TIZEN™

Tizen.IoT.Connectivity

w/ clouds

Tizen Talks

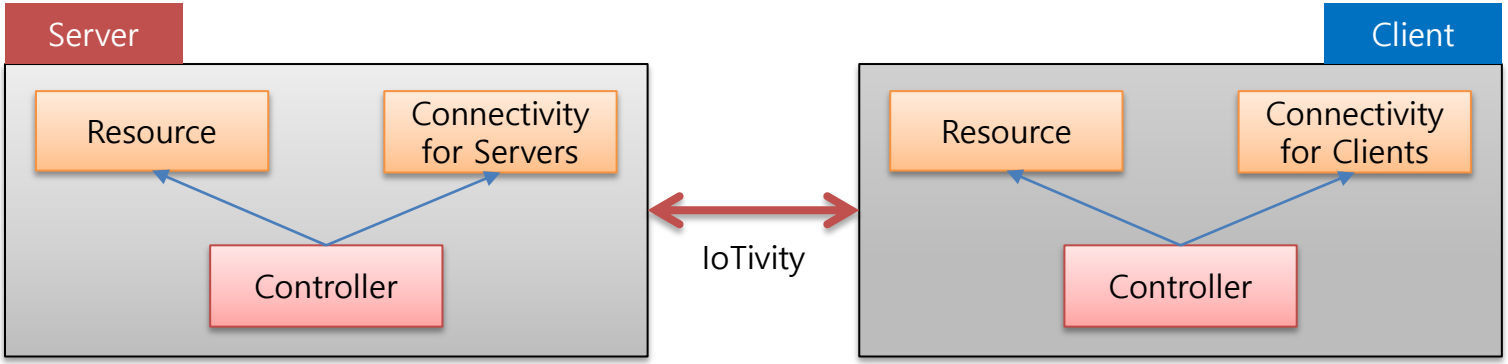
TIZEN

TIZEN™

Big Picture

TIZEN

IoTivity Server & Client

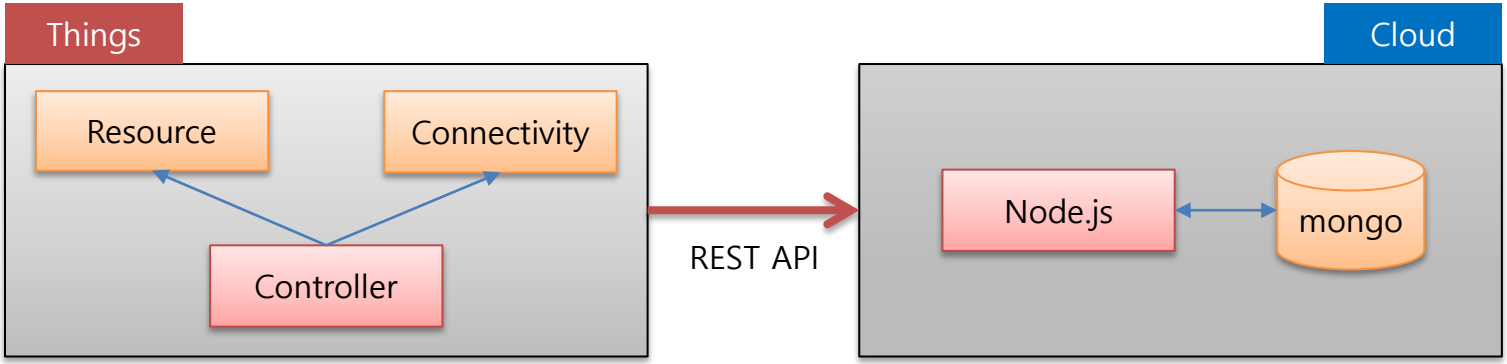


apps/native/position-finder-server

apps/native/position-finder-client

동영상 강좌
https://www.youtube.com/playlist?list=PLl68CcEDTcy7LswNVSmHppxabc_YAYNWZ

Things & Cloud



apps/native/position-finder-server

TIZEN™

Background

TIZEN

Background

REST(REpresentational State Transfer) **API** ℹ️

- (별도의 전송계층을 사용하지 않고) HTTP를 이용하여 서버와 클라이언트간 통신 방법
- 3가지 구성요소 (Resource, Method, Message)

1. Resource

"http://webserver/api" 같은 형태의 URI(Uniform Resource Identifier)

2. Method

HTTP 프로토콜의 4가지 method 사용 : CRUD (Create, Read, Update, Delete)

- POST(Create), GET(Read), PUT(Update), DELETE(Delete)

3. Message

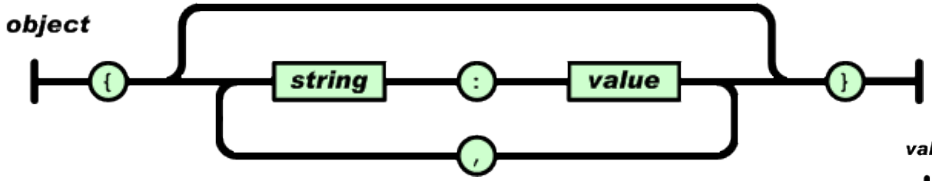
text, file 등 다양한 형태가 가능하나 JSON, XML 등 형식의 메시지를 일반적으로 사용

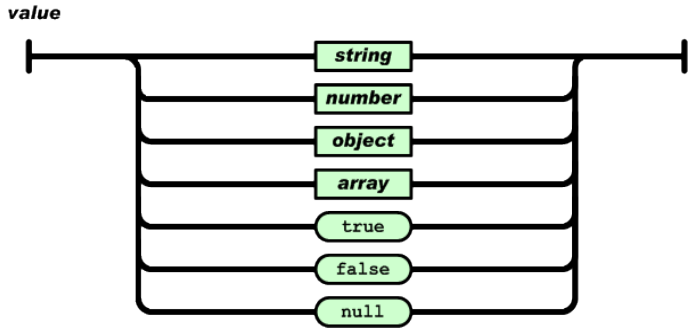
Background

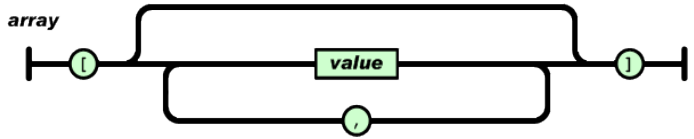
JSON(JavaScript Object Notation)

- JavaScript에서 생겨났지만, Data 교환을 위해 언어 독립적으로 사용 중

- 구조







- 예제

```
{
  이름 : 홍길동,
  학과 : 컴퓨터공학과,
  수강과목 : [인공지능, 알고리즘],
}
```

Background

Libcurl ℹ️

- URL기반의 데이터 전송 라이브러리
- HTTP, HTTPS, FILE, FTP, FTPS.... 등등 지원

- POST method 사용 예제

```
#include <curl/curl.h>

int web_util_noti_post(const char *url, const char *json_data)
{
  int ret = 0;
  CURL *curl = NULL;
  CURLcode response = CURLE_OK;
  struct curl_slist *headers = NULL;

  curl = curl_easy_init();

  headers = curl_slist_append(headers, "Accept: application/json");
  headers = curl_slist_append(headers, "Content-Type: application/json");

  curl_easy_setopt(curl, CURLOPT_URL, url);
  curl_easy_setopt(curl, CURLOPT_POST, 1L);
  curl_easy_setopt(curl, CURLOPT_HTTPHEADER, headers);
  curl_easy_setopt(curl, CURLOPT_POSTFIELDS, json_data);

  response = curl_easy_perform(curl);

  curl_slist_free_all(headers);
  curl_easy_cleanup(curl);

  return ret;
}
```

Background

Libcurl

- GET method 사용 예제

```
#include <curl/curl.h>

int web_util_noti_get(const char *url, char **response)
{
  int ret = 0;
  CURL *curl = NULL;
  CURLcode response = CURLE_OK;

  curl = curl_easy_init();

  curl_easy_setopt(curl, CURLOPT_URL, url);
  curl_easy_setopt(curl, CURLOPT_WRITEFUNCTION,
                   _get_response_write_callback);
  curl_easy_setopt(curl, CURLOPT_WRITEDATA, (void *)response);

  response = curl_easy_perform(curl);

  curl_easy_cleanup(curl);

  return ret;
}
```

```
static size_t _get_response_write_callback(void *ptr, size_t size,
                                           size_t nmemb, void *data)
{
  size_t res_size = 0;
  char **received = (char **)data;

  res_size = size*nmemb;

  if (received && res_size > 0)
    *received = strndup((char *)ptr, size*nmemb);

  return res_size;
}
```

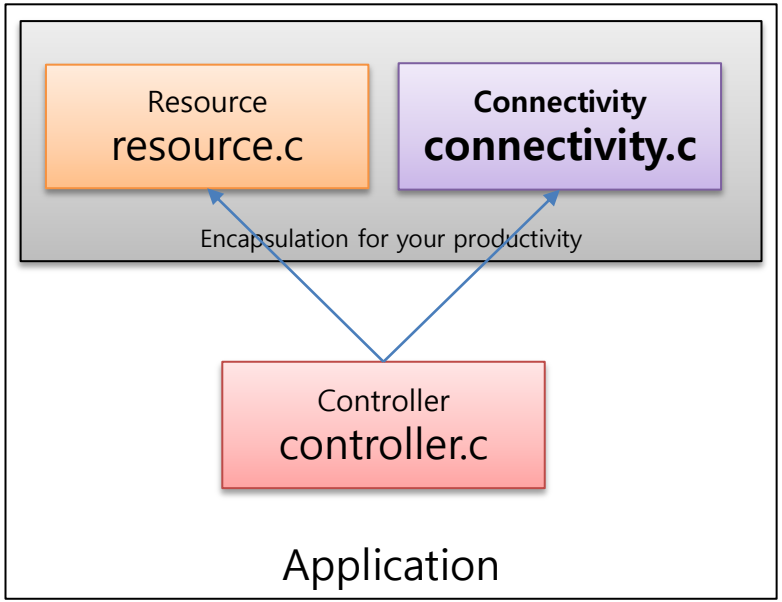
TIZEN™

RCC Pattern Connectivity

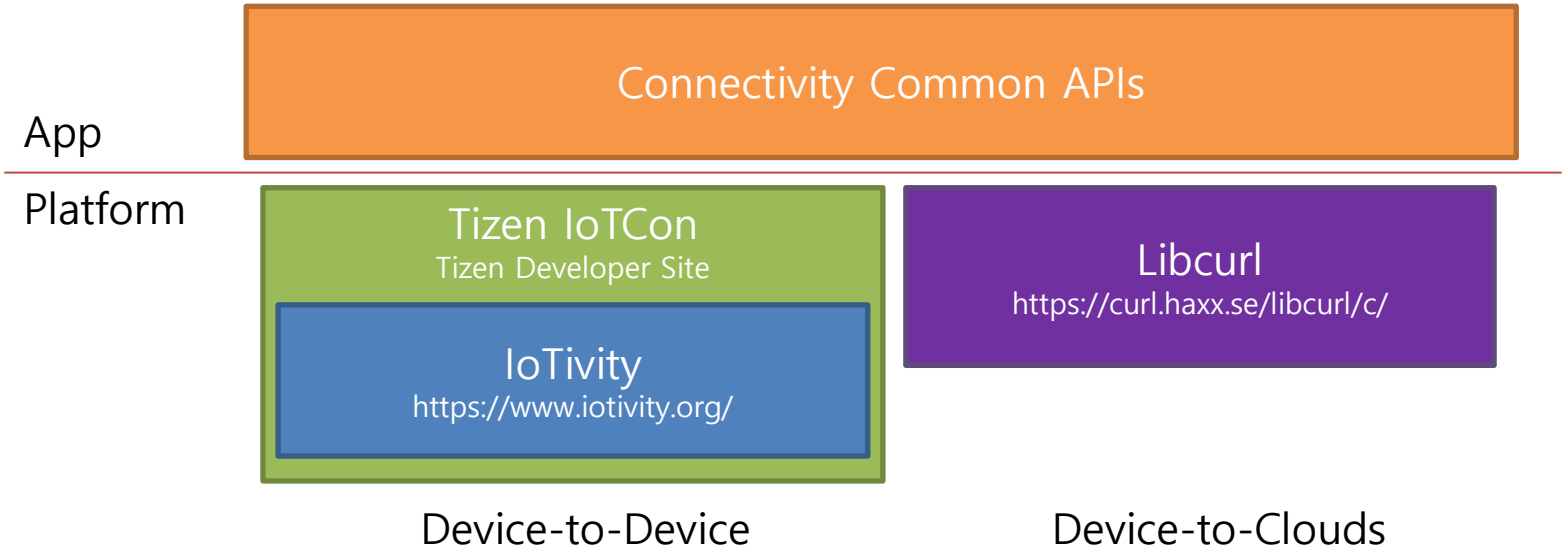
TIZEN

Resource Connectivity Controller

- **Resource**
  - represents real state content
  - collects data
  - Sensors, LED, …
- **Connectivity**
  - connectivity of devices
  - integrated into communication networks
  - local network or cloud network
- **Controller**
  - controls resources and connectivity
  - mainloop



Connectivity



TIZEN™

Practice

TIZEN

Back to Connectivity

Connectivity APIs

- Initialize

int **connectivity_set_protocol**(connectivity_protocol_e protocol_type);

int **connectivity_set_resource**(const char *path, const char *type, connectivity_resource_s **out_resource_info);

controller.c

```
static bool service_app_create(void *data)
{
    app_data *ad = data;
    ......
    ret = connectivity_set_connectivity_type(CONNECTIVITY_TYPE);
    ......
    ret = connectivity_set_resource(PATH, TYPE, &ad->resource_info);
    ......
    return true;
}
```

CONNECTIVITY_PROTOCOL_HTTP

```
typedef enum {
    CONNECTIVITY_PROTOCOL_DEFAULT = 0,
    CONNECTIVITY_PROTOCOL_IOTIVITY,
    CONNECTIVITY_PROTOCOL_HTTP,
    CONNECTIVITY_PROTOCOL_MAX
} connectivity_protocol_e;
```

"/door/??" "org.tizen.door"

※ finalize - void **connectivity_unset_resource**(connectivity_resource_s *resource);

실습 1/3

Back to Connectivity

Connectivity APIs

- Notify sensing data

- **API for simple data**

int **connectivity_notify_bool**(connectivity_resource_s *resource_info, const char *key, bool value);

int **connectivity_notify_int**(connectivity_resource_s *resource_info, const char *key, int value);

int **connectivity_notify_double**(connectivity_resource_s *resource_info, const char *key, double value);

int **connectivity_notify_string**(connectivity_resource_s *resource_info, const char *key, const char *value);

- **API for complex data**

int **connectivity_attributes_add_bool**(connectivity_resource_s *resource_info, const char *key, bool value);

int **connectivity_attributes_add_int**(connectivity_resource_s *resource_info, const char *key, int value);

int **connectivity_attributes_add_double**(connectivity_resource_s *resource_info, const char *key, double value);

int **connectivity_attributes_add_string**(connectivity_resource_s *resource_info, const char *key, const char *value);

int **connectivity_attributes_notify_all**(connectivity_resource_s *resource_info);

int **connectivity_attributes_remove_value_by_key**(connectivity_resource_s *resource_info, const char *key);

int **connectivity_attributes_remove_all**(connectivity_resource_s *resource_info);

Back to Connectivity

Connectivity APIs

- Notify sensing data

controller.c

```
static Eina_Bool _control_sensors_cb(void *data)
{
    app_data *ad = data;
    int value = -1;
    ......
    ret = resource_read_infrared_motion_sensor(21, &value);
    ......
    ret = connectivity_notify_int(ad->resource_info, KEY, value);
    ......
    return ECORE_CALLBACK_RENEW;
}
```

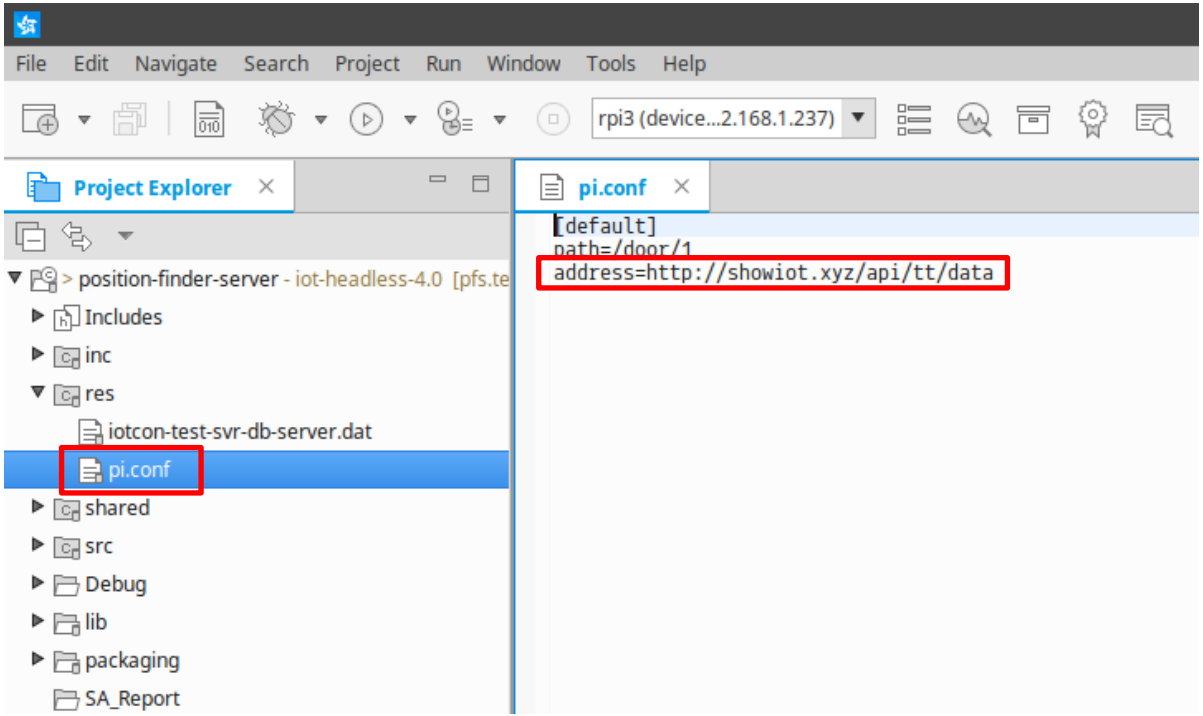
"Motion"

실습
2/3

Back to Connectivity

URI for REST API

- pi.conf 파일의 address 항목에 정의

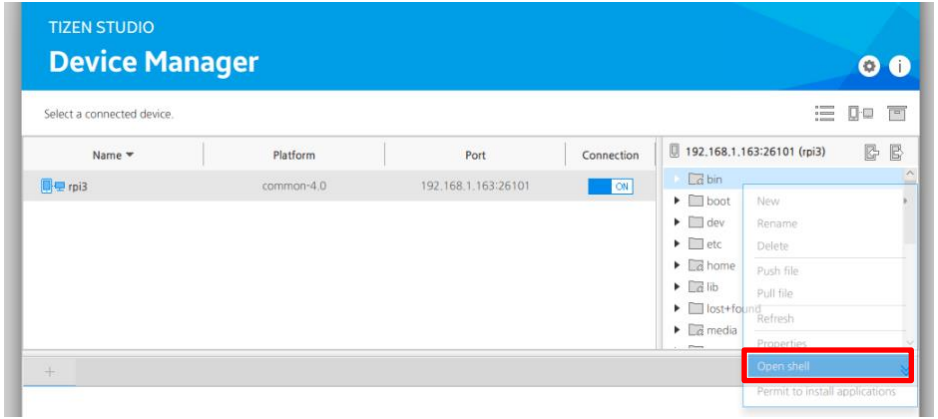


Runs your code on RPi3

1. Running the project on RPi3.

   Run > Run As > ⋯

2. How to open the shell.



** NOTE:* The password is not shown when you enter it.

3. How to view logs.

   Execute *dlogutil* in the opened shell.

   $ dlogutil <LOG_TAG> ex) dlogutil TT

```
[connectivity_notify_int:655] Notify key[Motion], value[10]
[web_util_noti_post:98] server : http://showiot.xyz/api/tt/data
[web_util_noti_post:99] json_data : {"SensorPiID":"/door/1","SensorPiType":"org.tizen.door","Motion":10}
[_post_response_write_callback:48] POST response : "values: {\"SensorPiID\":\"/door/1\",\"SensorPiType\":\"org.tizen.door\",\"Motion\":10}, time: 2017-11-07T06:30:38
```

TIZEN™

Thank you

Tizen Talks

TIZEN