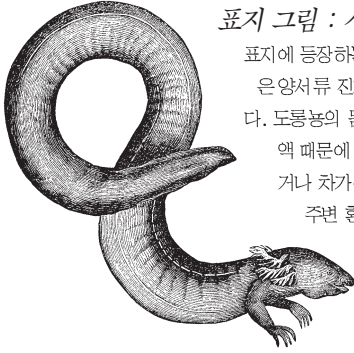

ActionScript

The Definitive Guide



표지 그림 : 사이렌

표지에 등장하는 동물은 북아메리카의 민물에서 서식하는 도롱뇽의 일종인 ‘사이렌(siren)’이다. 도롱뇽은 양서류 진화의 초기 단계에서 다른 양서류(개구리 같은)로부터 분화되었으며, 긴 꼬리를 가지고 있다. 도롱뇽의 몸은 수많은 분비선에서 나오는 체액으로 덮여 있어 항상 습도를 유지할 수 있으며, 그 체액 때문에 몸 표면이 부드럽다. 도롱뇽은 냉혈동물이기 때문에, 주변 환경이 너무 건조하거나 뜨겁거나 차가우면 진흙 속에 몸을 파묻고 신체의 모든 기능을 저하시킨다. 이 때 고치 같은 것을 만들어 주변 환경이 자신의 몸에 적합해질 때까지 그 속에 숨어서 기다린다.

물속에서만 서식하는 사이렌은 뒷다리가 없고 앞다리가 짧으며 아가미가 있다. 성체가 되기 전까지 유충과 같은 형태로 살아가며, 새끼가 물속에서 외부의 영양분으로 살아가기 때문에 도롱뇽 중에서도 매우 원시적인 종에 속한다. 아행 성이며 뱀장어 같이 등지느러미를 이용하여 몸을 좌우로 흔들면서 헤엄을 친다. 이렇게 몸을 움직이면서 작은 벌레나 유충을 잡아먹으며 살아간다. 그리고 식물성 물질은 아가미를 통해 내보낸다.

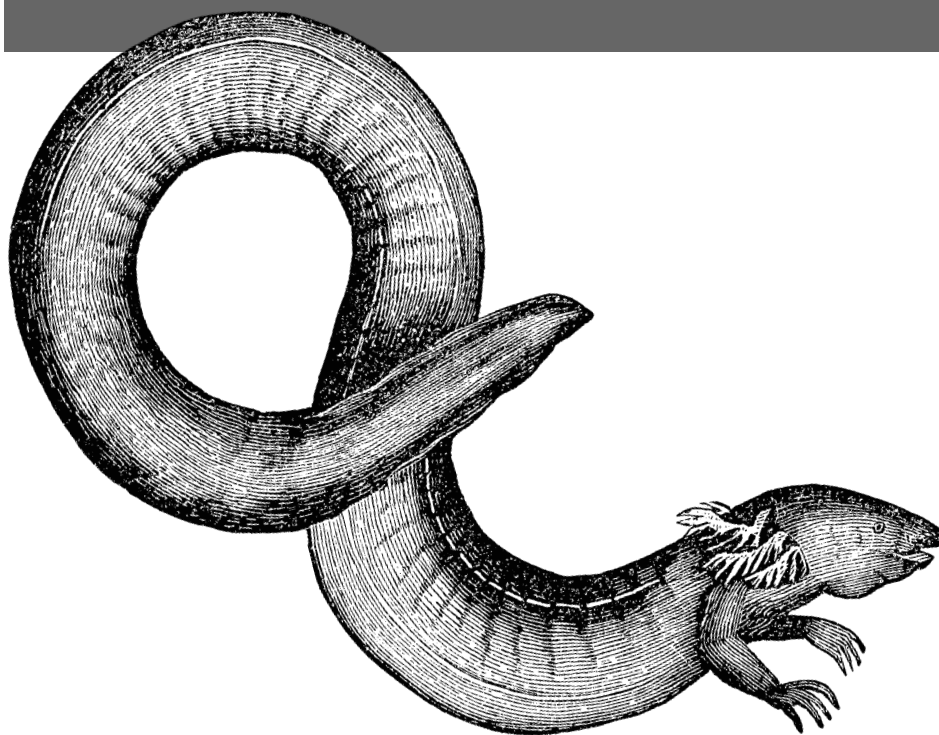
다른 양서류와 마찬가지로 사이렌도 균형 잡힌 생태계에서 매우 중요한 자리를 차지한다. 사이렌은 소형 육식동물에 해당하며, 다른 물고기나 새의 먹이가 된다. 미끌미끌한 피부와 상대적으로 약한 생존력 때문에 민물 생태계에서 중요한 공해의 지표로 쓰인다. 날카롭고 짝짝거리는 소리를 내는 것으로 유명한 ‘사이렌 인터미디어(siren intermedia)’는 원래 북아메리카 전역에 광범위하게 분포해 있던 종이지만, 양어장을 관리하기 위한 용도로 쓰이는 로테논(Rotenone)이라는 화학 물질 때문에 미시간주에서는 거의 그 모습을 찾아볼 수 없다.

©HANBIT Media, Inc., 2001. Authorized translation of the English edition, ©2001 O'Reilly & Associates, Inc. This translation is published and sold by permission of O'Reilly & Associates, Inc., the owner of all rights to publish and sell the same.

이 책의 한국어판 저작권은 오라일리사와의 독점 계약에 의해 한빛미디어(주)에 있습니다. 저작권법에 의해 한국 내에서 보호를 받는 저작물이므로 무단 전재와 무단 복제를 금합니다.

창시자 게리 그로스먼과 함께 배우는

플래시 액션스크립트



O'REILLY®

HB 한빛미디어
Hanbit Media, Inc.

저자 소개

콜린 무크

1995년부터 웹에 대해 연구하고 웹 디자인 및 개발 분야에서 일해왔다. 1997년까지는 소프트웨어(핫메탈 프로 제작사)에서 웹 마스터로 근무했다. 지금은 인터그레이티드커뮤니케이션즈앤드엔터테인먼트(ICE)에서 웹 에반젤리스트로 활동하고 있다. ICE에서는 웹에 대한 글을 쓰고 컨퍼런스에서 강연을 하거나 소니, 리바이스, 노르텔, 에어 캐나다, 휴렛팩커드 같은 회사의 인터랙티브 콘텐츠를 만들고 있다. 경이로운 플래시 작품과 플래시 개발자 사이트를 통해 플래시 커뮤니티에서 유명인사가 되었다. 매크로미디어 역시 웹사이트에서 콜린의 작품을 소개하고 그를 플래시 자문 위원으로 위촉함으로써 그의 능력을 공인하였다. 콜린은 ‘플래시 4 바이블’과 ‘플래시 5 바이블’ 제작에도 참여했다.

역자 소개

서환수

서울과학고등학교와 서울대학교 물리학과를 졸업하고 현재 서울대학교 물리학과 대학원에 재학중이다. 번역서로 『펄로 배우는 알고리즘』(한빛미디어, 2000)과 『제대로 배우는 자바2』(한빛미디어, 2001)가 있다.

플래시 개발자의 마지막 관문, 액션스크립트

웹이 처음 등장했을 무렵에는 대부분의 웹 페이지에서 글자와 간단한 그림 외에는 그다지 볼만한 것이 없었다. 사실 그 시절에는 인터넷 속도도 워낙 느렸기 때문에, 모뎀 사용자들에게는 조금 큰 그림도 상당히 부담스러운 것이었다.

어느새 세월이 흘러 인터넷에 어느 정도 관심이 있는 사람이라면 집에서 ADSL이나 케이블 모뎀과 같은 초고속 인터넷 서비스를 이용하고, 조금이라도 변화한 거리에 나가면 곳곳에 PC방이 줄지어 늘어 서 있는 풍경을 흔히 볼 수 있다. 또한 인터넷 접속 속도가 눈에 띄게 빨라져 이제 웹 페이지에도 글자보다는 그림이 더 많이 사용되고 있는 추세다.

하지만 그것만으로는 사람들의 관심을 끌기에 부족하다. 움직이지 않는 고정된 그림만으로는 뭔가 부족하기 때문이다. 멀티미디어 관련 기능과 사용자와의 상호 작용은 단순히 문자와 그림만으로는 구현하기 힘들다. 물론 예전에도 자바 애플릿처럼 인터넷을 통해 실행할 수 있는 프로그램이 있긴 했지만, 자바 애플릿의 경우 접속 속도가 느린 시스템에서는 사용하기 부담스럽다는 점과 프로그래밍에 익숙해야만 애플릿을 만들 수 있다는 점, 그리고 애플릿이 보통 웹을 통해 사용할 수 있는 ‘애플리케이션’ 성격이 강하다는 점 때문에, 요즘은 자바 애플릿이 그다지 많이 쓰이지 않는다(대신 서버측에서 실행되는 JSP가 인터넷에서 많이 활용되고 있다).

플래시는 작은 용량의 파일로도 충분한 시각, 청각적 효과를 얻을 수 있다는 점, 사용자와의 상호 작용을 매우 간단하게 구현할 수 있다는 점, 또한 무엇보다 만들기 쉽고 플래시 플레이어 자체 용량도 그다지 크지 않아 쉽게 다운로드할 수 있다는 장점 때문에 엄청난 속도로 확산되었다. 요즘 대부분의 사이트에서 플래시 애니메이션이나 플래시를 이용한 인트로 정도는 매우 쉽게 발견할 수 있다.

하지만 플래시에서도 사용자와의 상호 작용을 이루어내기 위해서 어느 정도의 프로그래밍은 필요하다. 또한 조금이라도 무비가 복잡해지면 일일이 그래픽 작업을

직접 하는 것보다는 프로그래밍을 이용해 무비 클립을 제어하는 것이 더 효율적이다. 예를 들어 간단한 퀴즈나 게임을 만들거나 실제에 가까운 움직임을 보여줄 때는 (예를 들면 어떤 물건이 떨어지는 모습을 나타낼 때 더 실감나는 모습을 보여주고 싶다면 중력 가속도와 같은 것도 어느 정도 고려해야 한다) 프로그램을 이용하지 않으면 작업하기 매우 힘들다. 이러한 요구를 해결하기 위해 등장한 것이 바로 액션스크립트다. 액션스크립트도 처음에는 그냥 마우스 클릭만으로 프로그램을 만들 수 있도록 제작되었다(물론 플래시 5에서도 일반 모드를 선택하면 메뉴에서 액션을 선택하는 방식으로 프로그래밍할 수 있다). 하지만 그러한 방식의 프로그래밍에는 한계가 있다. 지금의 액션스크립트는 ECMA-262 표준을 기반으로 제대로 된 스크립트 언어의 모양새를 갖추고 있다. 같은 ECMA-262 기반 스크립트 언어인 자바스크립트를 다뤄본 개발자라면 액션스크립트도 그다지 어렵지 않게 배울 수 있다(실제로 플래시를 다루는 개발자라면 자바스크립트도 어느 정도 익숙한 경우가 많을 것이다).

물론 단순한 플래시 애니메이션은 프로그래밍을 모르더라도 만들 수 있겠지만, 중급 이상의 플래시 애니메이션을 구현하려면 액션스크립트를 약간이라도 다룰 줄 알아야 하고, 더 멋진 플래시 무비를 만들고 싶다면 액션스크립트에 대한 정확한 이해와 경험이 필요하다.

『창시자 게리 그로스먼과 함께 배우는 플래시 액션스크립트』는 액션스크립트 창시자인 매크로미디어 플래시팀 수석 연구원 게리 그로스먼이 직접 추천사를 써주었으며, 플래시팀의 철저한 기술 감수를 거쳐 세상에 나온 책이다.

이 책은 액션스크립트 초보자, 하지만 플래시는 어느 정도 알고 있는 독자들을 위한 ‘액션스크립트’ 책이다. 부디 많은 독자들이 이 책을 통해 액션스크립트의 달인이 될 수 있기를 바란다.

끝으로 이 책이 나오기까지 부족한 역자에게 도움을 아끼지 않은 한빛미디어 가족들에게 감사드린다. 그리고 항상 나를 아껴주는 부모님과 군대에서 고생하고 있는 나의 동생 재호, 언제나 나에게 큰 힘이 되어주는 혜선에게 다시 한 번 감사의 마음을 전한다.

2001년 11월
서환수

액션스크립트가 걸어온 길

게리 그로스먼

매크로미디어 플래시팀 수석 연구원

1998년 여름, 플래시 팀에 합류하기 위해 매크로미디어사에 도착했을 때, 조그맣지만 활발하게 돌아가는 그룹에서 놀라운 제품을 만들어냈다. 당시 플래시 3은 웹에서 벡터 애니메이션의 표준으로 자리잡고 있었다. 재능 있는 플래시 제작자들은 하루하루 멋진 콘텐츠를 웹에 올려 자신들의 솜씨를 뽐내고 있었다.

액션스크립트의 기원은 플래시 4 제작 계획표에서 '향상된 상호 작용'이라는 주제로 적힌 한 줄의 문장에서 찾아볼 수 있다. 플래시 3에서는 플래시의 무비 클립 및 버튼 제어 기능 따위를 모아 놓은 기본적인 수준의 동작만을 제공했다. 그래서 웬만한 프로그래밍 언어를 이용하면 간단하게 만들 수 있는 틱-택-토(tic-tac-toe) 게임을 플래시 3을 이용해 만들려면 매우 어렵고 시간도 오래 걸린다는 사실을 알았을 때 적지 않게 놀랐던 기억이 지금도 떠오른다.

그 시절은 액션스크립트가 등장하기 전이었다. 지금은 플래시 4만을 이용하여 만든 동적인 웹사이트를 발견하는 것이 그다지 놀라운 일도 아니다. 최근에는 플래시 5의 더 복잡한 액션스크립트 기능을 채용한 사이트도 등장하고 있다.

액션스크립트의 핵심 목표는 접근성에 있다. 프로그래머가 아니더라도 액션스크립트를 손쉽게 이용할 수 있도록 하는 것이 매우 중요하기 때문이다. 그래서 플래시 무비에 상호 작용을 추가하기 위해 플래시 4에서는 공백만 나타나는 스크립트 편집창 대신 시각적인 요소가 포함된 이해하기 쉬운 인터페이스를 만들었다. 플래시 4 액션스크립트는 간단하게 만들었기 때문에 배우기도 쉽고 플래시 플레이어의 용량도 줄일 수 있다.

플래시 플레이어는 인터넷 접속 속도가 느린 사용자라도 손쉽게 다운로드할 수 있도록 용량을 작게 만든 프로그램이다. 플래시팀에서는 새로운 기능을 넣기 전에

만드시 '이런 기능을 플레이어에 추가하려면 코드를 어느 정도 추가해야 할까?' 라는 문제를 생각한다. 액션스크립트를 추가할 때도 마찬가지로 이런 점을 고려했다. 플래시 플레이어에 추가되는 다른 모든 기능을 만들 때와 마찬가지로 액션스크립트를 만들 때도 최소 자본(플레이어 크기가 늘어나는 정도)으로 최대 이익(새로운 기능)을 창출하기 위한 노력을 기울였다.

사용자들이 액션스크립트를 이용하여 우리가 생각하지도 못한 일을 해낼 것이라는 예상은 이미 하고 있었지만, 실제 사용자들이 만들어내는 플래시를 지켜보는 것은 매우 즐거운 경험이었다. 플래시 4를 출시한 지 한 달만에 전자상거래, 채팅, 게시판, 아케이드 게임, 보드 게임, 심지어는 플래시 사이트를 만드는 플래시 사이트까지, 액션스크립트를 활용한 놀라운 사이트들이 속속 등장했다. 이렇게 해서 그래픽 애니메이션을 이용할 수 있고 사용자와 상호 작용도 할 수 있는 새로운 웹 콘텐츠가 생겨났다.

플래시 5를 설계할 무렵에는 무엇보다도 액션스크립트에 함수, 객체, 복잡한 흐름 제어문, 다중 자료형 따위의 자바스크립트 같은 언어에서 프로그래머들이 자주 사용하는 기능을 모두 갖춘 제대로 된 스크립트 언어를 만드는 것이 주목적이었다. 이런 기능은 이미 다른 언어에서 도입된 강력한 도구로, 수많은 프로그래머들이 이와 같은 도구를 이용하여 생산성을 높일 수 있었기 때문에 액션스크립트에도 그런 기능을 추가하고 싶었다. 완전히 새로운 언어를 만드는 대신 클라이언트측에서 작동하는 인터넷용 스크립트 언어의 표준으로 자리잡은 자바스크립트를 모델로 삼기로 했다. 더 정확하게 말하자면 액션스크립트는 ECMA스크립트 표준(ECMA-262)을 모델로 삼아 만들어졌다. 따라서 자바스크립트를 접해본 경험이 있는 사람이라면 플래시의 액션스크립트가 그다지 낯설지 않을 것이다. 또한 액션스크립트를 사용하면서 액션스크립트에 대한 지식을 바탕으로 자바스크립트까지 공부하면 두 언어로 만들어진 코드를 자유자재로 고쳐서 사용할 수 있다.

플레이어를 만들 때는 접근성과 크기의 최소화가 모두 중요하다. 자바스크립트는 정교하면서 복잡한 언어이기 때문에 우리는 플래시 4 액션스크립트처럼 사용하기 편리하면서 동시에 자바스크립트의 모든 기능을 활용할 수 있도록 하는 방법을 강구해야 했다. 이런 이유로 플래시 5 액션 패널에는 플래시 4 액션스크립트 에디터와 같은 스트림라인(streamline) 버전인 일반 모드(Normal Mode)와 고급 사용자를 위한 텍스트 에디터인 전문가 모드(Expert Mode)가 모두 포함되어 있다. 플레이어 크기를 줄이기 위해서는 ECMA스크립트 호환성을 어느 정도 포기해야 했다. 예를

들면, 액션스크립트에서는 eval()을 이용한 런타임 컴파일링 코드를 지원하지 않는다. 이 기능을 이용하려면 플레이어에 액션스크립트 컴파일러를 전부 포함시켜야 하기 때문에 그 크기가 커질 수밖에 없다. 마찬가지로 이유로 정규식(regular expression) 기능도 제외했다. 이 두 기능은 꽤 유용한 것이기 때문에 플레이어의 크기와 기능 사이에서 균형을 맞춰야 하는 플래시팀에서는 심사숙고 끝에 이런 기능을 포기할 수밖에 없었다.

접근성과 크기 외에 호환성도 고려해야 했다. 플래시 4 스크립트를 자연스럽게 플래시 5 문법으로 업그레이드할 수 있도록 플래시 5 액션스크립트를 설계했다. 또한 플래시 5에서 플래시 4의 모든 기능을 지원하여 플래시 4 무비를 만들 때도 플래시 5를 이용할 수 있도록 했다. 이 책의 ‘부록 C. 하위 호환성’과 ‘부록 D. ECMA-262와 자바스크립트’에서는 각각 하위 호환성 문제와 액션스크립트와 자바스크립트 사이의 차이점 등을 호환성을 중심으로 알아본다.

플래시팀은 개발 과정 전반에 걸쳐 놀라운 재능과 열정을 가진 플래시 사용자 커뮤니티로부터 헤아릴 수 없을 만큼 중요한 조언을 많이 얻을 수 있었다. 플래시 커뮤니티에게 얻은 조언은 플래시 제품에 포함된 기능의 전체적인 윤곽을 구축하는데 크게 도움이 되었다. 매크로미디어에서는 고객의 요구를 만족시키는 소프트웨어를 만드는 것을 목표로 하며, 이런 목표를 달성하는 과정에서 고객들의 작업 방법, 고객들의 의견에서 많은 도움을 얻었다.

마지막으로 강조하고 싶은 점은 플래시는 여전히 진행중인 프로젝트이며 사용자들의 요구를 만족시키기 위해 꾸준히 노력할 것이라는 점이다. 플래시 개발자가 정보 시대의 예술가라면 플래시팀은 그들을 위해 최고의 붓과 조각칼을 만드는 사람들이다. 액션스크립트의 발전 과정에서 중요한 역할을 할 이 책은 액션스크립트만을 다룬 포괄적인 튜토리얼이자 레퍼런스이다. 이제는 액션스크립트도 꽤나 복잡해져서 이처럼 모든 내용을 담고 있는 좋은 책이 한 권쯤 필요할 때도 되었다고 생각한다.

이 책을 읽는 플래시 디자이너들이 플래시 5 액션스크립트를 즐거운 마음으로 배울 수 있기를 바란다. 우리 플래시팀에서는 수많은 디자이너들이 만들어낸 우수한 작품들을 기대할 것이다.

차례

역자 서문	5
추천사	7
차례	10
서문	17

1부. 액션스크립트 기초

1장. 프로그래밍을 모르는 독자를 위한 기본 소개	29
몇 가지 기본 문장	32
액션스크립트 심화 개념	43
퀴즈 프로그램 만들기	52
2장. 변수	71
변수 만들기(선언)	72
변수 값 대입	75
변수 값 변경 및 검색	77
값 유형	78
변수 영역	81
몇 가지 응용 예제	94

3장. 데이터와 데이터형	97
데이터 대 정보	97
데이터형에서 의미 찾기	98
데이터 생성 및 범주 나누기	99
데이터형 변환	102
원시 데이터와 복합 데이터	113
4장. 원시 데이터형	115
숫자형	115
정수와 부동소수점수	116
숫자 리터럴	116
숫자 계산	122
문자열	123
문자열 조작	128
부울형	153
Undefined	155
Null	156
5장. 연산자	159
연산자의 일반적인 특징	159
대입 연산자	164
수학 연산자	166
동치 및 부등 연산자	171
비교 연산자	177
문자열 연산자	181
논리 연산자	182
그룹 연산자	189
선험 연산자	190
void 연산자	191
기타 연산자	191

6장. 선언문	197
선언문의 유형	198
선언문의 문법	199
액션스크립트 선언문	201
선언문과 액션	210
7장. 조건문	213
if 선언문	214
else 선언문	216
else if 선언문	218
switch 선언문 흉내내기	220
간결한 조건문	222
8장. 순환문	225
while 루프	225
순환문 관련 용어	230
do-while 루프	231
for 루프	233
for-in 루프	234
루프 중단	236
타임라인 루프와 클립 이벤트 루프	240
9장. 함수	251
함수 생성	252
함수 실행	253
함수에 정보 전달하기	254
함수 종료 및 함수 값 리턴	258
함수 리터럴	262
함수 사용 범위와 유효 기간	262
함수 영역	264
함수 매개변수 다시 보기	269

재귀 함수	274
내부 함수	276
객체로서의 함수	278
코드 집중화	280
객관식 퀴즈 다시 보기	281
 10장. 이벤트 및 이벤트 핸들러	 287
동기 코드 실행	287
이벤트 기반의 비동기 코드 실행	288
이벤트 유형	289
이벤트 핸들러	290
이벤트 핸들러 문법	291
이벤트 핸들러 만들기	291
이벤트 핸들러 영역	295
버튼 이벤트	301
무비 클립 이벤트 개요	306
무비 재생 무비 클립 이벤트	307
사용자 입력 무비 클립 이벤트	315
실행 순서	323
클립 이벤트 핸들러 복사	325
updateAfterEvent를 이용한 화면 갱신	326
코드 재사용	327
동적 무비 클립 이벤트 핸들러	328
이벤트 핸들러 응용	329
 11장. 배열	 333
배열이란 무엇인가?	334
배열의 구조	335
배열 생성	336
배열 원소 참조	339
배열 크기 결정	342
이름이 있는 배열 원소	344

배열에 원소 추가하기	346
배열에서 원소 제거하기	352
범용 배열 조작 도구	357
다차원 배열	363
객관식 퀴즈, 버전 3	365
 12장. 객체와 클래스	 367
객체 구조	370
객체 인스턴스 만들기	371
객체 속성	372
메소드	374
클래스와 객체지향 프로그래밍	379
액션스크립트 내장 클래스와 객체	399
 13장. 무비 클립	 403
무비 클립의 객체성	404
무비 클립 유형	406
무비 클립 생성	409
무비와 인스턴스 스택 순서	418
인스턴스 및 메인 무비 참조	425
메인 무비 및 클립 인스턴스 제거	442
내장 무비 클립 속성	445
무비 클립 메소드	446
무비 클립 응용 예제	452
마지막 퀴즈	527
 14장. 렉시컬 구조	 463
공백	463
선언문 종료자(세미콜론)	465
주석	467
예약어	469

인식자	471
대문자와 소문자	471
15장. 고급 주제	475
데이터 복사, 비교 및 전달	475
비트 단위 프로그래밍	479
고급 함수 영역 문제	494
movieclip 데이터베이스	497

2부. 액션스크립트 응용

16장. 액션스크립트 저작 환경	501
액션 패널	501
프레임에 스크립트 추가하기	505
버튼에 스크립트 추가하기	506
무비 클립에 스크립트 추가하기	508
코드는 어디에?	508
생산성	509
액션스크립트 코드 외부화	511
컴포넌트를 스마트 클립으로 만드는 법	514
17장. 플래시 폼	523
플래시 폼 데이터 주기	523
플래시 입력 폼 만들기	527
18장. 온스크린 텍스트 필드	537
동적 텍스트 필드	537
사용자 입력 텍스트 필드	539
텍스트 필드 옵션	541
텍스트 필드 속성	546

HTML 지원	550
텍스트 필드 선택 영역	561
비어있는 텍스트 필드와 for-in 선언문	561
19장. 디버깅	563
디버깅 도구	564
디버깅 방법론	571

3부. 레퍼런스

액션스크립트 레퍼런스	583
-------------------	-----

4부. 부록

부록 A. 참고자료	927
부록 B. Latin 1 문자 범주 및 키코드	933
부록 C. 하위 호환성	941
부록 D. ECMA-262 및 자바스크립트와의 차이점	947
찾아보기	951

