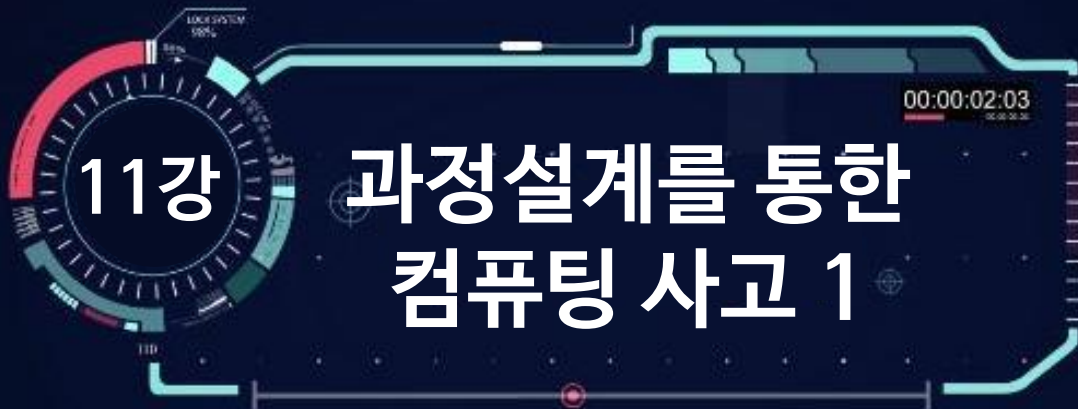


컴퓨팅 사고



고려사이버대학교
THE CYBER UNIVERSITY OF KOREA



1 차시

컴퓨팅 사고 기초 교육

2 차시

기초 교육과정

3 차시

기초 교육과정 설계 예제



1차시

컴퓨팅 사고 기초 교육

생각을 여는 질문

컴퓨팅 사고 기초 교육을 할 때는 어떤 점을
주의해야 할까요?

기초 교육과정 설계의 특징은 무엇이고
유의점은 무엇일까요?

- 컴퓨팅 사고를 위한 **기초 교육과정의 설계 시 고려할 점과 주의점**을 설명할 수 있다.
- 기초 교육과정의 **연령대별 교육목표**를 이해하고 프로그램 진행 시의 **유의점**을 말할 수 있다.

1 컴퓨팅 사고를 위한 기초 교육과정 설계

1 ▶ 연령과 언어는 설계의 첫번째 요소

- 모든 과목에는 연령에 적합한 첫 단계가 존재
- 그 출발점은 바로 언어. 예를 들어 ‘함수’ ‘조건문’ 같은 단어는 어린이를 대상으로 적합한 단어는 아님
→ 이 경우 다양한 상황에서 반복적으로 사용하며 이 분야의 언어를 자연스럽게 익히게 함



같은 연령이라도 같은 출발점이 아닙니다.
컴퓨터 과학의 경우 다른 학문보다 이 폭이 심합니다.

1 컴퓨팅 사고를 위한 기초 교육과정 설계

2 짝 혹은 그룹작업을 기본으로 설계하기

- 짝 작업은 한 명의 내비게이터와 한 명의 운전자로 조직
- 그룹작업을 진행할 수도 있음
- **공동 작업**은 자율적이면서도 문제해결적이고, 주석 달기에도 도움이 됨. 특히 이 방법은 텍스트 기반 코딩을 배울 때 효과가 큼

1 컴퓨팅 사고를 위한 기초 교육과정 설계

2 짝 혹은 그룹작업을 기본으로 설계하기

짝 프로그래밍 : 여러 사람이 하나의 코드를 같이 작성



〈출처〉 <https://thenounproject.com>

운전자

프로젝트를 완성하기 위해
키보드와 마우스를 이용해서
타이핑을 하거나 코드를 짤

내비게이터

프로젝트의 큰 그림에 집중하고
작성된 코드가 **논리적인지 확인**
함

1 컴퓨팅 사고를 위한 기초 교육과정 설계

2 짝 혹은 그룹작업을 기본으로 설계하기

짝 프로그래밍 : 여러 사람이 하나의 코드를 같이 작성

- 보통 두 명이 짝을 지어 컴퓨터 1대로 코딩을 함
- 짧은 시간 동안 자신의 역할을 수행하고 서로 역할을 바꿈
- 역할을 수행할 때 시간 제한이나 해결해야 할 문제의 개수, 그 밖의 측정기준을 만듦

1 컴퓨팅 사고를 위한 기초 교육과정 설계

3 컴퓨팅 사고는 ‘학습하는 방법’을 배우는 것

- 효과적인 컴퓨터과학 학습은 **탐구기반 학습**이어야 함
탐구기반 학습 : 계획, 실험, 수정의 과정 포함
- 실패할 수 있는 기회와 시간을 줄 것, 컴퓨터과학에서는 실패를 어떻게 다루는지 알려주는 것. 실패는 당연한 것
실패했을 때 불편한 마음은 당연한 것
포기하지 않는 태도를 긍정적으로 칭찬하기

1 컴퓨팅 사고를 위한 기초 교육과정 설계

4 수업의 준비는 아이디어를 현실화하는 것이 중심

- 컴퓨터 과학은 일상적인 암기는 없음. **학생들의 아이디어를 현실화하며 코딩으로 표현하는 것 뿐임**

수업 중에 인터넷, 폰 사용도 가능함

연령이 낮으면 힌트나 학습요령을 인쇄물로 배포

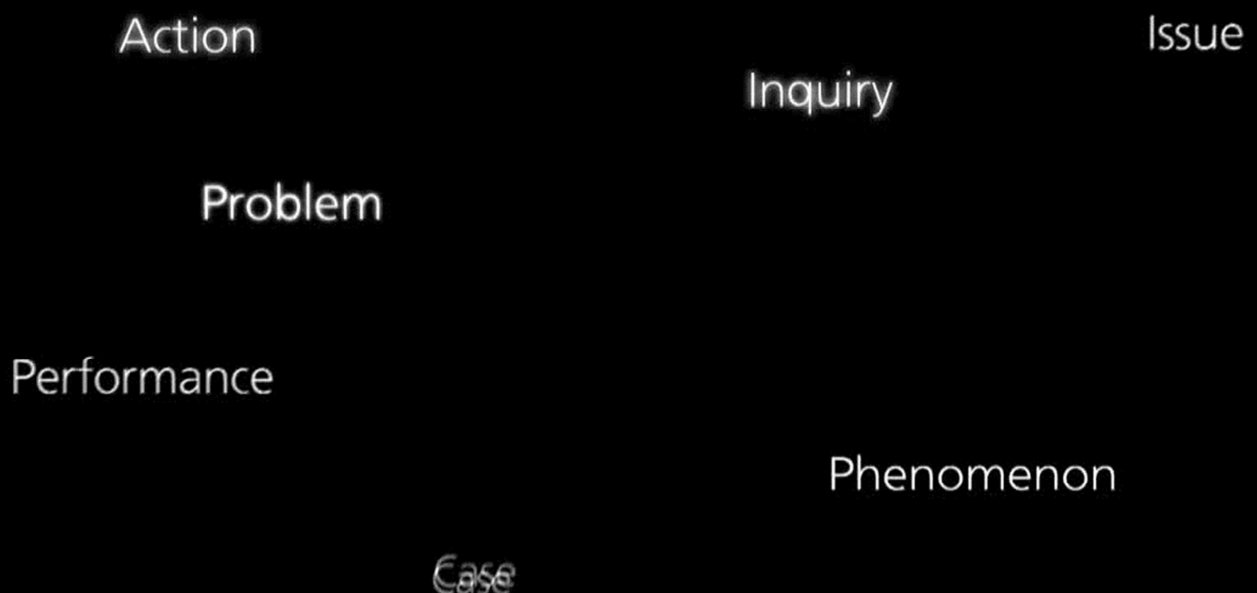
연령이 높으면 협동 사이트를 개발

- 몇가지 규칙으로 정하기(산만해 지지 않도록)

연필과 종이로 이야기하기

정답을 요구하는 질문 금지

학생 주도형 PBL(Problem-based learning) 교육



〈출처〉 <https://www.youtube.com/watch?v=JH4GIbnHNyl>

2 기초 교육과정의 특징

1. 연령별 특징 : 유치원/1학년

교육목표 | 컴퓨터 과학은 흥미롭고 재미있다.

- 집중할 수 있는 시간이 짧음. 대신 여러 경험을 강렬하게 받아들임 → 짧고 반복적인 형태로 새로운 경험을 진행
- 타인의 감정을 공감하기 어려움 → 캐릭터를 조정하는 종류의 코딩환경은 적절치 않음
- 언어 및 개념 세우기, 논리 연습하기, 시퀀스 생성하기 등의 주제로 언플러그드 교육활동이 좋음

2 기초 교육과정의 특징

1. 연령별 특징 : 2/3학년

교육목표 | 컴퓨터 과학은 내가 할 수 있고 즐길 수 있는 것이다.

- 본격적인 블록 코딩 가능. 한 세션당 30분 정도로 일주일에 1~2세션이 적당함
- 2~3줄의 코드가 실행될 때 자신의 캐릭터가 어떻게 움직일지 예측 가능. 바른 순서로 재구성 가능
- 끈기가 부족하고 지시문을 잘 읽지 못하며, 금방 산만해짐
→ 학생 스스로 자신의 능력을 조정해서 즐길 수 있게 해줌

2 기초 교육과정의 특징

1. 연령별 특징 : 4/5학년

교육목표 | 컴퓨터 과학은 내가 무엇인가 개발할 때 사용하는 것이다.

- 목적을 이루는 도구로서의 컴퓨터 과학이 본격화
- 컴퓨터를 컴퓨터과학의 도구로 본격적으로 사용(스크래치)
- 블록기반 → 텍스트 기반 프로그래밍 전환 유도
- 중학교때 진행할 본격적인 연계교육들을 위해 스스로 프로젝트를 할 수 있는 힘을 길러줘야 함
- 인터넷 안전 및 디지털 시민의식 교육 필요

2 기초 교육과정의 특징

2. 기초 교육과정 프로그램 진행 시 유의할 점

1 간단히 준비할 수 있는 수업을 선택하기

- 언플러그드 수업으로 시작. 기능을 가르치는 온라인 게임으로. 그 다음 컴퓨팅 사고의 개념을 강화하는 프로그램으로 마무리
- 학생들이 능동적으로 참여할 수 있도록 유도. 각 활동 및 예비 활동에 필요한 시간이 얼마인지 미리 알려주기

2 기초 교육과정의 특징

2. 기초 교육과정 프로그램 진행 시 유의할 점

2 중간에 시작해도 할 수 있는 수업을 디자인하기

3 집에서 숙제를 할 수 없는 환경이니 수업시간에 마무리 하기

- 과제는 현실적이지 않음. 1차시로 끝나지 않는다면 다음주 수업분으로 넘겨서 모든 학생이 같이 할 수 있도록 해주기

2 기초 교육과정의 특징

2. 기초 교육과정 프로그램 진행 시 유의할 점

4 지루해 하는 경우가 많으니 매주 환경이나 주제를 바꾸기

5 학생들의 최대 집중시간은 30분. 세션을 나누거나 수업을 짧게하기

2 기초 교육과정의 특징

2. 기초 교육과정 프로그램 진행 시 유의할 점

6 짝, 그룹 프로그래밍은 필수. 소리내며 작업할 수 있게 하기

7 짧고 독립적인 프로젝트로, 긴 프로젝트는 세션을 나누어 하기

- 1주일에 한번이라면 5주간 진행 시 한번에 4~5시간뿐
- 학생들이 작품을 완성하기에는 몹시 부족한 시간
- 매주 짧고 독립적인 프로젝트를 진행하기

2 기초 교육과정의 특징

2. 기초 교육과정 프로그램 진행 시 유의할 점

8 언어나 지시문이 없는 인터페이스를 사용하기

- 유치원 및 초등과정의 경우 읽고 쓸 수 없는 경우의 학생도 있음. 언어를 안 쓰는 Code.org의 코스 1로도 1년치 교육 과정 가능. 혹은 코드스파크의 Foos도 가능함

9 프로젝트냐 수업이냐 보다 중요한 것은 '즐거움'

활동 과제

다음 질문에 대해 생각해 보고 게시판에서 의견을 공유해 보세요.

Q

수업이 산만해 지지 않도록 하기 위해 여러분이 교사입장에서 학생들과 수업 중에 만들 수 있는 약속은 무엇이 있을까요?

또한 어떤 약속을 만들어야 효과적인 수업이 진행될까요?

Quiz!

Q. 기초 교육과정의 과정 설계 시 목표로 맞는 것은?

- 1 프로젝트나 수업보다 중요한 것은 ‘즐거움’이다.
- 2 유치원/1학년 연령대의 목표는 컴퓨터와 친숙해지고 컴퓨터 환경에 익숙해 지는 것이다.
- 3 4/5학년 연령대의 목표는 컴퓨터 과학은 나 스스로 할 수 있고 즐길 수 있는 것이라는 것을 깨닫는 것이다.
- 4 2/3학년 연령대의 목표는 컴퓨터 과학은 내 목표를 이루는 도구라는 것을 깨닫는 것이다.

기초 교육과정 수업 설계 시 고려할 점은 다음과 같다.

1. **연령과 언어**
2. **짝 혹은 그룹 작업으로 설계**
3. **학습하는 방법을 중심으로 설계**
4. **아이디어를 현실화 하는 것을 설계**

컴퓨팅 사고의 기초 교육 설계 시의 교육목표는 다음과 같다.

유치원/1학년 연령 : **컴퓨터 과학은 흥미와 재미**

2/3학년 연령 : **컴퓨터 과학은 내가 할 수 있고 즐길 수 있는 것**

4/5학년 연령 : **컴퓨터 과학은 목적을 이루는 도구**

참고자료

- ➔ 컴퓨팅 사고력과 코딩교육. 제인클라우드 외 1인. 한빛아카데미. 2017년
- ➔ 학생 주도형 PBL교육
<https://www.youtube.com/watch?v=JH4GIbnHNyl>

2차시

기초 교육과정



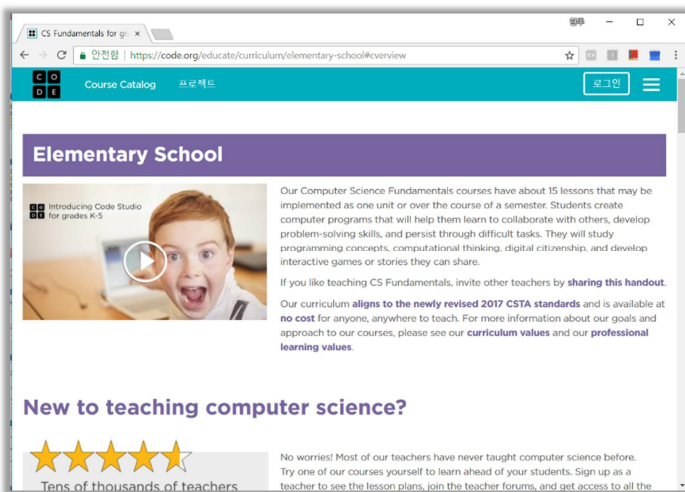
생각을 여는 질문

기초 교육과정의 예로는 어떤 것들이 있을까요?

- 기초 교육과정을 가지고 있는 **교육용 사이트들을 살펴보고**, 교육적 특징을 **비교하고 분석**할 수 있다.

1 Code.org 교육과정

Code.org의 CS기초 섹션



- Code.org내에 CS의 연령별 섹션이 마련되어 있음
- 주로 협동과 흥미 위주의 소재
- 결과의 공유 요소가 들어 있음

<https://code.org/educate/curriculum/elementary-school#cverview>

1 Code.org 교육과정

Code.org의 CS기초 섹션

- 미국 교육부 기준 K5 대상의 초등학교용 CS교육안과 동영상이 들어 있음
- 학생들이 서로 협동하고 문제 해결 능력을 기르며 어려운 과제를 통해 끈기를 기를 수 있는 컴퓨터 프로그램을 개발
- 이 코스가 끝나면 학생들은 자신만의 게임을 만들거나 이야기를 만들어서 서로 공유할 수 있음

1 Code.org 교육과정

Code.org의 CS기초 섹션

Kindergarten	1st Grade	2nd Grade	3rd Grade	4th Grade	5th Grade
Course A	Course B	Course C	Course D	Course E	Course F
Pre-Reader Express Course		Express Course			

1 Code.org 교육과정

Code.org의 CS기초 섹션

Kindergarten	1 st Grade	2 nd Grade	3 rd Grade	4 th Grade	5 th Grade
Course A	Course B	Course C	Course D	Course E	Course F
Pre-Reader Express Course		Express Course			

Course A

Recommended for Kindergarten (Supports pre-readers)

Teaches basic programming concepts such as loops and events. Lessons also teach students to collaborate with others meaningfully, investigate different problem-solving techniques, persist in the face of difficult tasks, and learn about internet safety. At the end of this course, students create their very own custom game or story they can share.

[View course](#)

[Lesson plans](#)

Course B

Recommended for 1st grade (Supports pre-readers)

Closely parallels Course A, but provides more complex unplugged activities and more variety in puzzles. Covers the basics of programming, collaboration techniques, investigation and critical thinking skills, persistence in the face of difficulty, and internet safety. At the end of this course, students create their very own custom game they can share.

[View course](#)

[Lesson plans](#)

1 Code.org 교육과정

Code.org의 CS기초 섹션

Course A

- 유치원에 권장
- 루프 및 이벤트와 같은 기본 프로그래밍의 개념을 가르침. 수업을 통해 공동 작업을 하고 문제 해결을 배우도록 함
- 끝나면 자신만의 맞춤 게임이나 스토리텔링을 만들 수 있음

Course B

- 1학년에게 추천
- 코스 A와 유사하지만 보다 복잡하고 분리된 활동과 퍼즐을 제공함
- 협업 조사 및 비판적 기술, 어려운 난이도에서 반복작업, 인터넷 안전에 대한 기본 사항을 배움
- 끝나면 공유가능한 맞춤 게임을 만들

1 Code.org 교육과정

Code.org의 CS기초 섹션

Kindergarten	1 st Grade	2 nd Grade	3 rd Grade	4 th Grade	5 th Grade
Course A	Course B	Course C	Course D	Course E	Course F
Pre-Reader Express Course		Express Course			

Course C

Recommended for 2nd grade

Students will create programs with loops and events. They will translate their initials into binary, investigate different problem-solving techniques, and discuss how to respond to cyberbullying. By the end of the course, students will create interactive games that they can share.

[View course](#)[Lesson plans](#)

Course D

Recommended for 3rd grade

Students develop their understanding of algorithms, nested loops, while loops, conditionals, and events. Beyond coding, students learn about digital citizenship.

[View course](#)[Lesson plans](#)

1 Code.org 교육과정

Code.org의 CS기초 섹션

Course C

- 2학년에게 추천
- 루프와 이벤트가 있는 프로그램을 직접 만들고 이니셜을 바이너리로 변환하는 법을 배움. 다양한 문제 해결 기법을 조사하며 사이버 괴롭힘에 대응하는 법을 배움
- 끝나면 공유가능한 대화형 게임을 만들

Course D

- 3학년에게 추천
- 알고리즘, 중첩된 루프와 일반적인 루프, 조건식과 이벤트에 대해 배움
- 디지털 시민권에 대한 개념을 배움

1 Code.org 교육과정

Code.org의 CS기초 섹션

Kindergarten	1 st Grade	2 nd Grade	3 rd Grade	4 th Grade	5 th Grade
Course A	Course B	Course C	Course D	Course E	Course F
Pre-Reader Express Course		Express Course			

Course E

Recommended for 4th grade

Students will practice coding with algorithms, loops, conditionals, and events before they are introduced to functions. In the second part of the course, students design and create a capstone project they can share.

[View course](#)[Lesson plans](#)

Course F

Recommended for 5th grade

Students create programs with different kinds of loops, events, functions, and conditionals. They will also investigate different problem-solving techniques and discuss societal impacts of computing and the internet. In the second part of this course students design and create a capstone project they can share.

[View course](#)[Lesson plans](#)

1 Code.org 교육과정

Code.org의 CS기초 섹션

Course E

- 4학년에게 추천
- 알고리즘, 루프와 조건식 및 이벤트를 통한 코딩을 연습하고 함수를 배움
- 학생들은 캡스톤 프로젝트를 디자인하고 만들 수 있음

Course F

- 5학년에게 추천
- 학생들은 여러 종류의 루프, 이벤트, 조건 프로그램을 만들고 다양한 문제 해결기법을 사용할 수 있음. 컴퓨터와 인터넷의 사회적 영향에 대해 토론
- 공유할 수 있는 캡스톤 프로젝트를 설계하고 만들

2 다양한 기초 교육과정

1) 코드스파크



- 초등 대상의 연령층을 타겟으로 게임 메이커와 학습, 커리큘럼이 구성되어 있음
- 사이트 내 학습은 글자 없이 구성

<https://codespark.com/>

2 다양한 기초 교육과정

1) 코드스파크

- 학습용 게임 메이커 및 학습게임, 커리큘럼이 제공됨
- 코드스파크는 컴퓨터 과학 교육을 위한 사이트로, 최신연구와 프로토타입을 기반으로 학습 게임을 만듦
- 글자가 전혀 없으며 K5 중심으로 교육과정을 진행할 수 있음

2 다양한 기초 교육과정

2) 비츠박스

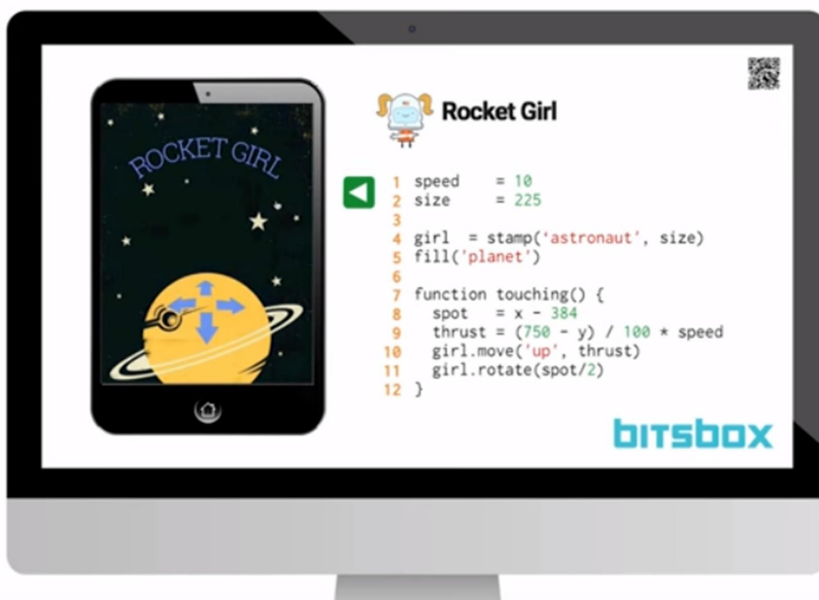


- 자바스크립트 언어 사용
- 코드를 수정하거나 명령어를 바꾸는 형태로 학습을 진행함
- 바로 본인 기기에 빌드할 수 있음

<https://bitsbox.com/>

2 다양한 기초 교육과정

2) 비츠박스





Coding for Kids – Bitsbox Review



〈출처〉 <https://www.youtube.com/watch?v=scjyAF7onGw>

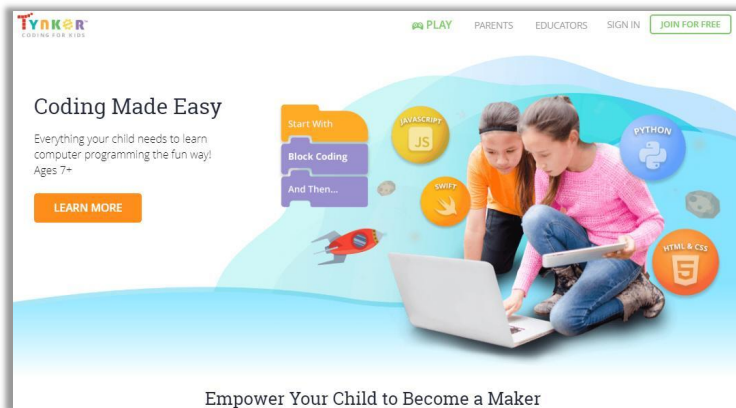
2 다양한 기초 교육과정

2) 비츠박스

- 자바스크립트 언어를 이용해서 어플이나 게임 제작 가능
- 스스로 프로그램이나 앱을 만들면서 코드를 익히고 프로그래밍을 배울 수 있게 설계된 에듀테인먼트
- 웹사이트에 연결되어 코딩공간과 가상의 태블릿을 제공하고 코드를 수정하거나 명령어를 바꾸면 앱을 변경할 수 있음

2

3) 텅커

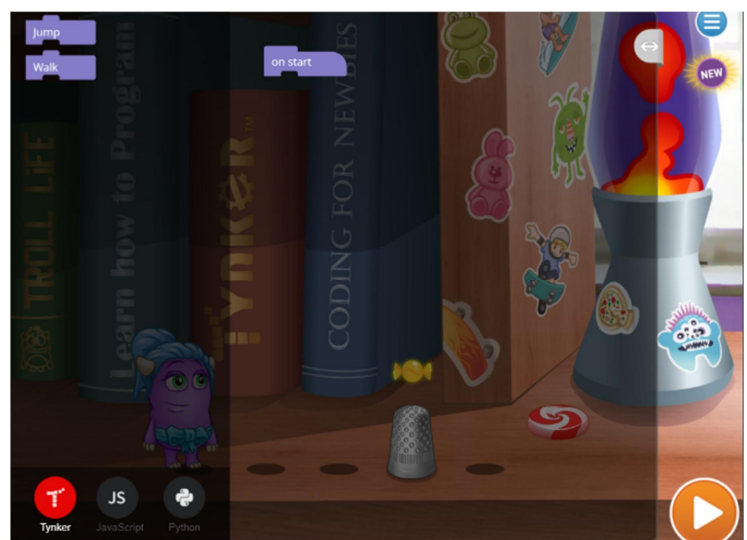


- 블록 코딩 기반형 학습도구
- 교육과정에 맞춰서 학습 스텝부터 도구, 평가요소에 이르기까지 셋트로 구성되어 있음

<https://www.tynker.com>

2

3) 텅커

[illegible]

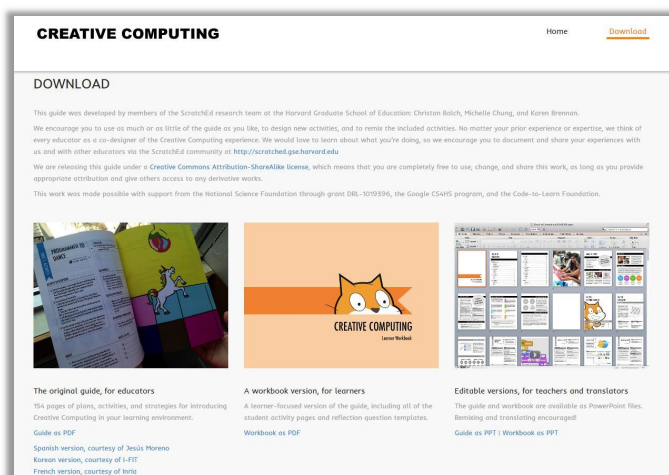
2 다양한 기초 교육과정

3) 퉁커

- 초등과정부터 K8까지 커리큘럼에 맞춰진 블록 코딩도구
- 상호작용 연습문제, 튜토리얼, 재미있는 창의도구와 퍼즐, 프로그래밍 가능한 학습도구로 이루어져 있음
- 학습 스텝부터 도구, 평가요소에 이르기까지 총 3개로 구성 되어 있음

2 다양한 기초 교육과정

4) 스크래치 창의컴퓨팅



- 하버드와 스크래치 공동제작
- 스크래치를 이용해서 각 연령 별, 효과별로 커리큘럼을 작성

<http://scratched.gse.harvard.edu/guide/download.html>

2 다양한 기초 교육과정

4) 스크래치 창의컴퓨팅



- 스크래치 창의 컴퓨팅의 한국어 번역 사이트

<http://computing.or.kr/창의컴퓨팅-가이드북>

2 다양한 기초 교육과정

4) 스크래치 창의컴퓨팅

- 하버드에서 제작한 것으로, 스크래치를 이용해서 각 연령별 (초등학생부터 대학원생까지), 효과별로 공부할 수 있는 커리큘럼을 작성해 놓음
- 추가적인 효과 및 자료를 위해 학습자 및 학생용 교안을 제공
- 국내에서는 번역본 및 번역 사이트가 있음

활동 과제

다음 질문에 대해 생각해 보고 게시판에서 의견을 공유해 보세요.

Q

여러분들이 알고 있는 기초 교육과정에 적용할 수 있는 교육용 사이트가 있나요?

관련 사이트를 찾아보고 해당 사이트의 특징과 내용을 정리해 봅시다.

Quiz!

Q. 기초교육과정에서 살펴본 내용과 가장 잘 맞는 것은?

- 1 퉁커-게이미메이커형으로 사이트 내 학습은 글자없이 구성됨
- 2 비츠박스 블록코딩 기반형으로 교육과정부터 학습 스텝 까지 셋트로 구성되어 있음
- 3 협동과 흥미위주 소재이고 결과를 공유할 수 있음
- 4 코드스파크 자바스크립트 언어로 코드 수정으로 학습을 진행함

기초 교육과정 예시로 Code.org의 CS기초 섹션, 코드스파크, 비츠박스, 텅커, 스크래치 창의컴퓨팅의 교육과정과 특징을 살펴보았다.

참고자료

- ➔ 컴퓨팅 사고력과 코딩교육. 제인크라우드 외 1인. 한빛아카데미. 2017년
- ➔ Coding for Kids - Bitsbox Review
<https://www.youtube.com/watch?v=scjyAF7onGw>
- ➔ Code.org의 CS기초 섹션
<https://code.org/educate/curriculum/elementary-school#cverview>

참고자료

- ➔ 코드스파크
<https://codespark.com>
- ➔ 비츠박스
<https://bitsbox.com>
- ➔ 텅커
<https://www.tynker.com/school/courses/index>
- ➔ 스크래치 창의컴퓨팅
<http://scratched.gse.harvard.edu/guide/download.html>



3차시

기초 교육과정 설계 예제

생각을 여는 질문

기초 교육과정을 위한 수업의 진행은
어떤 특징이 있을까요?

- 기초 교육과정을 위한 예제를 같이 진행해보고, 해당 예제를 이용해 **나만의 수업지도안**을 제작해 본다.

1 기초 교육과정 예제

Chase Crazy Legend



Build an Alligator Challenge!



2 기초 교육과정 수업지도안

기본 정보

- 1 프로그램 : 코드스파크의 The Foos
- 2 연계 교육과정 접목 : 디자인 사고
- 3 대상 연령 : 만 세
- 4 소요시간 분

2 기초 교육과정 수업지도안

개요

이 수업은 간단한 시간 게임을 만들어 봄으로써 학생들의 프로그래밍 지식을 적용하도록 유도합니다. 서로 다른 명령어를 루프 안에 순서대로 나열해 보고, 코드의 작은 변화가 컴퓨터의 동작에 어떤 영향을 미치는지, 게임이 얼마나 재미있어지는지 알아보게 하세요. 학생들은 이런 패턴을 새로운 문제 상황에 적용할 수 있게 됩니다.

2 기초 교육과정 수업지도안

용어

- **시퀀스** : 일련의 명령어 집합 속에 있는 단계별 순서
- **루프** : 일련의 명령어 집합이 계속 반복되도록 하는 명령어
- **매개변수** : 명령어의 어떤 동작을 변경하는 설정

2 기초 교육과정 수업지도안

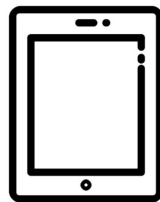
수업 목표와 준비물

- 특정한 동작을 명령할 수 있는 단계 설정하기
- 루프를 사용해 동작 반복시키기
- 코드 수정으로 결과가 어떻게 변하는지 설명하기
- 디자인 선택에 대해 탐색하고 설명하기

2 기초 교육과정 수업지도안

준비물과 자료

- The Foos가 설치된 태블릿
- 칠판이나 프로젝터



Created by IconTrack
from Noun Project



Created by Danishicon
from Noun Project

2 기초 교육과정 수업지도안

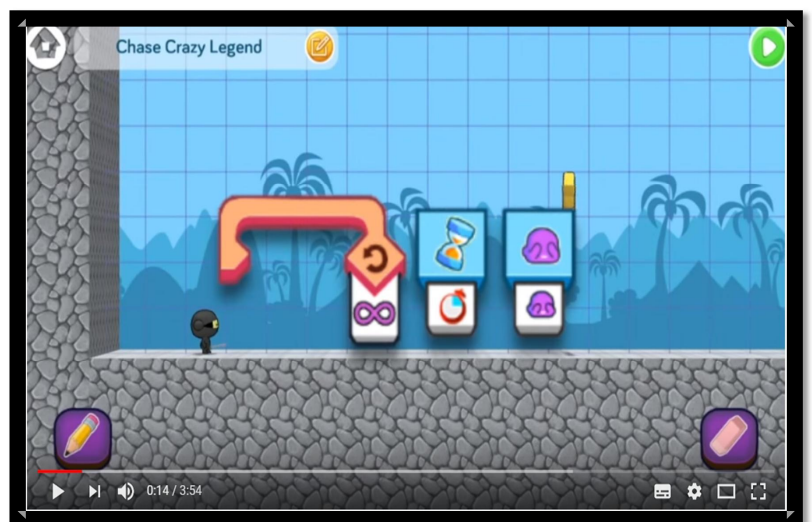
준비

- 1 지도안 읽기
- 2 짧은 비디오 클립
- 3 푸 스튜디오에 새로운 레벨을 만들고 비디오에 안내된 지시 따르기
- 4 학습지를 인쇄하거나 다운로드 하고 칠판이나 프로젝터를 통해 보여주기

2 기초 교육과정 수업지도안

활동 / 5단계

- 1단계 : 소개하기
컴퓨터 개념, 명령어,
아이콘을 보여주고
무슨 뜻인지 설명함



2 기초 교육과정 수업지도안

활동

- 2단계 : 본 활동

학생들에게 악어 배치
영상을 보여주고
만들게 함



2 기초 교육과정 수업지도안

활동

- 3단계 : 공유하기 & 4단계 : 토론하기

- ✓ 이 활동에서 가장 어려운 부분은 무엇이었나요?
- ✓ 왜 작업 시 대기 명령어를 두 번 사용했나요?

2 기초 교육과정 수업지도안

활동

- 3단계 : 공유하기 & 4단계 : 토론하기

✓ 어떤 것이 가장 재미있을까요? 어떤 것이 사용자에게 가장 불리할까요?

- ① 이빨이 빨리 나타나고 사라지기
- ② 이빨이 나타났다가 다시 나타나지 않음
- ③ 이빨이 빠르게 사라짐
- ④ 이빨이 사라지고 나타나기를 세번 반복
- ⑤ 이빨이 길게 나타남

2 기초 교육과정 수업지도안

활동

- 5단계 : 실생활에서 연결해 수업 마무리

- ✓ 보이기, 숨기기, 대기하기는 컴퓨터 과학 여러 곳에서 사용됨
- ✓ 동영상의 경우에도 플레이 버튼 멈춤 버튼으로 동일한 기능을 함
- ✓ 학생들에게 대기 명령이 사용되는 앱이나 게임을 찾아 보게 함

활동 과제

다음 질문에 대해 생각해 보고 게시판에서 의견을 공유해 보세요.

Q

앞서 학습한 예제를 다시 숙지해 봅시다. 이 중 좀 더 보강하거나 혹은 변경하고 싶은 부분이 있었나요?

해당 부분들을 이용해서 나만의 수업지도안을 제작해 봅시다.

Quiz!

Q. 기초교육과정 설계 예제의 내용으로 틀린 것은?

- 1 실 생활에 적용하는 단계에서는 프로젝트에서 한 명령어를 실제 사용하는 앱 찾아보기를 한다.
- 2 본 활동 단계에서는 직접 코드를 통해 만들어 본다.
- 3 공유-토론하기 단계에서는 활동의 어려웠던 부분을 서로 이야기 한다.
- 4 소개하기 단계에서는 악어 프로젝트의 영상을 보면서 만들 요소들을 따라해 본다.

기초 교육과정 설계 실습은 코드스파크의 The Foos 프로그램을 예제로 활용하여 **소개하기** - **본 활동** - **공유하기** - **토론하기** - **실 생활에서 적용하기** 로 수업지도안을 작성하는 학습을 진행하였다.

참고자료

- ➔ 컴퓨팅 사고력과 코딩교육. 제인크라우드 외 1인. 한빛아카데미. 2017년
- ➔ Build an Alligator Challenge!
<https://www.youtube.com/watch?v=YAvkRJdIFgs&feature>
- ➔ 코드스파크
<https://codespark.com>