

# 제 1강: Lua 기초와 Xilica 통합 (Lua Basics & Xilica Integration)

## 1. 소개

Xilica Designer의 Lua 모듈은 강력하지만, 시작하려면 기본 구조를 이해해야 합니다. 이번 강의에서는 Lua의 변수 타입과 Xilica의 데이터 입출력 방식인 InTable 과 OutTable에 대해 배웁니다.

## 2. Lua의 기본 문법

### 변수 (Variables)

Lua 변수는 타입을 명시하지 않습니다. (Dynamic Typing)

```
local name = "Xilica" -- 문자열 (String)
local volume = -10.5 -- 숫자 (Number, 실수/정수 구분 없음)
local is_on = true -- 불리언 (Boolean)
local list = {} -- 테이블 (Table, 배열/딕셔너리 겸용)
```

**중요:** local 키워드를 사용해야 해당 스크립트 내에서만 유효한 변수가 됩니다. 전역 변수는 최소화하세요.

### 테이블 (Table)

Lua의 핵심 자료구조입니다. 배열처럼 쓸 수도 있고, 키-값 쌍으로 쓸 수도 있습니다.

```
-- 배열처럼 사용 (인덱스는 1부터 시작!)
local inputs = {1, 0, 1}
print(inputs[1]) -- 결과: 1

-- 딕셔너리처럼 사용
local cmd_map = {
    ["POWER"] = "PWR_ON",
    ["VOL"] = "VOL_SET"
}
print(cmd_map["POWER"]) -- 결과: "PWR_ON"
```

## 3. Xilica의 입출력 구조

Xilica Lua 모듈은 외부 세계와 소통하기 위해 두 가지 특별한 테이블을 사용합니다.

### InTable (입력)

- 외부 핀으로 들어오는 값들이 저장됩니다.
- 예: 첫 번째 핀의 값은 InTable[1]에 있습니다.
- 주의:** 값이 없을 수도 있으므로(nil), 항상 안전하게 처리해야 합니다.

```
local vol = InTable[1]
if vol == nil then vol = 0 end
```

```
-- 또는 짧게: local vol = InTable[1] or 0
```

### OutTable (출력)

- 외부 핀으로 나갈 값들을 저장합니다.
- 예: 첫 번째 핀으로 값을 보내려면 OutTable[1] = 값
- Xilica에서는 핀의 타입(Logic, String, Int)에 따라 이름을 맞춰주는 것이 좋습니다.

```
OutTable[1] = "Hello"      -- 일반 인덱스  
OutTable["1_String"] = "Hello" -- 명시적 타입 지정 (권장)
```

## 4. 실습 예제 (Exercise)

attach/course/exercises/ex01\_variables\_template.lua 파일을 열어 다음 미션을 수행하세요.

미션 목표:

- 입력 1번(볼륨)을 받아 0보다 작으면 0으로 만드세요.
- 입력 2번(음소거 여부, 1 or 0)을 받아 true/false 불리언으로 변환하세요.
- 결과를 OutTable에 담아 리턴하세요.