은 매주 강의 안의 내용에 대한 이해도와 이해하기 어려운 부분, 교안을 본 시간, 교안 내용 설명의 적절성, 교안내의 멀티미디어 활용의 적절성, 교안을 실시간으로 보는데 느낀 어려움 등이다.

3.2 교안의 화면 구성

교안의 화면 구성은 가능한 단순하게 설계한다. 이렇게 함으로서 학습자가 교안 사용법을 배우지 않고도 쉽게 교안을 볼 수 있게 한다. 교안의 화면은 크게 세 부분으로 나눈다. 윗 부분은 표제 영역으로서 사이버 강의란 사실과 교과목명 등을 알리는 고정된 텍스트가 있다. 왼쪽 부분은 메뉴 영역으로서 고정된 교안의 메뉴와 로고가 있다. 메뉴는 과목 개요, 금주 교안, 숙제, 도움말, 참조 자료와 설치 정보와 같은 6 개의 메뉴 항목들을 포함한다. 이는 <그림 1>의 왼쪽에 보여져 있다. 오른쪽 부분은 변경될 수 있는 주 영역으로서 화면의 대부분을 차지하고 왼쪽에 있는 특정 메뉴 항목을 누르면 그에 해당하는 내용이 나온다. 예를 들면 <그림 3>에 보여진 대로 금주 교안 메뉴를 누르면 오른쪽 화면에 1주 분의 강의 내용이 나온다.

주 영역에 있는 특정 항목을 누르면 그 영역이 그 항목에 대한 내용으로 교체되거나 별도의 창이나 대화창이 뜬다. <그림 3>의 화면에서 '제 3과: 객체의 진화'를 누르면 <그림 6>의 화면이 나온다. 교안의 모든 내용은 대부분의 경우 마우스를 3 번 누르면 볼 수 있고 최대 4 번 누르면 볼 수 있다.

화면 구성을 일관성이 있고 정돈되게 하여 단기간 내에 학습자가 교안의 구성에 익숙하게 한다. 또한 화면 공간을 넉넉하게 사용하도록 설계

한다. 모든 화면에 충분한 여백을 두어 학습자가 지루함을 느끼지 않고 여유를 가지고 교안을 볼 수 있도록 한다. 또한 느낌이 좋고 편안한 색을 사용하여 좋은 기분으로 학습하게 한다. <그림 6>는 이러한 화면의 한 좋은 예이다.

3.3 교안의 제작

교안의 주 콘텐츠(금주 교안)는 자바 프로그래밍 과목을 가르친 경험이 있는 세 명의 교수들에 의해 개발되었다. 교안을 작성한 후 두 명의 대학 원생과 두 명의 학부생이 교안의 내용에 대해 학습자 입장에서 검토하였다.

이러한 검토 의견을 반영하여 교안을 수정하고 보완하였다. 프로그래밍 과목물은 면 대 면 강의에서 개발한 내용을 보완하여 만들었다. 금주 강의 내용은 스튜디오에서 녹화하여 인코더를 사용하여 비디오/오디오 파일로 변환하였다. 시작 페이지의 환영 화면과 금주 교안내의 애니메이션은 시나리오에 따라 본 대학 교내 방송국의 성우들이 녹음하고 포토샵과 플래쉬를 사용하여 제작하였다. 교안에 사용된 매체는 대부분 텍스트이나 이미지, 그래픽스, 비디오, 오디오와 애니메이션도 적지 않게 포함되었다. 매체별 제작 도구는 <표 1>과 같다. 교안은 외부 용역 업체에 위탁하여 제작되었다. 웹 디자이너, 웹 프로그래머, 응용 프로그래머가 각 1명씩 참여하였다.

<표 1> 매체별 제작 도구

매체	제작 도구	포맷
텍스트	나모 웹에디터 3.0	html
이미지, 그래픽스	포토샵 5.0	gif/jpg
음성, 동영상	리얼 미디어 인코더	rm
애니메이션	플래쉬 5.0	fla

교안 외에 채점 프로그램을 개발하였다. 본 과목은 프로그래밍 과목이므로 이론 강의뿐만이 아니라 실제 문제를 프로그래밍 하는 과제물이 과목을 성공적으로 이수하는데 필수적이다. 이에 따라 대부분의 프로그래밍 과목에서는 많은 과제물을 내 주고 있다. 웹 기반 강의에서는 수강생들이 수

업에 출석하지도 않으므로 강의 수강에 소홀할 수 있어 더욱 과제물이 중요하다.

국내 대학 여건상 과목에서 많은 조교를 쓸 수 없는 상황이다. 따라서 과제물을 자동으로 채점할 수 있는 프로그램의 개발이 필요하다. 과제물은 미완성 프로그램에 입출력 부분을 확정하여 포함 시킨 형태로 개발되었기 때문에 수강생이 정확하게 프로그램을 완성하면 정답과 일치해야 한다. 채점 프로그램은 이러한 과제물의 특성을 이용하여 수강생이 제출한 과제물의 결과와 정답을 비교하여 일치 여부를 알려 준다. 채점 프로그램을 이용하여 프로그램의 정확성에 대해 평가하고 나머지 평가는 수작업으로 한다.

4. 교안의 평가

본 교안은 본 대학 전자정보통신공학부 3학년 학생 24명과 정보통신 사이버대학 협의회에 소속된 대학의 학생 29명을 합쳐 총 53명을 대상으로 사이버 강의로 한 학기동안 운영되었다. 타 대학생들의 소속은 정보기술 관련학과가 15명이고 나머지는 타 전공 학과이다. 따라서 타 대학 수강생들은 본 대학 수강생들에 비해 학생 구성이 다소 이질적이라고 볼 수 있다.

수강생들은 매 주 교안을 학습한 후 학습 내용에 대한 이해도를 파악하고 교안의 내용과 접근성을 개선하기 위해 <표 2>와 같은 내용의 설문 조사를 하였다. 한 주 교안의 설문 조사는 53명중 44명이 응답하였고 그 결과는 <표 3>과 같다. 교안의 내용을 반 이상 이해한 수강생의 비율은 77%이므로 대부분의 수강생들이 교안을 이해했음을 알 수 있다. 또한 교안의 내용 설명이 적절하다는 수강생의 비율이 63.6%이고 그 이상이라는 수강생의 비율이 13.6%이므로 교안이 거의 모든 수강생들이 이해하기 쉽게 작성되었음을 알 수 있다. 교안을 보는 데 걸린 시간이 1~4 시간이라는 수강생의 비율이 61.4%이고 4시간 이상이라는 수강생의 비율이 34.1%이므로 교안의 한 주 분량이 적절했다는 것을 확인할 수 있다. 교안의 멀티미디어 활용 정도도 적절하거나 그 이상인

<표 2> 한 주 교안의 설문 조사 내용

문항 1. 이번 주 강의 내용을 어느 정도 이해했는가? (1) 거의 모두 이해하지 못했다. (2) 반 보다 많이 이해하지 못했다. (3) 반 정도 이해했다. (4) 반 보다 많이 이해했다. (5) 거의 모두 이해했다.
문항 2. 이번 주 교안의 내용 설명이 적절한가? (1) 매우 적절치 못하다. (2) 적절치 못하다. (3) 적절하다. (4) 만족스럽다. (5) 매우 만족스럽다.
문항 3. 이번 주 교안을 보는데 걸린 시간은? (1) 1시간 미만 (2) 1~2시간 (3) 2~3시간 (4) 3~4시간 (5) 4시간 이상
문항 4. 이번 주 교안의 멀티미디어 활용은 적절한가? (1) 더 많이 활용되어야 한다. (2) 조금 더 활용되어야 한다. (3) 적절하다. (4) 조금 더 활용도를 줄여야 한다. (5) 많이 줄여야 한다.
문항 5. 이번 주 교안을 실시간으로 보는 데 어려움은 없었는가? (1) 다음 페이지로 넘어가는 데 오랜 시간이 걸렸다. (2) 애니메이션을 보는 데 많은 시간이 걸렸다. (3) 음성의 질이 낮았다. (4) 그림을 다운받는데 시간이 많이 걸렸다. (5) 어려움이 없었다.

<표 3> 한 주 교안의 설문 조사 결과

(응답률 %)

문항 \ 응답	(1)	(2)	(3)	(4)	(5)
1	4.5	18.2	36.4	20.4	20.4
2	0	22.7	63.6	9.1	4.5
3	4.5	22.7	27.3	11.4	34.1
4	11.4	9.1	65.9	13.6	0
5	15.9	13.6	4.5	6.8	59.1

라는 수강생의 비율이 79.5%이므로 적절했음을 알 수 있다. 교안을 접근하는 데 어려움이 없었다는 수강생의 비율이 59.1%이므로 교안이 접근성도 좋았음을 알 수 있다.

본 교안을 사용하여 한 학기 강의를 마친 후 수강생들을 대상으로 강의 전반에 관한 설문 조사를 본 대학 가상교육원 주관으로 실시하였다. 이 설문 문항중 교안의 내용과 관련된 문항은 <표 4>과 같다. 이 설문 조사에 대한 총 수강생 53명중 19

명이 응답한 결과는 <표 5>와 같다. 제시된 수업 목표와 수업 내용이 일치하였다는 질문에 대해 모든 응답자가 보통 이상이라는 긍정적인 응답을 하였다. 수업 내용이 체계적이고 합리적으로 구성되고 전개되었다는 질문에 대해서는 36.8%는 보통이고 52.6%는 보통보다 낫다는 응답을 하였다. 강좌의 난이도의 적절성에 대한 질문에 대해서는 57.9%의 응답자가 보통 이상이라는 긍정적인 응답을 하였다. 교안의 주요 콘텐츠중의 하나인 프로그래밍 과제물의 제시와 평가에 대한 질문에 대해서는 68.5%의 응답자가 보통 이상이라는 긍정적인 응답을 하였다. 교안 내용의 이해 여부를 묻는 질문에 대해서는 31.6%는 보통이고 42.1%는 보통보다 낫다는 응답을 하였다. 따라서 주 별 설문 조사 결과와 기말 설문 조사 결과를 보면 본 교안이 학습자들에게 좋은 평가를 받았음을 알 수 있다.

또한 본 교안은 정보통신부의 정보통신 사이버 대학 지원사업의 지원을 받아 개발이 완료된 후 콘텐츠 전문가들에게 평가를 받았다. 평가 항목은 콘텐츠의 우수성, 수강생들의 접근성, 수강생과의 상호작용성과 학습 효과 평가 방법 등이다. 평가

결과 본 교안은 개발된 총 26개 교안중 우수 교안으로 선정된 4개의 교안중 하나였다. 따라서 본 교안은 콘텐츠 전문가 그룹에게도 우수하게 평가되었다.

5. 결론

본 논문은 기개발된 웹 기반의 자바 프로그래밍 교안들이 프로그래밍 기법에 대한 내용 부족과 프로그래밍 실습 환경과 분리되어 있는 문제점들을 해결하기 위해 프로그래밍 기법과 실습을 강조한 교안 개발 사례를 다루었다. 이를 위해 개발 교과목의 특성을 분석하고 교수자와 학습자의 요구사항을 반영하여 프로그래밍 능력을 배우도록 하기 위한 개발 방법을 제시하고 그에 따라 기획하고 설계하여 개발된 교안을 소개하였다. 개발된 교안은 대학생들을 대상으로 운영한 웹 기반 강의에서 좋은 평가를 받았으며 콘텐츠 전문가들에게도 우수 교안으로 평가받았다. 향후 프로그래밍 과목의 사이버 강의를 계획하고 있는 교수자들에게 하나의 참고 사례로서 활용되어 사이버 강의의 확산 및 발전에 기여하기를 기대한다.

개발된 교안을 이용한 웹 기반 강의의 교육 효과는 기대에 미치지 못하였다. 특히 중도에 포기하는 수강생들이 적지 않았다. 이는 강의 운영을 통해 확인한 바에 의하면 프로그래밍을 처음 배우는 초보자가 적지 않았고 수강생들의 학습 의지가 부족하여 쉽게 배우려고 하였으나 학습 요구 양이 너무 많았기 때문인 것으로 사료된다. 프로그래밍 과목은 역시 초보 학습자에게 학습 동기를 부여하기도 어렵고 프로그래밍을 배우게 하기가 쉽지 않다는 것을 확인하였다. 따라서 웹 기반의 프로그래밍 과목의 교육 효과를 높이기 위한 교안 개발 방법론에 대한 지속적인 연구가 필요하다.

<표 4> 기말 설문 조사 내용의 일부

문항 1. 제시된 수업 목표와 수업 내용이 일치하였습니까?
문항 2. 전체적인 수업 내용이 체계적이고 합리적으로 구성되고 전개되었습니까?
문항 3. 학생이 수강하는 강좌의 난이도는 적절하다고 생각합니까?
문항 4. 과제물의 제시와 평가는 적절하였습니까?
문항 5. 이 과목의 내용을 전반적으로 잘 이해하게 되었습니까?

<표 5> 기말 설문 조사 결과

응답 문항	아주 그렇다	대부분 그렇다	보통이다	그렇지 않다	아주 그렇지 않다
1	5.3	68.4	26.3	0	0
2	26.3	26.3	36.8	10.5	0
3	0	21.1	36.8	31.6	10.5
4	5.3	31.6	31.6	26.3	5.3
5	5.3	36.8	31.6	26.3	0

[참고문헌]

1. 강숙희 (1999), 정보통신 기술을 활용한 교수-학습 사례 연구, 정보과학회지, 17(9), 12-18.
2. 김동식, 서삼준 (2000), 효율적인 사이버 강의

- 를 위한 전자회로 원리 이해용 자바 애플릿의 개발, 공학교육 연구, 3(2), 38-43.
3. 이종연 (1999), 사례: 삼성 사이버 교육 현황 및 발전 방향, 정보과학회지, 17(9), 47-51.
 4. 이충기 (2001), 자바 프로그래밍 교육에서 웹 기반 강의와 면대면 강의의 학습 효과 비교 분석 연구, 공학교육 연구, 4(2), 3-10.
 5. 허원 (2001), Java Beans 기술을 이용한 효과적 공학 교육용 Applet 개발에 관한 연구, 공학교육 연구, 4(2), 27-34.
 6. Ausserhofer, A. (1999a), "Web-Based Teaching and Learning: A Panacea?," IEEE Communications Magazine, March 1999.
 7. Ausserhofer, A. (1999b), "S.E.A.L. - A New Approach in Teaching Computing." Proceedings of the IASTED International Conference, May 1999.
 8. McCormack, C.; Jones, D. (1998), Building a Web-Based Education System ; John Wiley & Sons : New York, U.S.A. ; p 446.
 9. Ross, J. (2000), "An Exploratory Analysis of Student Achievement Comparing A Web-Based And A Conventional Course Learning Environment," <http://schoolnet.ca/mlg/sites/acolccael/en/report.html>

감사의 글
본 연구는 정보통신 사이버대학 1999년도 지원 사업의 연구 지원으로 수행되었습니다.

부록 1 프로그램 과제 설명 파일

프로그래밍 숙제 4

학습제목 : 객체(object)에 대한 소개

□ 학습 목표

- 클래스
- 객체 (instance)
- 참조 (reference)
- 생성자 (constructor)
- 접근자/변경자 메소드
- 추상화된 인터페이스

□ 제공된 파일

- Prog4info.hwp - 실습 설명 파일
- Television.java - 실습 프로그램 파일
- Prog4ans.txt - 이 실습에 대한 정답 파일
- Prog4Example.java - 컴파일하고 실행하기 위한 예제 프로그램
- Prog4Exampenfo.txt - 예제 프로그램에 대한 예비 지식

□ 제출 파일

○ Television.java

□ 유의할 점

클래스들의 이름은 바꾸지 않아야 합니다! 또한 ASCII 모드로 파일을 올려 놓아야 합니다.

□ 예제

시작하기 전에 “lab2Example.nfo”를 읽어보자. 여기에는 매개변수를 넘겨주는 것이 자바에서 어떻게 이루어지는지를 기술한다. 이것을 보여주기 위해 “lab2Example.java”는 당신이 컴파일 할 예제이다. 여러분은 이러한 파일들 중의 어떤 것도 제출할 필요는 없다. 이 실습을 하기 전에 이 파일을 먼저 실행해 보라. 이 과정에 대해서는 지난 주 실습에서 설명했다.

□ 예비지식

객체(Object)는 상태(state)와 행동(behavior)을 가지고 있는 어떤 것이다. 객체가 자신에 관해 알고 있는 정보를 표현한 것이 상태(state)이고 행동(behavior)은 객체가 어떻게 하는가에 대해 알고 있는 것을 나타낸다. 예를 들면, 자동차는 상태(종류, 색깔, 크기 등)와 행동(시동을 건다, 시동을 끄다, 전진한다, 후진한다 등)을 가지고 있다. 당신은 단지 정적인 메소드들에 대해서만 작업해 왔고 자바의 객체에 대해서 아주 조금만을 보아왔다. 이 실습에서 객체가 어떻게 동작하는지를 보게 될 것이다.

□ 개요

우리는 Television 클래스를 만들 것이다. 그리고 나서 이 클래스의 여러 객체들을 만들 것이다. 클래스는 객체들을 기술하는 청사진이다. 예를 들면, int형이 있고, int형 변수들이 있다. 마찬가지로, Television이라는 클래스가 있고, Television이라는 클래스의 많은 객체들이 있다. 이 클래스를 위해서 접근자 메소드(객체 변수에 접근하는 메소드)와 변경자 메소드(객체변수의 값을 변경하는 메소드)가 구현될 것이다.

□ 실습

명심할 점은, 점수를 받기 위해 제출하려는 모든 파일의 맨 위에 헤더 정보를 붙여야 하는 것을 기억해라. 만약 이것을 잊는다면 감점을 받을 것이다. 파일의 맨 아래에 있는 main() 메소드로 가라. 거기에는 다음과 같은 5가지의 변수 선언이 있다:

```
Television televisionOne = null;
Television televisionTwo = null;
Television televisionThree = new Television(true, 5);
Television televisionFour = new Television(false, 60, "LG");
Television televisionFive = new Television(true, 31, "Samsung", "Black");
```

Television은 클래스의 이름이다. televisionOne, televisionTwo, televisionThree, televisionFour, televisionFive는 변수의 이름이고, 객체를 가리키는 참조들(REFERENCES)이다. 첫 번째 줄(Television televisionOne = null;)은 Television 클래스의 객체를 가리킬 수 있는 televisionOne이라고 불리는 참조 변수를 선언한다. 그것은 참조이기 때문에 null로 초기값을 줄 수 있다. 이 파일을 컴파일하고 실행해 보라. 이 클래스의 이름이 Television이라는 것을 유념해라. 그러면 당신이 컴파일 했을 때,

“Television.class”라는 파일을 얻을 수 있을 것이다. 그것이 여러분이 실행하고자하는 파일이다. 이것을 실습 1에서 했던 방법과 마찬가지로 그 파일을 실행시켜 보자. 당신은 다음과 같은 것을 얻어야만 한다:

```
java.lang.NullPointerException
```

여러분의 프로그램으로 돌아와서, 아래쪽에 있는 main() 메소드 근처로 가라. 그리고 다음과 같은 줄을 찾아 보자: `System.out.println("televisionOne is on: "+televisionOne.isOn());`

이 줄이 잘못된 것이다. 이것이 잘못된 이유는 televisionOne이라는 참조 변수의 값이 null이기 때문이다. 이것은 그것이 어떠한 것도 가리키지 않는다는 것을 의미한다. 그것이 어떤 것을 가리키게 할 수 있는 유일한 방법은 그것이 가리키는 새 객체를 만들어 내는 것이다. 이 오류는 다음의 의사코드(pseudocode)에서 nil 포인터를 참조하게 하는 것과 유사하다. 의사코드에서,

```
Head ← nil
```

```
Head^.Data
```

은 작동하지 않을 것이다 : 자바에서

```
televisionOne = null;
```

```
televisionOne.isOn();
```

은 NullPointerException을 만들어 넘으로써 작동되지 않는다. 그러면 다음 줄과 동반되는 주석을 지워라 :

```
System.out.println("televisionOne is on: "+televisionOne.isOn());
```

부록 2 과제 프로그램 파일

```
//INSERT YOUR HEADER HERE AFTER DELETING THIS LINE
import util;
class Television {
. . .

//                               INSTANCE VARIABLES

boolean bPowered      = false;           // Is the TV on or off?
int iChannel           = MINCHANNEL;     // What channel is the TV on?
String strBrandName   = "LG";           // What brand name TV is this?
String strCaseColor   = "Green";        // The color of the casing of this TV
. . .
public Television(boolean bNewPowered, int iNewChannel) {
    setPowered(bNewPowered);
    setChannel(iNewChannel);
}
public Television(boolean bNewPowered, int iNewChannel, String
```

```

        strNewBrandName) {

//// FILL IN HERE
}

    public int channel() {return(iChannel);}
    public String brandName() {

//// Fill IN HERE
//// REMOVE THIS LINE - THIS IS TO ALLOW COMPILATION
        return(new String());
    }

    .
    .
    public void setPowered(boolean bNewPowered) {
        bPowered = bNewPowered;
    }
    public void setBrandName(String strNewBrandName) {
        // make sure String exists
        util.ASSERT(strNewBrandName != null,
            "Error: New Brand Name must have a value.");
    }

//// FILL IN HERE
}

    public void turnOn() { bPowered = true;
}

    public void turnOff() {

//// FILL IN HERE
}

    .
    .

// MAIN
    public static void main(String[] argv) {

//// DO NOT MODIFY THESE!!!
        Television televisionOne = null;
        Television televisionTwo = null;
        Television televisionThree = new Television(true, 5);
        Television televisionFour = new Television(false, 60, "LG");
        Television televisionFive = new Television(true,31,"Samsung", "Black");

        //// This line will attempt to see if televisionOne is on via method
        //// isOn(), but televisionOne does not exist at this point.
        //// Instance televisionOne does not exist since we have not called
        //// new() at this point yet.
        System.out.println("televisionOne is on: "+ televisionOne.isOn());

        //// This calls the Television default constructor

```

```
televisionOne = new Television();

//// This calls the Television constructor with one parameter
//// and the one parameter is a boolean
televisionTwo = new Television(true);

System.out.println("televisionOne\n" + televisionOne.toString());
System.out.println("televisionTwo\n" + televisionTwo.toString());
}
}
```