



프로그래머와 프로그래밍 언어

프로그래머가 하는 일은?

문제를 해결하는 사람

{

도메인 지식 +

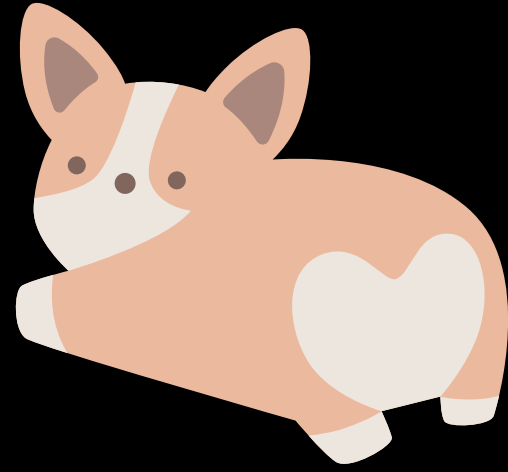
프로그래밍 +

커뮤니케이션

}

강아جي 분양 서비스를 만들어 줘

문제 나누기



기획

디자인

개발

배포

프론트엔드

백엔드

앱 웹

API DB DevOps

아주 작은 문제로 계속 나누어 해결하기

- 강아지 등록
 - DB설계
 - 스키마 정의
 - 백업 전략 선택
 - API구현
 - 강아지 프로필 등록
 - 프로필 저장
 - 사진 처리
 - 저장
 - 업데이트
 - 등록하지 않을 경우
 - 주인 프로필 등록
- ...

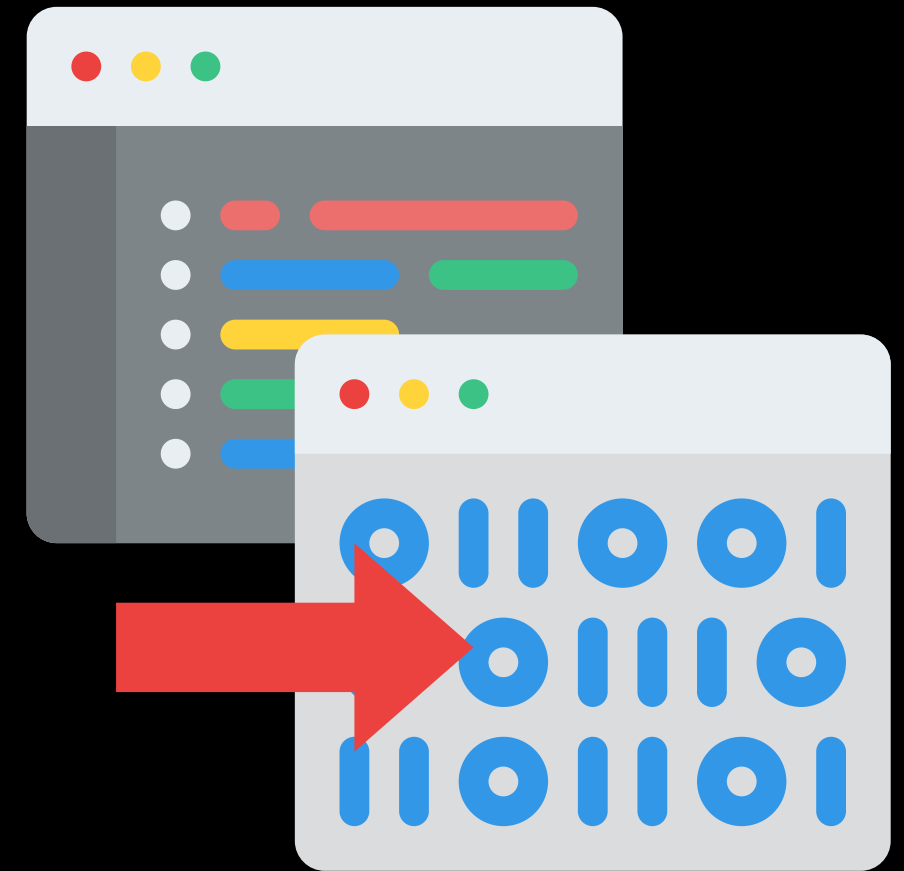
프로그래밍 언어

저수준(Low Level) 언어 VS 고수준(High Level) 언어





Interpreter



Compiler

```
# Python  
myVariable = 10  
myVariable = "Hello"
```

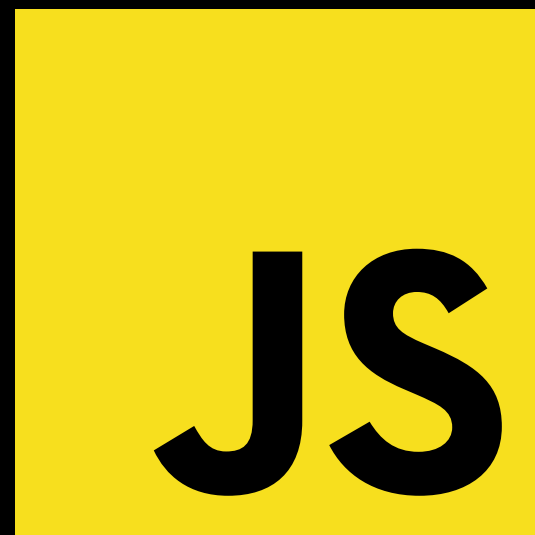
OK

```
// C  
int myVariable = 10;  
  
// Error  
myVariable = "Hello";
```



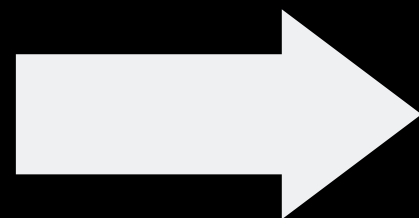
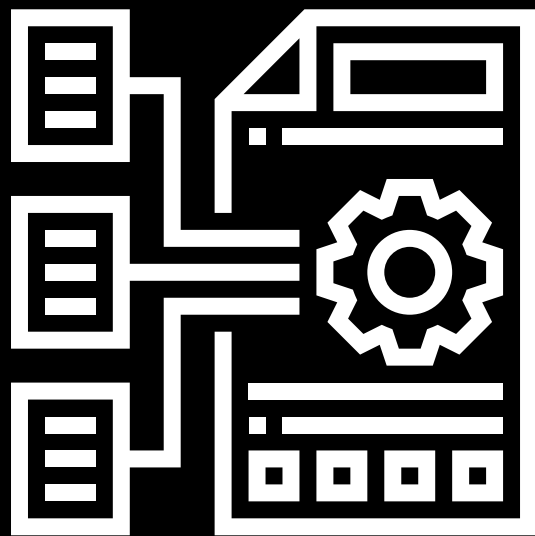


=

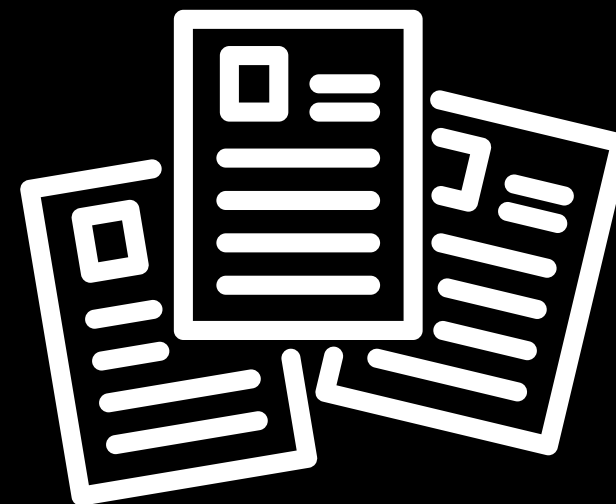


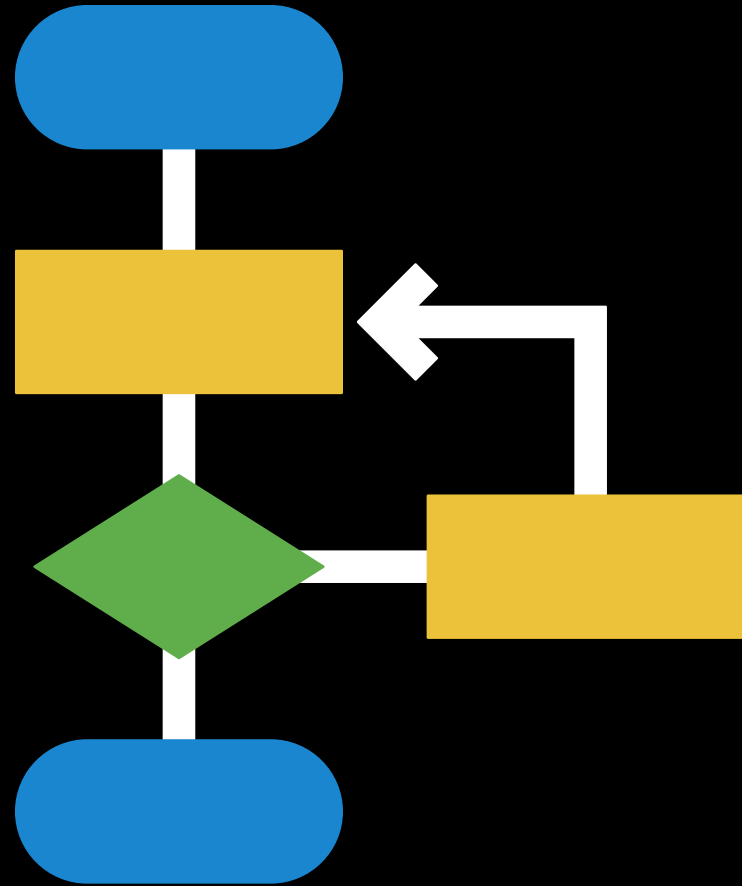
+ **정적 타입 기능**

*.ts tsc

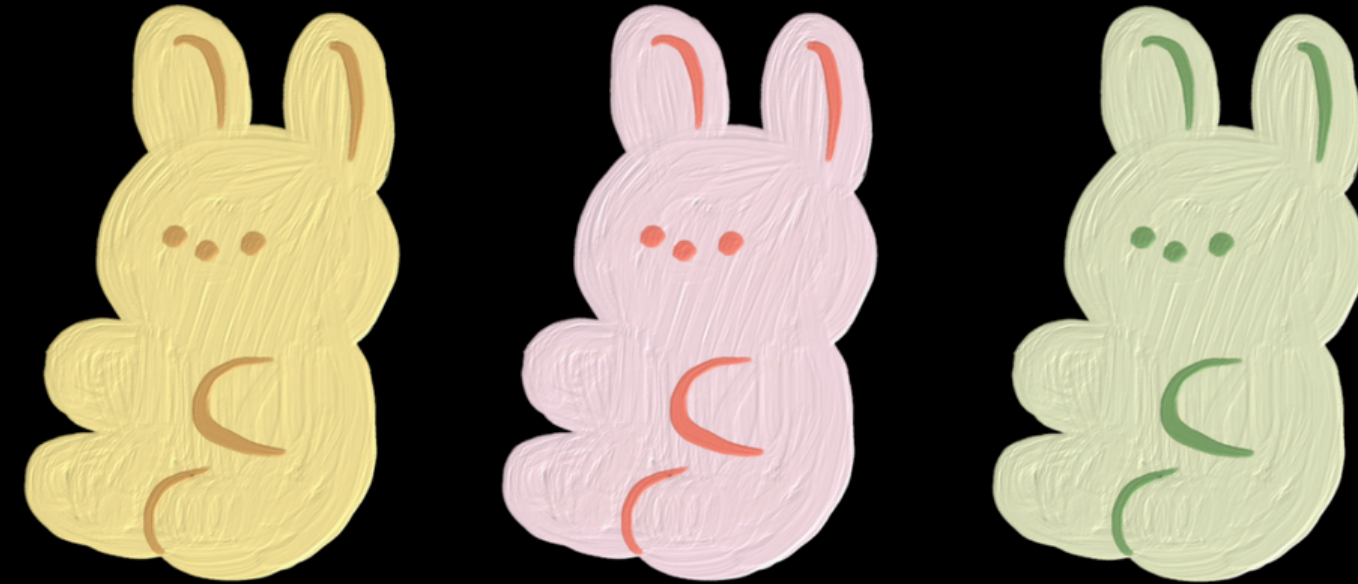


*.js





**Procedure-oriented
Programming**



Object-Oriented Programming

$$f(x) \cdot g(x)$$

**Functional
Programming**



SUM

```
function sum(a, b) {  
  return a + b;  
}  
sum(10, 20);
```



SUM

```
async function sum() {  
  const a = await fetch('/api/a');  
  const b = await fetch('/api/b');  
  return a + b;  
}
```

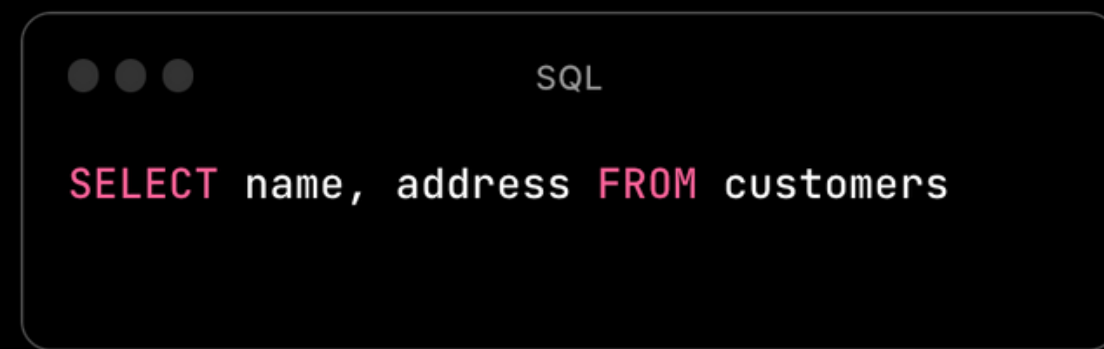


SUM

```
async function sum(): Result<number, Error> {  
  try {  
    const a = await fetch('/api/a');  
    const b = await fetch('/api/b');  
    return Result(a + b, null);  
  } catch (error) {  
    return Result(null, error);  
  }  
}
```

SUM

```
function sum(): Promise<number> {  
  return new Promise<number>((resolve, reject) => {  
    try {  
      const a = await fetch('/api/a');  
      const b = await fetch('/api/b');  
      resolve(a + b);  
    } catch (error) {  
      reject(error);  
    }  
  })  
}
```



A code editor window with a title bar containing three small circles and the text 'SQL'. The editor contains the following SQL query:

```
SELECT name, address FROM customers
```

Domain-Specific Languages



프로그래머와 프로그래밍 언어