

5장. 기본 연산

5.1 연산 블록(1/5)

- 수식 계산하기 위해서는 연산 기능 필요
- 산술연산, 관계연산, 논리연산
- 난수 생성

〈표 8.1〉 연산 블록

블록	기능
	주어진 두 피연산자에 대한 더하기 연산 결과를 얻는다.
	주어진 두 피연산자에 대한 빼기 연산 결과를 얻는다.
	주어진 두 피연산자에 대한 곱하기 연산 결과를 얻는다.
	주어진 두 피연산자에 대한 나누기 연산 결과를 얻는다.
	주어진 두 수 사이에서 임의의 수(난수)를 얻는다.
	왼쪽 수가 오른쪽 수보다 크면 참, 그렇지 않으면 거짓의 결과를 얻는다.
	왼쪽 수가 오른쪽 수보다 작으면 참, 그렇지 않으면 거짓의 결과를 얻는다.
	입력한 두 수가 같으면 참, 그렇지 않으면 거짓의 결과를 얻는다.
	논리곱 연산의 결과를 얻는다.
	논리합 연산의 결과를 얻는다.
	논리역 연산의 결과를 얻는다.

5.1 연산 블록(2/5)

연산 블록(계속)

- 문자열 결합, 문자열내의 문자 얻기, 문자열 길이, 문자열내에 특정 문자열 포함 여부 판별
- 나누어 나머지 구하기
- 반올림
- 절대값 등의 수학함수 연산

<input type="radio"/> 와(과) <input type="radio"/> 결합하기	입력한 두 문자열을 결합한 결과 문자열을 얻는다.
<input type="radio"/> 의 <input type="radio"/> 번째 글자	입력한 문자열에서 입력한 정수 번째 글자를 얻는다.
<input type="radio"/> 의 길이	입력한 문자열의 길이를 얻는다.
<input type="radio"/> 이(가) <input type="radio"/> 을(를) 포함하는가?	왼쪽에 입력한 문자열에 오른쪽에 입력한 문자열이 포함되어 있으면 참, 그렇지 않으면 거짓을 얻는다.
<input type="radio"/> 나누기 <input type="radio"/> 의 나머지	왼쪽에 입력한 정수를 오른쪽에 입력한 정수로 나눈 나머지를 얻는다.
<input type="radio"/> 의 반올림	주어진 실수에서 소숫점 이하의 수가 0.5 이상이면 반올림하여 정수를 얻는다.
절댓값 ▾ <input type="radio"/>	입력한 수에 대한 다양한 수학 함수(절대값, 버림, 올림, 제곱근, sin, cos, tan, asin, acos, atan, ln, log, e^x , 10^x) 계산 결과를 얻는다.

5.1 연산 블록(3/5)

[예제 5.1] 입력한 연산 종류와 두 피연산자에 대한 사칙 연산하기

- 덧셈, 뺄셈, 곱셈, 나눗셈 중에서 연산을 선택하고, 연산에 이용될 두 피연산자 값을 읽어 들여 연산을 수행하고 그 결과를 알려준다.
- 준비 단계
 - ① 연산 변수와 피연산자를 저장하기 위한 변수 a, b를 생성한다.
 - ② 스프라이트는 Devin을 이용하고 배경은 Light를 사용한다.
- 단계 1 : 묻고 기다리기를 이용하여 사용자로부터 덧셈, 뺄셈, 곱셈, 나눗셈 중 하나를 읽어 들이고, 역시 묻고 기다리기를 2번 이용하여 두 피연산자를 읽어 들인다.
- 단계 2 : 만약 연산이 덧셈이면 두 피연산자를 더하여 말한다.
아니면 만약 연산이 뺄셈이면 두 피연산자를 빼서 결과를 말한다.
아니면 만약 연산이 곱셈이면 두 피연산자를 곱해서 결과를 말한다.
아니면 만약 연산이 나눗셈이면 두 피연산자를 나누어 결과를 말한다.
아니면 "연산 종류가 잘못되었습니다."라고 말한다.

5.1 연산 블록(4/5)

[예제 5.1] 스크립트

스크립트	설명
	시작하기 버튼을 클릭했을 때 묻기 기다리기를 이용하여 사용자로부터 덧셈, 뺄셈, 곱셈, 나눗셈 중 하나를 읽어 들여 연산 변수에 저장한다. 역시 묻고 기다리기를 2번 이용하여 두 피연산자를 읽어 들어서 a, b 변수에 저장한다.
	중첩된 선택 구조를 이용하여(만약 < > 라면 ~ 아니면 ~ 블록을 중첩해서 사용) 연산이 덧셈이면 두 피연산자를 더하여 결과를 2초 동안 말한다. 만약 연산이 뺄셈이면 두 피연산자를 빼서 결과를 2초 동안 말한다. 만약 연산이 곱셈이면 두 피연산자를 곱해서 결과를 2초 동안 말한다. 만약 연산이 나눗셈이면 두 피연산자를 나눴서 결과를 2초 동안 말한다. 아니면 연산 종류가 잘못되었음을 말한다.

5.1 연산 블록(5/5)

[예제 5.1] 실행 화면



그림 8.1 [예제 8.1]의 실행 화면

연습

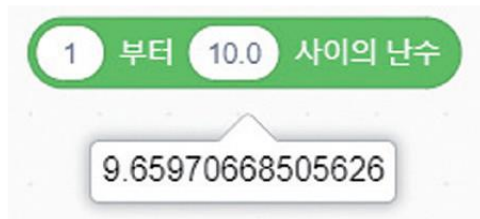
이차 방정식 $ax^2 + bx + c$ 의 실근의 개수를 구하는 스크립트를 작성하라.

힌트

판별식 $D = b^2 - 4ac$, $D > 0$ 이면 2개 실근, $D = 0$ 이면 1개의 중근, $D < 0$ 이면 허근(실근 0개)

5.2 난수 생성 블록(1/10)

- 난수(random number) : 임의의 수
- 게임을 만들 때 개체가 화면에 나타나는 위치나 시간 등을 임의로 정하고 싶을 때 사용
- 확률적인 시뮬레이션 프로그램에서 활용
- 주어진 상한과 하한이 정수이면 정수 값을 생성하며, 하나라도 를 입력하면 실수 값을 얻음
- 만약 1부터 10.0 사이의 난수를 얻으면 1.0부터 10.0 사이의 무수히 많은 실수 중에서 임의의 값을 얻는다




5.2 난수 생성 블록(2/10)

[예제 5.2] 주사위 던지기 시뮬레이션

- 주사위를 100번 던져서 3이 몇 번 나오는가를 실험하는 시뮬레이션 스크립트를 작성한다.
- 준비 단계
 - ① 발생 횟수 변수를 생성한다.
 - ② 스프라이트는 Dani를 고르고 배경은 Room 2를 이용한다.
- 단계 1 : 발생 횟수를 0으로 정한다.
- 단계 2 : 단계 3을 100번 반복한다.
- 단계 3 : 주사위 값을 얻기 위해서는 1에서 6 사이의 난수를 생성하면 된다.
이때 얻어진 값이 3이면 발생 횟수 변수를 1 증가시킨다.
- 단계 4 : 발생 횟수를 10초 동안 말한다.

5.2 난수 생성 블록(3/10)

[예제 5.2] 스크립트

스크립트	설명
	<p>시작하기 버튼을 클릭했을 때 스프라이트 크기를 70%로 줄인다. 위치를 (-60, -60)으로 이동한다. 발생 횟수를 0으로 정한다. 다음 단계를 100번 반복한다. 1에서 6 사이 난수를 생성하여 값이 3과 같으면 발생 횟수를 1 증가시킨다. 발생 횟수 결과를 10초 동안 말한다.</p>



5.2 난수 생성 블록(4/10)

[예제 5.3] 임의의 크기를 갖는 별 50개 만들기

- 실행 화면의 임의의 위치에 10에서 100 사이의 퍼센트(%) 크기로 0.5초 간격으로 별이 하나씩 총 50개 생성되는 프로그램이다.
- 준비 단계
 - ① 스프라이트 고르기에서 star를 선택한다.
 - ② 확장 기능 고르기에서 펜 기능을 선택한다.
펜 기능에서 도장 찍기 블록을 사용하면 스프라이트의 모양을 화면에 나타낼 수 있다.
- 다음을 50번 반복한다.
 - star 스프라이트를 무작위 위치로 이동하기 블록을 이용하여 이동시킨 다음, 10에서 100 사이의 난수를 얻어서 크기를 난수 값 %로 정한 다음 도장 찍기 블록을 사용한다.

5.2 난수 생성 블록(5/10)

[예제 5.3] 스크립트


스크립트	설명
	시작하기 버튼을 클릭했을 때 모두 지우기 블록으로 화면을 지운다. 다음 절차를 50회 반복한다. 무작위 위치로 이동한다. 난수를 발생시켜 별 크기를 난수 % 크기로 정한다. 도장 찍기를 한다. 0.5초 동안 기다린다.



그림 8.4 [예제 8.3]의 실행 화면

5.2 난수 생성 블록(6/10)

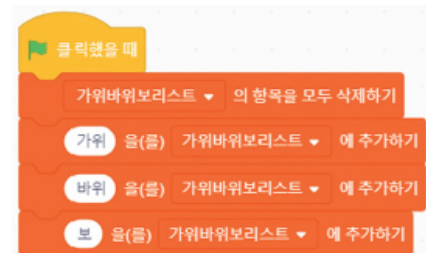
[예제 5.4] 가위바위보 게임

- 컴퓨터와 사용자가 가위바위보 게임을 한다.
 - 컴퓨터는 내부적으로 난수를 발생시켜 가위바위보를 결정한다.
 - 사용자는 가위, 바위, 보 중에서 하나를 선택한다.
 - 컴퓨터의 선택과 사용자의 선택을 비교하여 판정하여 말한다.
-
- 준비 단계
 - ① 가위바위보 리스트를 생성한다.
 - ② 컴퓨터 선택과 사용자 선택 변수를 생성한다.

5.2 난수 생성 블록(7/10)

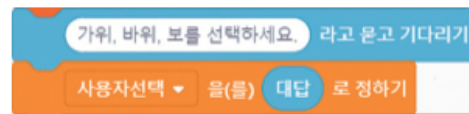
[예제5.4] 스크립트

단계 1



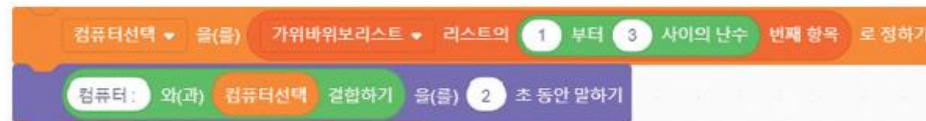
시작하기 버튼을 클릭했을 때
리스트의 이전 항목을 모두 삭제한다.
리스트에 가위, 바위, 보를 순서대로 추가한다.

단계 2



사용자로부터 묻고 기다리기로 가위, 바위, 보 중 하나
를 입력받는다.

단계 3



1에서 3 사이의 난수를 발생시켜서 그 수에 해당하는 가위바위보 리스트의 항목을 컴퓨터선택으로
정한다.
컴퓨터선택을 먼저 2초 동안 말한다.

단계 4



컴퓨터선택과 사용자선택이 같으면 “비겼습니다.”를 2초
동안 말한다.

5.2 난수 생성 블록(8/10)

[예제5.4] 스크립트

단계 5



컴퓨터선택과 사용자선택이 다른 모든 경우.

- 1) 컴퓨터선택 = 가위, 사용자선택 = 바위: 결과는 “이겼습니다.”
- 2) 컴퓨터선택 = 가위, 사용자선택 = 보: 결과는 “졌습니다.”
- 3) 컴퓨터선택 = 바위, 사용자선택 = 가위: 결과는 “졌습니다.”
- 4) 컴퓨터선택 = 바위, 사용자선택 = 보: 결과는 “이겼습니다.”
- 5) 컴퓨터선택 = 보, 사용자선택 = 바위: 결과는 “졌습니다.”
- 6) 컴퓨터선택 = 보, 사용자선택 = 가위: 결과는 “이겼습니다.”

5.2 난수 생성 블록(9/10)

[예제 5.5] 임의로 움직이는 쥐

- 쥐 스프라이트가 임의의 방향으로 임의의 속도로 움직이도록 한다.
 - 끊임없이 움직이도록 하기 위해서 무한 반복을 이용한다.
 - 반복 처리되는 절차는 먼저 방향을 변경하고, 임의의 거리만큼 움직인 다음, 임의의 시간만큼 기다린다.
-
- 준비 단계
 - ① 스프라이트 고르기에서 mouse1 스프라이트를 추가한다.
 - ② 무대 배경은 Forest로 바꾼다.
-
- 단계 1 : 스프라이트의 크기를 50%로 줄이고 위치는 무대 중앙으로 이동한다.
 - 단계 2 : 다음 단계 3, 4, 5를 무한 반복한다.
 - 단계 3 : 현재 방향에 -90에서 90 사이의 난수를 생성하여 더한다.
 - 단계 4 : 움직이는 거리를 1에서 50 사이의 난수만큼 움직인다.
 - 단계 5 : 0.2에서 1 사이의 난수만큼 기다린다.

5.2 난수 생성 블록(10/10)

[예제5.4] 스크립트



그림 8.5 [예제 8.5]의 스크립트와 실행 화면

5.3 논리 연산 블록(1/5)

- 논리 연산 : 논리곱(AND), 논리합(OR), 논리역(NOT)

〈표 8.2〉 논리 연산

P	Q	P OR Q	P AND Q	NOT P
참	참	참	참	거짓
참	거짓	참	거짓	거짓
거짓	참	참	거짓	참
거짓	거짓	거짓	거짓	참

- 블록의 형태가 육각형 형태, 블록은 참(true) 아니면 거짓(false) 값을 가짐

5.3 논리 연산 블록(2/5)

관계 연산

- 두 값이 같은가를 판단하거나 대소 관계를 판단하는 연산
 - 예) $3 < 4$ 관계 연산의 결과는 참이 되며, $2 > 5$ 관계 연산은 거짓이 됨. 또 $7 = 3$ 은 거짓
- 변수를 포함하는 관계 연산의 경우에는 변수에 따라 참이 되기도 하고 거짓이 되기도 하는 조건이 됨.
 - 예를 들어서 x 변수가 1에서 10 사이인가를 판별하기 위해서는 $x \geq 1$ AND $x \leq 10$ 이 조건이 된다.



- 선택구조의 조건이 될 경우



- 관계 연산자와 논리 연산자를 복합적으로 사용
 - 만약 나이가 18세 이상이고 남자인 조건 :



5.3 논리 연산 블록(3/5)

[예제 5.6] 성적에 의한 합격 판정

- 컴퓨터 성적이 90점 이상이면 합격, 컴퓨터 성적이 70점 이상 90점 미만인 경우는 토익 성적이 700점 이상일 때만 합격으로 처리
- 준비 단계
 - ① 컴퓨터 성적과 토익 성적 변수를 생성한다.
 - ② 스프라이트는 Dee, 배경은 Room2를 사용한다.
- 단계 1 : 묻고 기다리기를 2번 이용하여 컴퓨터 성적과 토익 성적을 각각 읽어 들어서 해당 변수에 저장한다.
- 단계 2 : 선택 구조를 이용하여 만약 컴퓨터 성적 ≥ 90 또는 (컴퓨터 성적 ≥ 70 그리고 토익 성적 ≥ 700)이면 합격 그렇지 않으면 불합격으로 판정한다.

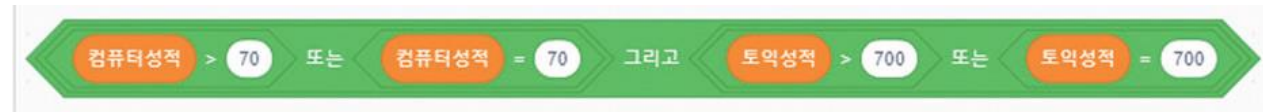
5.3 논리 연산 블록(4/5)

[예제 5.6] 조건 블록 구성

- 컴퓨터 성적 ≥ 90



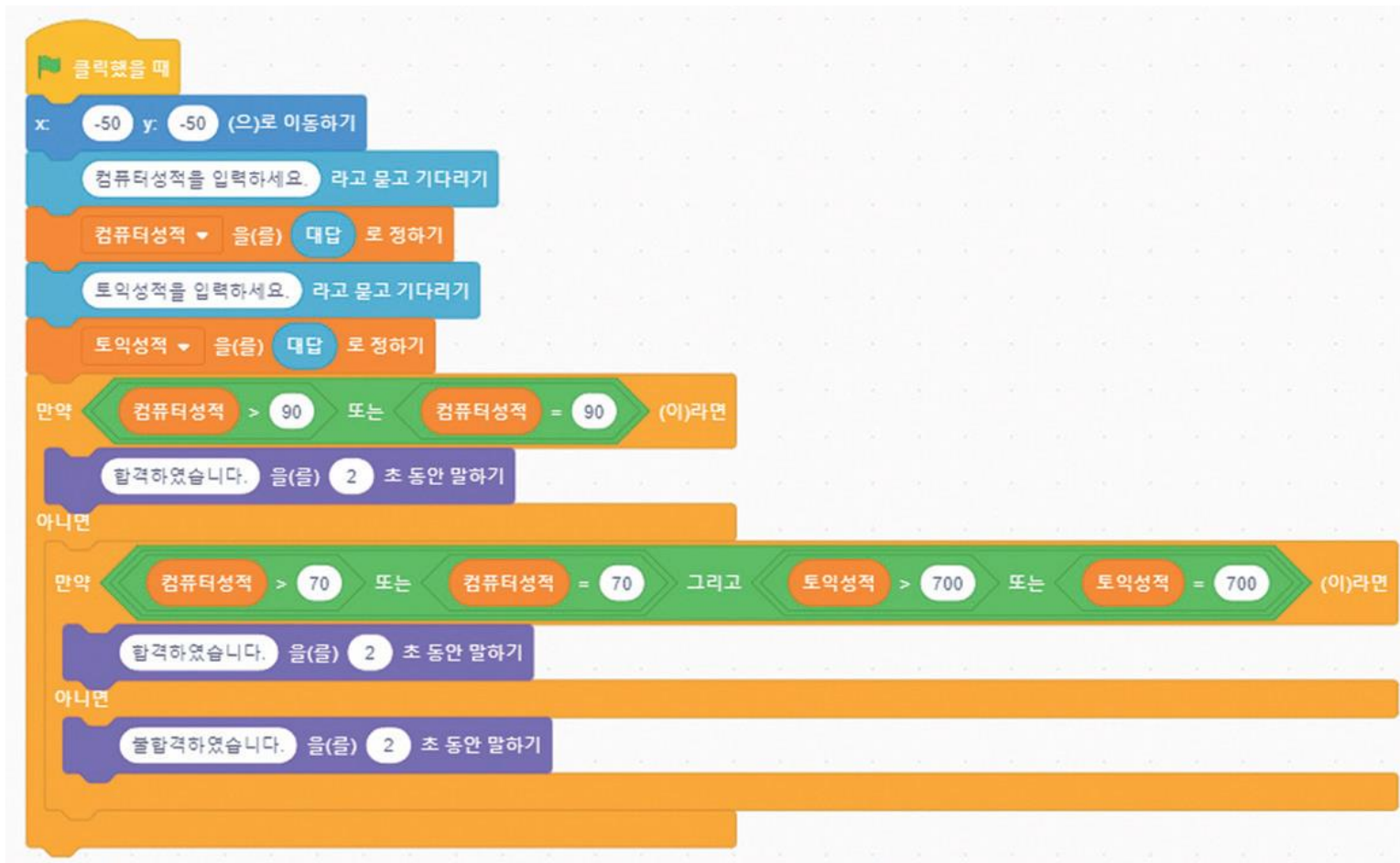
- 컴퓨터 성적 ≥ 70 과 토익 성적 ≥ 700



- 컴퓨터 성적 ≥ 90 과 (컴퓨터 성적 ≥ 70 그리고 토익 성적 ≥ 700)의 두 조건을 논리합(OR)으로 연결하여 하나의 조건으로 만들 수 있지만 블록 표현이 너무 복잡해서 이를 선택 구조를 2중으로 중첩시킴.

5.3 논리 연산 블록(5/5)

[예제 5.6] 스크립트



- 합격 판정 부분 다른 방법
- 선택구조 중첩시킴

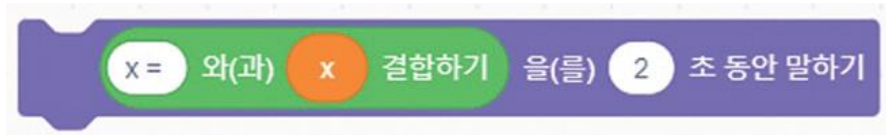


5.4 문자열 처리(1/7)

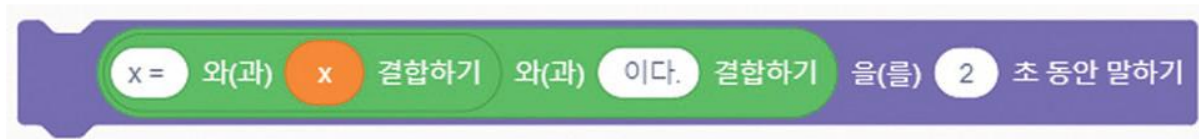
- 두 문자열을 결합하여 하나의 문자열로 만들 때



- x 변수의 값을 말하기 할 때 "x = " 문자열과 x 변수 값을 문자열 결합하기 블록을 이용하면



- "x = " 과 x 변수 값과 "이다." 를 결합하기(중첩된 결합하기)



5.4 문자열 처리(2/7)

[예제 5.7] 학교 학과 학번 이름을 읽어 들어서 하나의 문자열로 통합하기

- 준비 단계
 - ① 학교명, 학과명, 학번, 이름 출력 문자열 변수를 생성한다.
 - ② 소년 스프라이트를 위해서 Devin을 스프라이트 고르기에서 선택한다.
 - ③ 배경은 학교(School)를 이용한다.
- 단계 1 : 묻고 기다리기 블록을 이용하여 학교명을 읽어 들인다.
- 단계 2 : 묻고 기다리기 블록을 이용하여 학과명을 읽어 들인다.
- 단계 3 : 묻고 기다리기 블록을 이용하여 학번을 읽어 들인다.
- 단계 4 : 묻고 기다리기 블록을 이용하여 이름을 읽어 들인다.
- 단계 5 : 출력 문자열에 학교명과 학과명을 결합하여 저장하고 출력 문자열에 학번, 이름을 차례대로 결합한다.

5.4 문자열 처리(3/7)

[예제5.7] 스크립트

- 문자열 결합 절차
 - 출력문자열 <- 학교명 + ''
 - 출력문자열 <- 출력문자열 + 학과명 + ''
 - 출력문자열 <- 출력문자열 + 학번 + ''
 - 출력문자열 <- 출력문자열 + 이름



5.4 문자열 처리(4/7)

[예제 5.8] 단어의 역순으로 출력 단어 만들기

- 하나의 단어를 묻고 기다리기로 읽어 들어서 문자를 역순으로 말하는 스크립트를 작성
- 준비 단계
 - ① 출력 문자열, n 변수를 생성한다.
 - ② 스프라이트는 Dani를 선택하고, 배경은 Light를 이용한다.
- 단계 1 : 출력 문자열을 공백으로 정한다.
- 단계 2 : 묻고 기다리기 블록으로 단어를 읽어 들인다.
- 단계 3 : n을 대답의 길이로 정한다. 이는 대답의 마지막 문자의 위치값이 된다.
- 단계 4 : n이 0이 될 때까지 다음 단계 5, 6을 반복한다.
 - 단계 5 : n 위치에 있는 문자를 출력 문자열에 결합시킨다.
 - 단계 6 : n을 1 감소시킨다.
- 단계 7 : 출력 문자열을 2초 동안 말한다.

5.4 문자열 처리(5/7)

[예제 5.8] 스크립트

“대한민국” 단어를 입력하면 대답의 내용은 다음과 같다.

단어 :	대	한	민	국
위치 :	1	2	3	4

- 반복 과정에서 변수 값
n=4, 출력 문자열 = “국”
n=3, 출력 문자열 = “국민”
n=2, 출력 문자열 = “국민한”
n=1, 출력 문자열 = “국민한대”

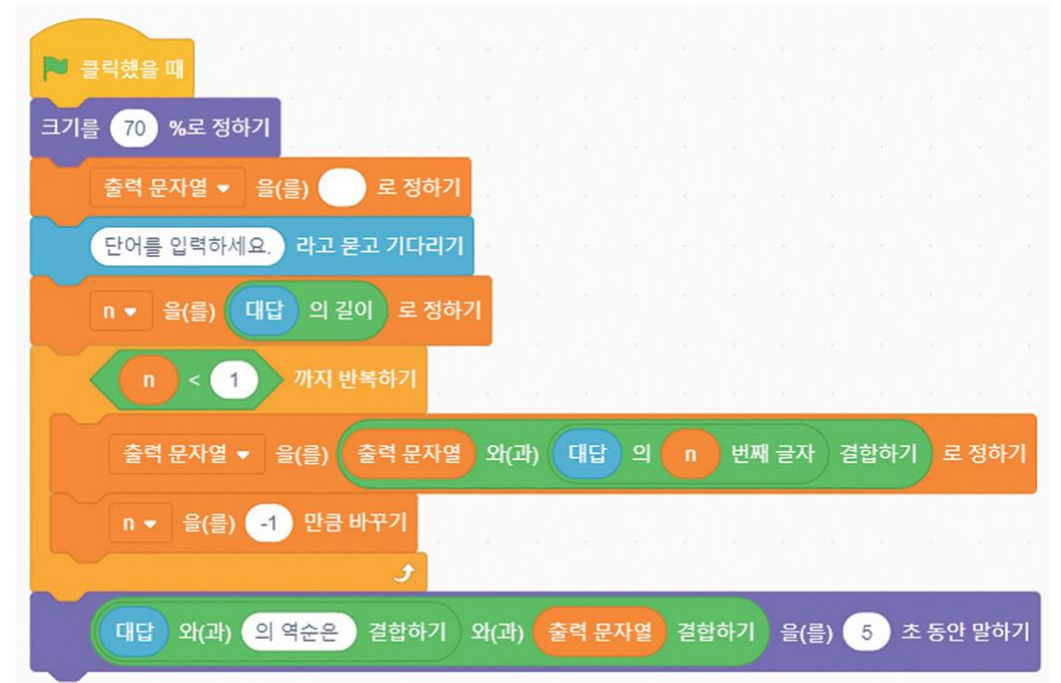



그림 8.8 [예제 8.8]의 실행 화면

5.4 문자열 처리(6/7)

특정 단어 포함 문장 말하기

연습

다음은 문장 중에 특정 단어가 포함되어 있는가를 알아내어 포함되어 있으면 그 문장을 말하게 하는 스크립트를 살펴보자. 문장에 특정 문자나 문자열이 포함되어 있는지를 알려주는  블록을 이용한다.

먼저 메모장을 열어서 간단한 편지글을 예로 작성하였다. 주의할 점은 한글 문서인 경우에는 저장할 때 유니코드로 저장해야 한다는 점이다. 메모장에서 기본적으로는 ANSI로 저장하도록 되어 있다. 그림 8.9는 메모장에서 편집된 글이다.

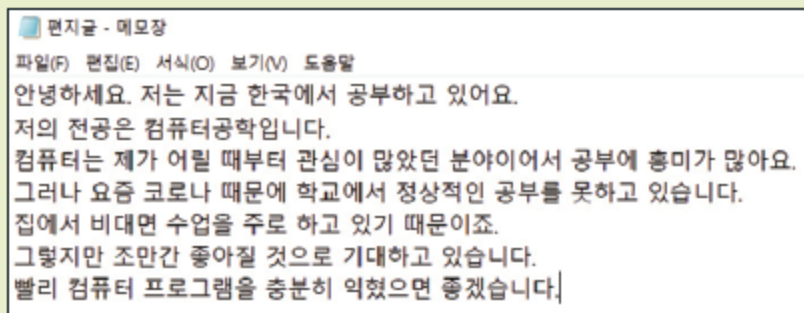


그림 8.9 메모장 문서의 예

5.4 문자열 처리(7/7)

연습 힌트

- 준비 단계
 - ① 편지글을 읽어 들어서 저장할 편지 리스트를 먼저 만든다.
 - ② 실행 화면에서 편지 리스트에서 마우스 오른쪽 버튼을 클릭하여 '불러오기'를 실행하여 해당 편지글.txt 문서를 읽어 들인다. 문서 내용에서 한 줄씩 리스트의 항목에 추가된다.
 - ③ i 변수를 생성한다.
- 단계 1 : i를 1로 정한다.
- 단계 2 : 편지 리스트의 길이만큼 다음 단계 3, 4, 5를 반복한다.
 - 단계 3 : 만약 편지 리스트의 i 번째 항목에 "컴퓨터"가 포함되어 있다면
 - 단계 4 : 편지 리스트의 항목을 2초 동안 말한다.
 - 단계 5 : i 변수를 1 증가시킨다.

5.5 수학 함수 연산 블록(1/8)

- 여러 가지 수학 함수(절댓값, 버림, 올림, 제곱근, sin, cos, tan, asin, acos, atan, ln, log, e^x , 10^x)를 계산하는데 사용할 수 있는 블록



그림 8.10 수학 함수 연산 블록

5.5 수학 함수 연산 블록(2/8)

[예제 5.9] 직각 삼각형의 밑변과 높이를 읽어 들어서 삼각형의 면적과 빗변의 길이를 구하기

삼각형의 면적 = $0.5 \times \text{밑변} \times \text{높이}$, 빗변의 길이 = $\sqrt{(\text{밑변}^2 + \text{높이}^2)}$ 공식을 이용하여 계산한다.

■ 준비 단계

밑변, 높이 변수를 생성한다.

단계 1 묻고 기다리기 블록을 2번 이용하여 밑변과 높이를 읽어 들인다.

단계 2 면적 공식을 이용하여 계산한 결과를 말한다.

단계 3 빗변의 길이 공식을 이용하여 계산한 결과를 말한다.

5.5 수학 함수 연산 블록(3/8)

[예제 5.9] 스크립트



5.5 수학 함수 연산 블록(4/8)

[예제 5.10] 두 정수를 읽어 들여서 나누어 몫과 나머지 구하기

■ 준비 단계

a, b, 몫, 나머지 변수를 생성한다.

단계 1 묻고 기다리기를 2번 이용하여 두 정수를 a, b 변수에 읽어 들인다.

단계 2 a를 b로 나누고 0.5를 더한 후, 버림을 한 다음 그 값을 말한다.(몫만을 구한다.)

단계 3 a를 b로 나눈 나머지를 구하여 말한다.



5.5 수학 함수 연산 블록(5/8)

연습

임의의 실수(소수점이 있는 실수)를 읽어 들여서 정수 부분과 소수점 이하의 실수 부분을 나누어서 각각을 말하는 스크립트를 작성하라.

실수에서 정수 부분만 얻기 위해서는 버림 블록을 이용하여 소수점 이하는 없애고 정수만을 얻는다. 소수점 이하 부분의 값은 처음 실수에서 정수를 빼면 얻을 수 있다.



5.5 수학 함수 연산 블록(6/8)

[예제 5.11] $y = \sqrt{x}$ 에 대한 그래프를 그리는 스크립트를 작성하라.

■ 준비 단계

- ① 화면에 스프라이트는 나타나지 않기 때문에 스프라이트는 임의의 것을 사용해도 무방하다.
- ② 배경은 모눈종이 배경을 고른다. 모눈종이 배경은 배경 고르기에서 Xy-grid, Xy-grid-20px, Xy-grid-30px 등 3가지가 제공된다. 그래프의 해상도를 고려해서 그래프의 거리 20을 정수 1에 대응하기 위해 Xy-grid-20px을 이용한다.




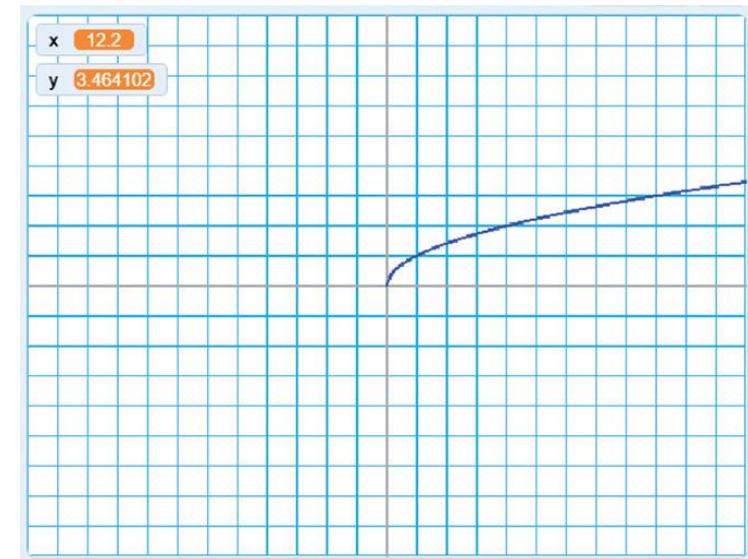
그림 8.12 모눈종이 배경

- x의 값은 0에서 12까지 0.2씩 증가시키면서 다음을 반복한다.
 - 제공된 함수식을 계산하여 y 변수에 저장하고 (20x, 20y) 좌표로 이동시킨다.

5.5 수학 함수 연산 블록(7/8)

[예제5.11] 스크립트

스크립트	설명
	<p>시작하기 버튼을 클릭했을 때 스프라이트를 숨긴다. 이전에 그린 선을 모두 지운다. 그래프를 그리기 위한 시작점(0, 0)으로 이동한다. x 값을 0으로 정한다. 그리기 위해서 펜을 내린다. x 값이 12를 초과할 때까지 다음 절차를 반복한다.(12를 20배하면 240이 되고 벽에 도달하므로 종료한다.) y를 x의 제곱근 값으로 정한다. (20x, 20y)로 이동한다. x를 0.2만큼 바꾼다. 그리기를 멈추기 위해서 펜을 올린다.</p>



5.5 수학 함수 연산 블록(8/8)

그래프 그리기 연습

연습

$y = 1 - e^{-x}$ 의 그래프를 그려라.

힌트

e^{\square} ()

$\square \times \square$

$\square - \square$

블록을 이용한다. 모눈종이 배경을 위해서 Xy-grid-20px
배경을 고른다.

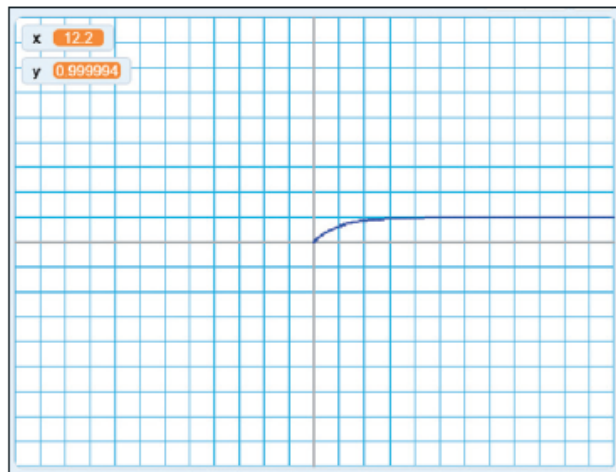


그림 8.13 연습의 실행 화면