

공개 SW 프로젝트 챌린지 2007  
스쿼드 이토이 기반의  
교육용 프로그래밍 환경 개선

2007年 11月

InC Lab.

김승범, 박동희, 장운재

# 차례

제 1 장	서론	1
1	팀 소개 . . . . .	1
2	개발 배경 . . . . .	2
2.1	아이들을 위한 컴퓨터? . . . . .	2
2.2	교육용 프로그래밍 언어 . . . . .	3
2.3	스쿼 이토이 . . . . .	4
2.4	스쿼 이토이의 유용성 분석 . . . . .	6
3	개발 목적 . . . . .	7
4	개발 환경 . . . . .	7
제 2 장	개선 작업	8
1	Knob Morph . . . . .	9
1.1	Problem . . . . .	9
1.2	Solution . . . . .	9
1.3	Design and Implementation . . . . .	10
1.4	Examples . . . . .	10
2	LED Meter Morph . . . . .	11
2.1	Problem . . . . .	11
2.2	Solution . . . . .	11
2.3	Design and Implementation . . . . .	11
2.4	Examples . . . . .	12

3	Object Market . . . . .	13
3.1	Problem . . . . .	13
3.2	Solution . . . . .	13
3.3	Design and Implementation . . . . .	14
3.4	Examples . . . . .	14
4	SkSheet e-Toy . . . . .	15
4.1	Problem . . . . .	15
4.2	Solution . . . . .	15
4.3	Design and Implementation . . . . .	16
4.4	Examples . . . . .	17
5	객체와 뷰어 관계 개선 . . . . .	19
5.1	Problem . . . . .	19
5.2	Solution . . . . .	19
5.3	Design and Implementation . . . . .	19
5.4	Examples . . . . .	20
6	Category menu 개선 . . . . .	21
6.1	Problem . . . . .	21
6.2	Solution . . . . .	21
6.3	Design and Implementation . . . . .	21
6.4	Examples . . . . .	22
	<b>참고 문헌</b>	<b>23</b>

# 제 1 장

## 서론

### 1 팀 소개

InC Lab. 팀은 고려대학교 컴퓨터교육학과의 대학원생과 학부생으로 구성된 팀으로, 개발자 커뮤니티인 한국 스킵 사용자 모임 (<http://squeak.kr>) 과 스킵을 활용한 교육용 커뮤니티인 알고스킵 (<http://algo.squeak.kr>) 을 운영하고 있다. 스킵 이토이를 이용한 프로그래밍 교육을 개발하고 공교육 및 대안 교육 현장에서 적용하고 실험하고 있다.

**김 승 범** - 한국 스킵 사용자 모임 (<http://squeak.kr>) 의 운영자로 InC Lab. 팀의 리더이다. 대안언어축제 (<http://altlang.org>) 등의 각종 오픈소스 커뮤니티를 통해 스킵 스몰토크 (squeak smalltak) 를 한국에 알려왔고, 스킵 이토이 (squeak eToys) 등의 교육용 프로그래밍 환경을 통한 일반 사용자 프로그래밍 (end-user programming) 을 연구하고 현장에 적용하는 노력을 하고 있다.

**박 동 희** - 스쿼커(squeaker) 이다! 다양한 미디어를 다루는것을 즐기고, 피지컬(physical)컴퓨팅에 관심이 많다. 미디어 아트 축제인 p.art.y 2007에서 관객들의 몸으로 연주하는 공연인 CC Real Mixer에 참가 하였다.

**장 윤 재** - 고려대학교 컴퓨터교육과 학부생으로 동대학교 김승범군을 만나면서 스킵과 스쿼커라는 길에 입문하였다. 현재 연세대학교 청년문화원에서

진행하는 온라인 학습생태계 프로젝트 (<http://gazi.tistory.com/>) 에서 대안학교 학생들에게 스킵 이토이를 가르치는 스킵 강사로 활동하고 있으며, 정보교과 중에서 교육용 프로그래밍 언어에 관심이 많다. 스킵 이토이를 통해 학생들이 즐겁게 프로그래밍 할 수 있는 방법에 대해 고민하고 있다.

## 2 개발 배경

### 2.1 아이들을 위한 컴퓨터?

아이들은 컴퓨터 앞에서 무엇을 할까? 그리고 컴퓨터를 통해 무엇을 배울 수 있을까? 초고속 인터넷 인프라를 세계 최고 수준으로 갖추고 있고, 국가 차원에서 IT 강국을 외치는 대한민국에서 어린 자녀를 둔 부모들은 컴퓨터 앞에 하루 종일 앉아 있는 그들의 아이들을 걱정한다. 변화하는 환경에 적응하고 국가의 경쟁력을 유지하기 위하여 무엇보다 IT 능력을 갖춘 고급 인재의 양성이 절실히 요구된다고 말하면서, 동시에 컴퓨터는 '필요악'이라고 말한다. 게임 소프트웨어 산업의 세계적 발전에 대해서는 자랑스러워하면서 아이들이 게임을 하는 모습에는 탐탁치가 않다. 우리 아이들이 컴퓨터 산업의 무분별한 소비자로 전락하였기 때문이다.

단순 소비자인 아이들은 컴퓨터 앞에서 정신과 에너지를 무의미하게 소모하기만 한다. 그들에게 컴퓨터는 단지 쾌락의 도구가 되었다. 그들에게 컴퓨터라는 도구를 던져주고는 정작 어떻게 사용하는지에 대해서는 제대로 가르쳐주지 못했기 때문이다. 대부분의 아이들은 컴퓨터를 통해서 자신의 창의적인 아이디어를 실험하고 스스로 깨우치는 건강한 생산자적인 경험을 하지 못하고 있다. 컴퓨터에 대한 왜곡된 인식과 잘못된 정보교육을 바로 잡기 위한 노력이 절실히 요구된다.

세계 각국에서는 이러한 정보교육의 필요성을 인식하고 많은 연구를 진행 중에 있다. 컴퓨터과학교육 표준에 관한 연구를 다루는 ACM협회의 보고서에 의하면, “컴퓨터과학이란 컴퓨터와 알고리즘적 처리과정에 대한 연구로써 21 세기를 살아가는 시민의 필수 교양으로 컴퓨터과학을 학생들과 일반인에게 교

육시켜야 한다.” 고 밝히고 있다. 그리고 알고리즘적 사고를 위해서 초 중등학교 교육과정에서도 프로그래밍 교육의 필요성을 주장하고 있다.

그러나 기존의 프로그래밍 언어와 교육방법들은 학생들의 많은 인지적 부담을 요구하고 있다. 프로그래밍 언어의 기본 문법과 구조를 이해하고 사용하는 방법을 익히는 과정에서 많은 시간과 노력이 요구되어 학생과 교사 모두 프로그래밍을 어렵게 여기고 있다. 결국 프로그래밍이라는 행위를 통해 배울 수 있는 가치에 접근하기도 전에 프로그래밍 언어라는 도구에 지쳐버리고, 학생들에게 컴퓨터라는 부담스러운 과목을 새로 안겨준 꼴이 되었다.

아이들에게는 자신의 꿈과 아이디어가 눈앞에 실현이 되는 경험이 필요하다. 선생님과 교과서에 의해 주입이 되는 공부가 전부가 아니라 스스로 원리를 깨우치고 새로운 개념을 창조하는 즐거운 공부가 있다는 것을 알게 해야 한다. 내가 배운 지식과 경험들이 세상 속에서 어떻게 적용되고 유용하게 사용될 수 있는지를 이해해야 한다.

## 2.2 교육용 프로그래밍 언어

1960년대 부터 수많은 프로그래밍 언어와 환경들이 여러가지 목적에 따라 만들어졌다. 기계에 가까운 언어에서 인간에 가까운 언어, 여러 목적으로 쓰일 수 있는 범용 언어에서 특정 대상과 목적을 위한 언어까지 다양한 프로그래밍 언어가 존재한다. 그 중에 교육적인 목적으로 여러 연령층의 엔드유저를 위해 설계된 언어가 바로 교육용 프로그래밍 언어 (Educational Programming Language, 이하 EPL) 이다.

프로그래밍 언어에는 여러가지 패러다임이 있다. 절차형 언어, 객체지향 언어, 이벤트 기반 언어, 함수형 언어, 실연에 의한 프로그래밍 (programming by demonstration), 그래픽 기반 규칙 다시쓰기 (graphcial rewrite rule), 자율 에이전트 (autonomous agents), 데이터 흐름 기반 (data flow), 규칙 기반 프로그래밍 (rule-based programming), 논리 프로그래밍 (logic programming), 병렬 프로그래밍 (parallel programming) 등 이 있으며 EPL에서도 이런 다양한 패러다임이 사용되고 있다[8].

전문 개발자에 비해 프로그래밍에 익숙하지 않은 사용자는 프로그래밍 언어의 패러다임에 따라 결과에 영향을 받는 정도가 크다 [10]. 이는 EPL을 통한 프로그래밍 학습 효과가 패러다임에 의해 영향을 받을 수 있다는 것이다. 그러나 어떤 패러다임이 EPL로써 가장 좋다고 말 할 수 없으며, 오히려 상업 프로그래밍 언어에 비해 다양한 패러다임들이 시도되고 있고 학습 대상자의 연령층이나 목표로하는 학습 내용에 따라 적합한 패러다임을 선택하고 있다.[5].

## 2.3 스쿼크 이토이

스쿼크 이토이 (Squeak eToys) 는 엔드유저를 위한, 그 중에서도 아이들을 위해 설계된 비주얼 기반의 타일 스크립트 프로그래밍 환경이다.

스쿼크 이토이는 미국, 일본, 유럽, 남미 등지에서 교육용 매체로 성공적으로 적용되고 있고 [11], 최근에는 MIT의 니그로폰테 교수가 이끌고 있는 OLPC (One Laptop Per Child) [7] 프로젝트에 교육용 프로그래밍 환경으로 스쿼크 이토이가 탑재되었다.

한국에서는 개발자 커뮤니티인 스쿼크 사용자 모임이 필자에 의해 운영되고 있고 [6], 스쿼크 이토이의 교육적 활용에 대한 연구는 필자를 포함한 고려대학교 컴퓨터교육학과 InC 연구실에서 교육방법과 교육용 프로그래밍 환경에 대한 연구가 활발히 진행되고 있다[1, 3].

### 2.3.1 스쿼크 이토이의 철학

스쿼크 이토이를 통해 아이들은 주변에서 관찰되는 현상들을 컴퓨터에 모델링하고 시뮬레이션하면서 현상의 원리들을 ‘실연에 의한 증명’ (Proof by Demonstration) 을 통해서 스스로 깨우치게 된다[2, 4]. 로고 (LOGO) 를 개발한 세이무어 페퍼트 (Saymour Papert) 는 이를 그의 저서 마인드스톰 (Mindstorm) 에서 ‘파워풀 아이디어’ (Powerful Idea) 라고 표현하였다[9].

스쿼크 이토이는 세이무어 페이퍼트, 피아제, 몬테소리, 듀이, 비고츠키, 제롬 브루너의 교육적 철학을 기반으로 만들어진 컴퓨팅 환경으로 LOGO, Smalltalk, Hypercard, Starlogo 등의 영향을 받았다. 아이들은 아이디어에 대해 동작으로 해보고, 비주얼 혹은 소리, 상징적인 표현 등을 통해 직접 구성해 나가면서 그

개념을 가장 잘 배워나갈 수 있다는 이론을 기초로 하고 있다.

### 2.3.2 스킵 이토이의 특징

다양한 교육용 프로그래밍 언어가 존재하지만 스킵 이토이를 선택한 이유로는 다음과 같은 특징을 갖고 있기 때문이다.

스킵 이토이의 특징은 다음과 같다.

**비주얼하고 다이나믹한 그래픽 객체와 멀티미디어 사용** 스킵 이토이에서 사용하는 대상은 대부분 그래픽 객체로 이뤄져있다. 사용자가 그린 그림은 바로 조작할 수 있는 객체가 되고, 비주얼한 프로그래밍 방식을 채택하고 있다. 그리고 사진, 그림, 소리 등의 다양한 멀티미디어 자료를 쉽게 사용할 수가 있는 특징을 갖고 있다.

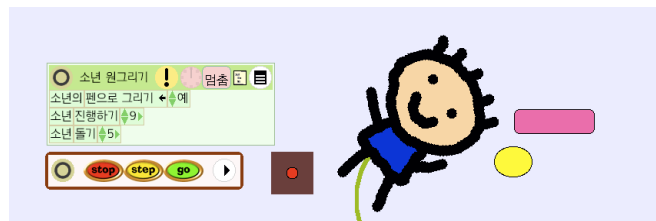
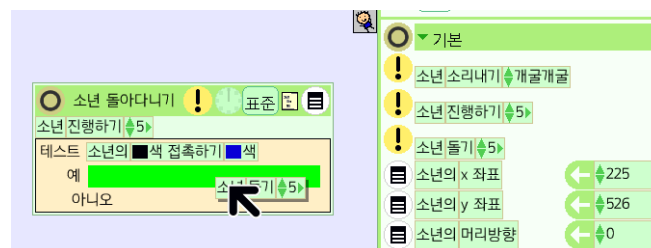


그림 1.1: 그래픽 객체로 이뤄진 스킵 이토이

타일 스크립팅 (tile-scripting) 을 통한 프로그래밍 스킵 이토이는 텍스트 스크립팅과 비주얼 스크립팅의 중간 방식인 타일 스크립팅으로 프로그래밍을 한다. 이를 통해 텍스트 스크립팅이 가진 단점인 ‘오타로 인한 문법 오류’를 최소화하고, ‘상자와 선 (box-and-line)’ 방식을 사용하는 비주얼 프로그래밍 보다 순차적인 코드의 구성이 더 용이한 장점을 갖고 있다.





## 그림 1.2: 스킵 이토이의 타일 스크립팅

**다양하고 유용한 객체와 도구** 스킵 이토이 내부에는 조이스틱 컨트롤러나 다양한 객체처럼 바로 사용이 가능한 다양한 객체와 도구들이 마련되어 있다. 이들은 각각의 기능은 다르지만 조작하는 방식에 있어서는 타일 스크립팅이라는 일관성을 가지고 있어서 사용하기가 쉽다.

**확장성 높은 오픈소스 환경** 스킵 이토이는 스킵 스몰토크라는 확장성이 매우 높은 오픈소스 컴퓨팅 환경 위에서 동작한다. 누구나 스킵의 소스코드를 수정하고 이토이를 개선할 수 있다.

**폭 넓은 커뮤니티** 스킵 이토이는 미국, 일본, 유럽, 남미 등지에서 교육용 매체로 사용되고 있다. 관련 정보를 교류하는 메일링 리스트가 활발히 운영되고 있고, 매년 교육자와 개발자들이 모이는 컨퍼런스도 열리고 있다. OLPC 노트북에 기본 교육용 프로그래밍 환경으로 탑재되면서 앞으로 더 많은 사용자와 교육자 커뮤니티가 생겨날 것이다.

## 2.4 스킵 이토이의 유용성 분석

스킵 이토이는 본 연구진들을 통해 초등학생부터 대학교 학부생 그리고 예비 교사들에게 프로그래밍 교육 도구로 적용되었다. 그 과정을 통해 여러가지 문제점들을 관찰하였고, 유용성 분석으로 접근할 필요성이 생겼다. 교육용 프로그래밍 환경을 고려한 유용성 평가 프레임워크인 EPE-UF (유용성 평가를 위한 heuristic evaluation 의 확장 프레임워크로 김승범이 개발) 를 기반으로 스킵 이토이를 분석하고 다음과 같은 영역에서 개선이 필요하다.

- 카테고리 메뉴 관련
- 객체와 스크립트 제어 관련
- 타일 스크립트 관련
- 객체의 확장 기능 추가 관련
- 공유/협업 관련
- 용어와 기호의 의미 관련

- 스킵 스몰토크 시스템 관련
- 각종 메뉴의 제어 관련
- 그림판 제어 관련

EPE-UF 유용성 평가에 대한 프레임워크 개발 내용과 연구 방법은 본 보고서의 범위를 벗어나므로 생략하겠다.

### 3 개발 목적

스킵 이토이는 그 자체로 훌륭한 교육용 프로그래밍 환경이다. 그러나 여러 차례의 현장 적용 경험을 통해 많은 문제점들이 지적되어왔다. 제기된 여러 문제점들을 유용성 분석을 통해 체계적으로 분석하고, 개선점들을 도출하였다. 본 개발을 통해 문제점들의 개선을 시도하였다.

### 4 개발 환경

개발 환경은 다음과 같다.

**Squeak Smalltalk (OLPC 2.0)** - 스몰토크는 프로그래밍 언어이자 컴퓨팅 환경이다. 일반적으로 널리 사용되는 프로그래밍 언어가 텍스트 기반의 소스코드를 컴파일해서 특정 운영체제나 VM에서 실행되는 바이너리를 생성한다면, 스몰토크는 그 자체가 동적인 컴퓨팅 환경으로 실시간으로 실행되고 있는 구조를 변경하고 적용할 수 있다. 스킵 스몰토크 내부에 개발도구와 디버거, 실행환경이 모두 포함되어 있다. 이번 개발에 사용된 스킵은 OLPC 2.0버전으로 OLPC에 탑재될 스킵 이토이를 위해 개선된 버전이다.

## 제 2 장

# 개선 작업

스쿼 이토이의 교육용 프로그래밍 환경 개선은 다음과 같은 형식으로 정리한다.

**Name** 개선에 대한 제목

**Problem** 스쿼 이토이에서 발생하는 문제, 필요한 개선을 정리하고 문제를 이해하는데 도움이 되는 상황을 적는다.

**Solution** 문제를 개선할때 사용한 모델이나, 아이디어 내기

**Design and Implementation** 해결하는 과정을 리뷰하고, 해결되어 나온 결과물의 설계와 스쿼상의 구현을 정리한다.

**Examples** 개선된 결과물을 예로 보인다.

# 1 Knob Morph

Usability Issue: 객체의 확장 기능 추가 관련

## 1.1 Problem

스킵 이토이에는 대표적인 입력 도구로 조이스틱과 슬라이더가 있다. 조이스틱은 객체를 동작시키는 용도로 사용이 되고, 슬라이더는 값을 변경하는 용도로 사용된다. 조이스틱과 슬라이더는 일상 생활에서 사용되는 입력 도구의 모델을 따르고 있어서 직관적인 스킵 이토이 프로그래밍에 도움이 된다. 그러나 일상 생활에서는 우리에게 익숙한 그 외의 입력도구들이 있다. 스킵 이토이에 더 다양한 입력 도구를 제공해주면서 프로그래밍 방식에는 일관성을 가지도록 제공할 줄 필요가 있다.



그림 2.1: 일상 생활의 노브 컨트롤러

## 1.2 Solution

대표적으로 많이 사용되는 입력 도구로 노브 (knob) 를 들 수 있다. 노브를 메타포로 스킵 이토이 입력 도구를 추가하였다. 노브는 볼륨을 조절하거나 객체의 속성값을 변경하는 목적으로 사용한다.



그림 2.2: 노브 입력도구

### 1.3 Design and Implementation

노브 형식의 모프를 사각형, 원형 모프를 이용하여 구성하였다. 최소값, 최대값, 현재 값에 대한 속성을 이토이 인터페이스로 확장하여 다른 이토이 프로그래밍의 입력 도구로 사용할 수 있도록 만들었다.

- Class name : KnobMorph

### 1.4 Examples

LED 모프의 예제에서 입력도구로 사용된 예제를 확인할 수 있다.

## 2 LED Meter Morph

Usability Issue: 객체의 확장 기능 추가 관련

### 2.1 Problem

스쿼크 이토이에는 객체의 상태를 보여주는 출력도구가 부족하다. 특히 교육용 프로그래밍 환경에서는 객체의 상태를 표시하는 도구는 현상을 시뮬레이션하고 결과를 관찰하는데 중요하다.

일상 생활에서 익숙한 출력 도구 중에는 LED가 있다. 전광판이나, 전자 회로에 들어가는 빛을 발하는 소자이다. 전자회로에서 상태를 알려주는데 LED를 사용한다. LED의 on/off, 깜박이는 정도 차이로 회로의 상태를 보여준다.

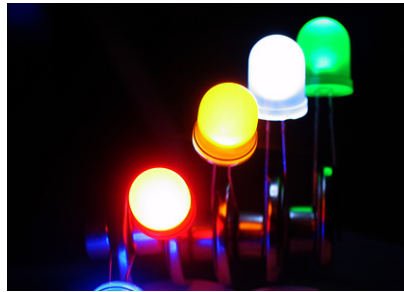


그림 2.3: LED

### 2.2 Solution

전자회로의 LED를 이토이의 도구로 만들었다. LED의 동작을 재현하는 Morph를 만든 후, 이토이에서 사용할 수 있도록 LED Morph에 이토이 프로토콜을 추가하였다.

### 2.3 Design and Implementation

스쿼크의 Morph에서 시간당 움직임(frames/time)을 표현하기 위해서 step이라는 메시지를 사용한다. 각 스텝은 Morph의 동작 또는 모양을 나타내고 각 스텝의 시간은 stepTime 메시지를 통해서 조정 할 수 있다.

LedMeter Morph는 이토이 타일을 통해 stepTime을 입력받고, step을 통해 LED동작 (깜박임, 켜기, 끄기) 을 조정한다.

- Class name : LedMeterMorph

## 2.4 Examples



그림 2.4: LED 모프 사용 예제

Knob Morph를 입력 도구로, LedMeter Morph를 출력 도구로 사용하는 예 이다.

### 3 Object Market

Usability Issue: 공유/협업 관련

#### 3.1 Problem

스킵 이토이를 사용하는 프로그래밍 교육 수업에서 학생들 간의 협업을 위한 도구가 부족했다. 학생들의 프로젝트를 공유하는 방법은 있었으나 객체를 쉽게 공유 하는 방법이 없었다.

온라인 생태학습 프로젝트로 진행하는 대안학교 프로그래밍 교육수업 중 에 학생들 간의 협업을 돕는 환경으로 '시장에서 물건을 교환' 하는 것 처럼 스킵 환경 안에서 객체를 교환 할 수 마켓이 필요했다.



그림 2.5: 시장

#### 3.2 Solution

객체를 교환할 수 있는 장소를 스킵 안에 만들었다. 오브젝트 마켓은 야마미야 (Takashi Yamamiya) 의 Super Parts Bin 을 사용하였다. Super Parts Bin을 사용하면 네트워크에서 다른 사람이 만든 객체를 쉽게 가져올 수 있다. 이제 오브젝트 마켓을 사용하면 친구와 협업 프로젝트를 더 쉽게 할 수 있게되었다.





그림 2.6: 오브젝트 마켓

### 3.3 Design and Implementation

오브젝트 마켓을 기존의 도구 플랩과 동일한 개념으로 사용할 수 있도록 화면의 하단에 배치시켰다. 오브젝트 마켓의 저장소 역할로 외부 서버에 WebDAV를 세팅하고, 스크립트를 실행하면 오브젝트 마켓 용 플랩을 자동으로 실행하도록 SuperPartsBin을 확장하였다.

- Class name : LedMeterMorph

### 3.4 Examples

제출된 스크립트 이토이를 실행하면 하단에서 오브젝트 마켓 플랩을 확인할 수 있다.

## 4 SkSheet e-Toy

Usability Issue: 객체의 확장 기능 추가 관련

### 4.1 Problem

스킵 이토이에서는 숫자와 같은 정밀한 데이터를 저장하고 다룰 수 있는 도구가 부족하다. 시뮬레이션 작성 도중이나 결과에서 나오는 유의미한 수치 정보를 활용할 수 없고, 그 정보들을 시뮬레이션의 입력값으로 사용하기도 어렵다. 또한 스킵 이토이에서 시간의 흐름에 따라 발생하는 값들, 특히 수치 데이터를 모두 순서대로 저장해서 보기가 어렵다. 학습자가 원하는 시뮬레이션을 제작하기 위해서 그 객체가 움직이는 형태를 통해서만 스크립트의 피드백을 받을 수 있다. 이는 스킵 이토이를 사용해서 조금 복잡한 시뮬레이션을 제작하는데 불편함을 초래한다.

### 4.2 Solution

스킵 이토이에 스프레드시트 스타일의 인터페이스를 사용한다. 스프레드시트는 데이터를 쉽게 다룰 수 있는 가장 성공적이고 효과적인 엔드-유저 프로그램으로 이미 검증되어 많은 분야에서 사용하고 있다. 학습자는 스프레드시트를 활용한 프로그래밍 과정에서 메모리 할당, 변수이름 생성, 데이터타입 선언 과정없이 바로 데이터를 사용할 수 있다. 또한 '=' 기호를 통해 변수들간의 관계를 수식 기호로 나타낼 수 있어 데이터를 쉽게 참조할 수 있다.

이 개선작업에서는 야마미야 (Takashi Yamamiya) 가 만든 Skeleton Spreadsheet (SkSheet)을 활용해서 이토이 인터페이스로 기능을 확장시킨다. SkSheet에 커서를 움직일 수 있는 타일을 추가시켜 객체의 움직임에 따라 변하는 값을 커서가 움직이는 순서대로 저장할 수 있는 스크립트를 작성할 수 있도록 한다. 또한 커서가 움직이면서 그 해당 값을 읽어 객체의 특정 속성값에 부여함으로써 새로운 입력 인터페이스로 활용할 수 있다.

스킵 이토이에서 사용할 수 있는 SkSheet에 이토이 명령어 타일을 접목시킴으로써 비주얼 프로그래밍 환경에서 학습자가 쉽게 데이터를 다룰 수 있는

인터페이스로 활용할 수 있을 것이다.

### 4.3 Design and Implementation

스프레드시트는 행과 열로 이루어져 있다. 행과 열이 만나는 셀을 하나의 데이터 저장 공간으로 사용하며 행 또는 열의 값을 이용해서 셀을 이동할 수 있다. 또한 셀 안에는 수치 데이터를 저장하거나 다른 셀의 참조값을 사용한 수식값을 넣을 수 있다.

스프레드시트를 구성하는 요소인 행과 열을 사용하여 스크 이토이에서 명령어 타일로 제어할 수 있도록 한다.

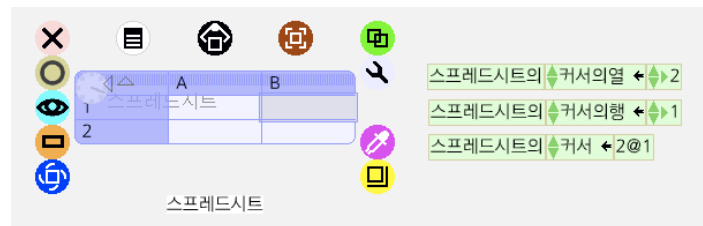


그림 2.7: 현재 선택된 셀의 행, 열, 커서 타일

행과 열로 선택된 셀의 수치 데이터나 수식을 직접 수정 또는 사용할 수 있도록 값을 읽고 저장할 수 있는 타일을 제공한다.



그림 2.8: 현재 선택된 셀의 값 또는 식 타일

객체가 움직임에 따라 발생하는 수치 데이터를 순서대로 저장할 수 있도록 하기 위해서는 다음과 같은 방식으로 스크립트를 작성할 수 있다.

1. 셀이 일정한 간격으로 움직일 수 있는 스크립트 작성

## 2. 일정한 간격으로 셀안에 객체의 속성값을 저장하는 스크립트 작성

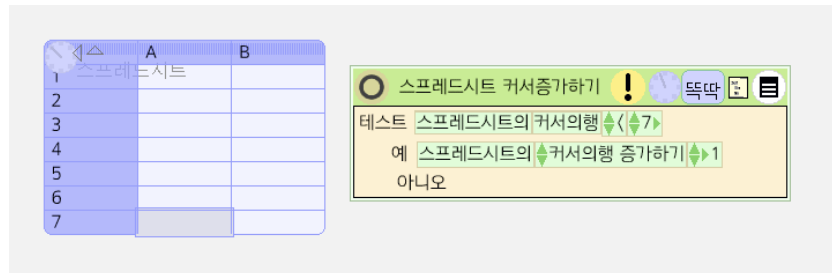


그림 2.9: 커서 증가하는 스크립트

SkSheet에 추가한 이토이 타일은 다음과 같다.

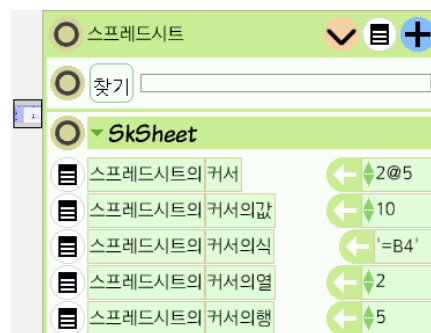


그림 2.10: SKSheet의 이토이 타일

실제 구현은 SkSheetMorph에 스프레드시트 관련 코드를 추가하고 Morphe-Scripting의 Player에 이토이 타일과 연결짓는 코드를 추가하였다.

- Class name : SkSheetMorph, Player

## 4.4 Examples

**중력 가속도 실험하기** 등속도운동과 가속도운동을 비교할 수 있도록 시트에 숫자로 저장하고 일정시간동안 차이를 계산한다. 객체가 떨어지는 동안 좌표값을 스프레드시트의 커서위치에 저장하며 일정한 간격으로 커서가 움직이도록 한다. 또한 셀간의 차이를 계산해서 다른 곳에 표현함으로써

객체가 떨어지는 동안의 차이를 볼 수 있도록 한다.

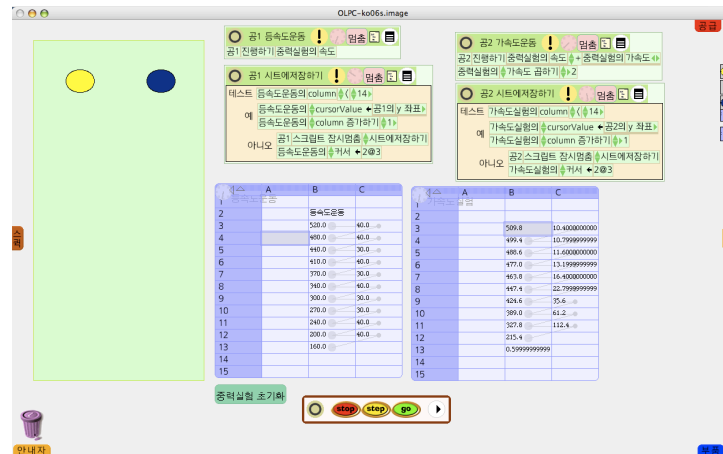


그림 2.11: sksheet를 이용한 중력 가속도 실험

로봇맵 만들기 시트에 숫자를 넣어 움직이되 특정한 숫자에 따라 로봇의 행동이 바뀌도록 만든다. 예를 들어 커서가 움직이다가 100을 만나면 객체는 소리를 내고, 200을 만나면 객체가 오른쪽을 90회전하도록 해서 로봇 객체를 움직일 수 있는 맵을 작성한다.

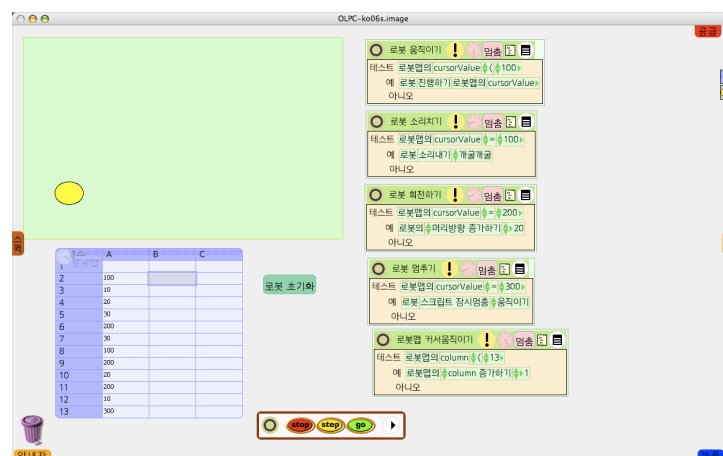


그림 2.12: sksheet를 이용한 로봇맵 만들기

## 5 객체와 뷰어 관계 개선

### 객체와 스크립트 제어 관련

#### 5.1 Problem

스틱 이토이를 수업에서 사용하다보면 비주얼 프로그래밍 환경이라는 특성때문에 시각화된 정보에 관련한 문제가 많이 발생한다. 대표적인 문제 중 하나로 유사한 객체를 많이 생성했을 때, 해당 객체의 뷰어 플랩을 잘못 선택해서 여는 경우가 발생한다. 다른 객체의 명령어 타일을 사용하게 되면서 프로그램에 오류를 발생시킨다.

#### 5.2 Solution

뷰어 플랩에는 이미 객체의 모양을 나타내는 축소된 그림이 있다. 다양한 모양의 객체를 사용할 때에는 그림 플랩으로 충분하지만 만약 유사한 객체를 여러개 만들게 되면 플랩의 그림만으로는 원하는 객체의 뷰어를 여는데 힘들어진다. 물론 객체의 할로 메뉴를 열어 바로 뷰어를 열 수 있지만, 대부분의 사용자들은 눈에 보이는 뷰어 플랩을 통해 바로 접근하게 된다. 그래서 뷰어 플랩에 마우스 커서가 올라가면 해당하는 객체가 “저예요” 라고 알릴 수 있도록 개선하였다.

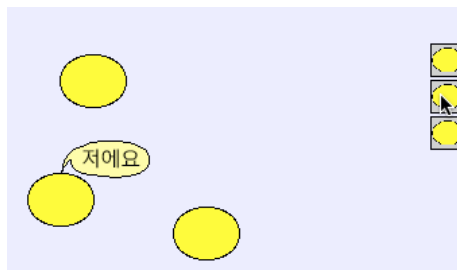


그림 2.13: 뷰어와 관련된 객체 보여주기

#### 5.3 Design and Implementation

ViewerFlapTab에서 마우스 진입 이벤트를 받을 수 있도록 추가하고, 마우스 진입 시 해당하는 객체가 풍선 도움말 기능을 이용해서 자신을 알릴 수 있도록

Player를 확장하였다.

- Class name : ViewerFlapTab, Player

## 5.4 Examples

제출된 스크 이토이를 실행하면 확인할 수 있다.

## 6 Category menu 개선

## 카테고리 메뉴 관련

## 6.1 Problem

객체는 명령어와 속성을 담은 뷰어를 열 수 있고, 뷰어에는 명령어와 속성에 종류에 따라 카테고리로 묶여있다. 각 카테고리 별로 여러가지 명령어들이 포함되다보니 때로는 어떤 명령어가 어느 카테고리에 포함되어 있는지 찾기 힘들어진다. 스킵 이토이에 익숙한 사용자도 원하는 명령어를 찾기 위해 여러 카테고리를 반복해서 들어가게 되는 문제가 자주 발생한다.

## 6.2 Solution

뷰어의 다른 카테고리를 선택하는 메뉴에서 해당 카테고리에 어떤 명령어가 존재하는지 미리 보여주는 기능으로 문제를 해결해 줄 수 있다.

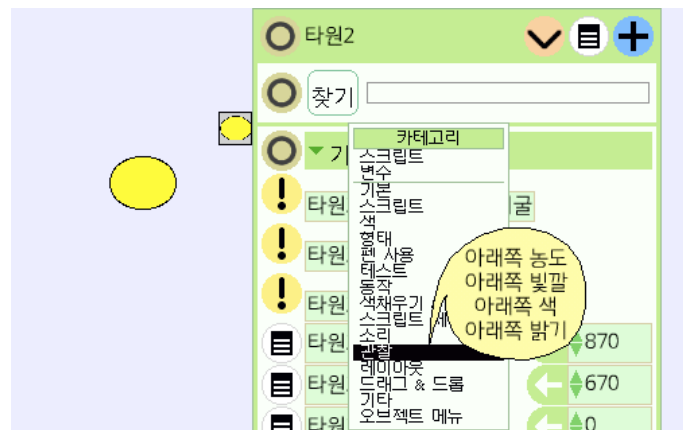


그림 2.14: 카테고리 메뉴 선택 개선

### 6.3 Design and Implementation

스퀘어 이토이에서 카테고리 메뉴를 보여주는 부분은 여러 클래스가 관련이 되어 있다. 해당되는 클래스를 바로 수정하면 앞으로 호환성에 문제를 발생시킬 수 있어, 기존의 클래스를 상속받아 확장하였다. CategoryViewer에서 새롭게



확장된 `EtoyViewerCustomMenu`를 생성하고, UI를 구성하는 `MenuMorph`와 `MenuItemMorph`를 `EtoyViewerMenuMorph`와 `EtoyViewerMenuItemMorph`로 확장하였다. 카테고리를 변경하는 메뉴를 열 때 해당 객체의 카테고리별 명령어를 한번만 수집하여 풍선 도움말 형식으로 보여주었다.

- Class name : `CategoryViewer`, `EtoyViewerCustomMenu`, `EtoyViewerMenuMorph`, `EtoyViewerMenuItemMorph`

## 6.4 Examples

제출된 스크 이토이를 실행하면 확인할 수 있다.

## 참고 문헌

- [1] Algo squeak korea. <http://algo.squeak.kr>.
- [2] ALLEN-CONN, B., AND ROSE, K. *Powerful Ideas in the Classroom*. Viewpoints Research Institute, 2003.
- [3] Inc lab. department of computer science education, graduate school, korea university. <http://inc.korea.ac.kr>.
- [4] KAY, A. Children learning by doing squeak etoys on the olpc xo. VPRI Research Note, 2007.
- [5] KELLEHER, C., AND PAUSCH, R. Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys* 37, 2 (June 2005), 83–137.
- [6] Korean squeak user group. <http://squeak.kr>.
- [7] One laptop per child (olpc) website. <http://laptop.org/>.
- [8] PANE, J. F., AND MYERS, B. A. Usability issues in the design of novice programming systems. Tech. rep., CMU Human-Computer Interaction Institute, Aug 1996.
- [9] PAPERT, S. *Mindstorms: children, computers, and powerful ideas*. Basic Books, Inc., 1980.

- [10] PETRE, M. Programming paradigms and culture: implications of expert practice. In *ProgrammingLanguage Choice:PracticeandExperience*. International Thomson Computer Press, 1996, pp. 29–44.
- [11] Squeakland website. <http://squeakland.org/>.