

프로그래밍 언어

- 언어 = 대화 규칙
- 프로그래밍 언어 = 컴퓨터와의 대화 (명령)
- 로직 = 논리적인 흐름(작업순서)

컴퓨터와 숫자맞추기 게임

- 작업순서(로직)

1. 컴퓨터가 임의의 수를 생성
2. 사용자가 숫자를 입력
3. 컴퓨터는 사용자가 입력한 수와 미리 정해둔 수를 비교한다
4. 같으면 "정답"이라고 출력하고 끝낸다
5. 작으면 "작다"라고 출력하고 다시 2번 과정 숫자입력으로 돌아가서 반복한다.
6. 크면 "크다"라고 출력하고 다시 2번 과정 숫자 입력으로 돌아가서 반복한다.

컴퓨터와 숫자맞추기 게임

1. 컴퓨터가 임의의 수를 생성
2. 사용자가 숫자를 입력
3. 두 수를 비교하여 같으면 "정답"이라고 출력하고 끝낸다
4. 작으면 "작다"라고 출력하고 다시 2번 과정 반복한다.
5. 크면 "크다"라고 출력하고 다시 2번 과정부터 반복한다.

```
let num = Math.floor(Math.random() * 10) + 1;

while(true) {
  let guess = prompt()

  if(num == guess) {
    alert("정답");
    break;
  }
  if(num < guess) {
    alert("작다");
  }
  if(num > guess) {
    alert("크다");
  }
}
```

컴퓨터와 숫자맞추기 게임

- 요구사항

간단한 숫자 맞추기 게임을 만들어주세요. 무작위로 선택한 1과 100 사이의 수를 플레이어에게 10턴 내에 알아내는 게임입니다. 각 턴 후에는 플레이어가 고른 숫자가 맞았는지 아니면 틀렸는지 알려줘야 하고, 만약 틀린 경우에는 플레이어의 숫자가 너무 낮았는지, 아니면 너무 높았는지를 알려줘야 합니다. 플레이어가 이전에 추측한 숫자들의 기록도 보여줘야 합니다. 게임은 플레이어가 숫자를 맞히거나, 턴을 모두 소모하면 끝납니다. 게임이 끝난 뒤에는 다시 게임을 시작할 수 있는 옵션을 제공해야 합니다.

프로그래밍 언어에서 알아야할 내용

- 변수

```
let num = Math.floor(Math.random() * 10) + 1;
```

- 함수

- 입력함수
- 출력함수

```
while(true) {  
    let guess = prompt();
```

- 연산자

- 반복문

- 제어문

```
    if(num == guess) {  
        alert("정답");  
        break;  
    }  
    if(num < guess) {  
        alert("작다");  
    }  
    if(num > guess) {  
        alert("크다");  
    }  
}
```