会津大学 講習会

簡単な IoT を作成

 \mathbbm{C} 2018 The University of Aizu

目次

1	課題.		. 1
	1.1 課題	題説明	. 1
	1. 1. 1	課題 5-1	.1
	1. 1. 2	課題 5−2	. 1
2	Goog	le API	. 2
	2.1 Gn	nail API とは	. 2
	2.2 Gn	nail API 設定	. 2
	2. 2. 1	Google アカウント作成	. 2
	2. 2. 2	Google Developers Console内で設定	. 8
	2. 2. 3	Raspberry Pi 上で GmailAPI を使える様にインストール	19
	2. 2. 4	正常に動作するか確認するために Quickstart を実行	21
	2.3 Py	thon でメールを出す	26
3	温度	を取得してメールで送信するプログラムの作成	30
	3.1 温度		30
	3.2 温度	度を取得して送信するプログラムを作成	30
	3.3 プロ	ログラムの実行	30
4	温度	を取得してメールを1分毎に送信するプログラム	31
	4.1 シュ	ェルスクリプトとは	31
	4.2 ショ	ェルの書き方	31
	4.2.1	作り方・実行の仕方	31
	4.2.2	変数	31
	4. 2. 3	条件分	32
	4. 2. 4	ループ文	33
	4. 2. 5	コマンドの実行	34
	4.3 ショ	ェルスクリプト作成	35

1 課題

1.1 課題説明

1.1.1 課題 5-1

Raspberry Pi に接続した温度センサから温度を取得して Gmail で送付する



Gmail API を使用して Python プログラム上から Gmail を送信する簡単な IoT システムを作成する。

1.1.2 課題 5-2

課題1で温度をメールで送付するプログラムを作成しました。次はこのプログラムを1分毎に 起動するように設定しましょう。

動かす方法はシェルスクリプトを使用します。

2 GOOGLE API

Google API とは提供するプラットフォームやサービスを使用出来る API

2.1 Gmail APIとは

メッセージの送信や受信など Gmailの様々な機能をプログラムから操作をするための API で Googel APIの一種です。

2.2 Gmail API 設定

Gmail API を使用するためには以下の設定が必要です。

- Google アカウントの作成 Gmail 使用するために Google のアカウントを作成します。
- 2. Google Developers Console内で設定
 - 1) プロジェクトの作成
 - 2) Gmail API を有効化
 - 3) 認証情報を作成
 - 4) client_id. json をダウンロード
- 3. 使用するコンピュータ内で設定
 - 1) インストール

2.2.1 Google アカウント作成

最初に使用する Google のアカウントを作成します。

1. 「ログイン」ボタンを押下。

G Google × +		- 🗆 X
\leftrightarrow \rightarrow C $$ https://www.google.co.jp		☆ ⊖ :
		Gmail 画像 👥 ログイン
	Google	
	Ļ	
	Google 検索 I'm Feeling Lucky	
日本		
広告 ビジネス Googleについて		プライバシー 規約 設定

2. 「アカウント作成」リンクを押下。

G ログイン - Google アカウント × +		-		×
← → C A https://accounts.google.com/S	erviceLogin/identifier?hl=ja&passive=true&continue=https%3A%2F%2Fwww.google.co.jp%2F&flowNa	☆	9	:
	Google ログイン お客様の Google アカウントを使用 メールアドレスまたは電話番号 メールアドレスを忘れた場合 ご自分のパソコンでない場合は、ゲストモードを使用 して非公開でログインしてください。ヘルプ			

3. 全項目(姓,名,ユーザ名,パスワード)を入力。

G Google アカウントの作成	× +		-		×
\leftrightarrow \rightarrow C \triangleq http	s://accounts.google.com/signup/v2/webcreateaccount?continue=http	s%3A%2F%2Fwww.google.co.jp%2F&hl=ja&flowN	lame=Glif ☆	Θ	:
	Google アカウントの作成 # 名 ユーザー名 @gmail.com 半角英字、数字、ピリオドを使用できます。 代わりに現在のメールアドレスを使用 パスワード 確認				*
	半角英字、数字、記号を組み合わせて 8 文字以上で入力し てください	1つのアカウントで Google のす べてのサービスをご利用いただけ ます。			1
	代わりにログイン 次へ 次へ				•

4. 入力内容を確認し誤りがなければ、「次へ」ボタンを押下。

G Google アカウントの作成	ε × +		-			×
\leftrightarrow \rightarrow C $https://dots.com$	ps://accounts.google.com/signup/v2/webcreateaccount?continue=http	os%3A%2F%2Fwww.google.co.jp%2F&hl=ja&flowNam	07	☆	θ	:
	<mark>Google</mark> Google アカウントの作成					•
	姓 名 (会津) 太郎					1
	ユーザー名 @gmail.com 半角英字 数字 ドリオドを使用できます					ł
	代わりに現在のメールアドレスを使用					
	* 単角英学、数字、記号を組み合わせて 8 文字以上で入力し てください	1 つのアカウントで Google のす べてのサービスをご利用いただけ ます。				
	代わりにログイン 次へ					ł
						-

5. SMS で確認コードが届くため, SMS 受信可能な電話番号を入力。 ※端末やブラウザによっては,下記の画面が表示されない。その場合項番8へ進む。

G Google アカウントの作成	× +	-	I		×
\leftrightarrow \rightarrow C $$ http://	ps://accounts.google.com/signup/v2/webgradsidvphone?continue=https%3A%2F%2Fwww.google.co.jp%2F&hl=ja&flowNam	07	☆	Θ	:
	Google 第二日前日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日日				

6. 電話番号に誤りがなければ、「次へ」ボタンを押下。

G Google アカウントの作成	× +	-			×
\leftrightarrow \rightarrow C a http	bs://accounts.google.com/signup/v2/webgradsidvphone?continue=https%3A%2F%2Fwww.google.co.jp%2F&hl=ja&flowNam	07	☆	θ	:
	Googe 加二のののののののののののののののののののののののののののののののののののの				

7. SMS で届いた"確認コード"を入力し、「確認」ボタンを押下。

G Google アカウントの作成 × +	-			×
← → C 🌲 https://accounts.google.com/signup/v2/webgradsidvverify?continue=https%3A%2F%2Fwww.google.co.jp%2F&hl=ja&flowNam	07	☆	Θ	:
Google 第二部の子の確認 サント・マント・ジント Google では本人である たるを確認する必要があります。6 桁の確認コード が記載されているテキスト メッセージが Google から送信されます。通常のデータ通信料がかります。 「ロー・ロー・ 「」」」」 「ロー・ロー・ 「」」」」 「ロー・ロー・ 「」」」」 「ロー・ロー・ 「」」」」 (日本) (日本)				

8. 必須項目(生年月日,性別)を入力。

※電話番号とメールアドレスは省略可。

Coogle / //// Off Ma	× +	_	· .	_	×
 ← → C a https; 	 × ▲ // ccounts.google.com/signup/v2/webpersonaldetails?continue=https%3A%2F%2Fwww.google.co.jp%2F&hl=ja&flowNa Coogle Coogle Coogle へようここ ③ @ @gmail.com ● @ @gmail.com<td>67</td><td>*</td><td>Θ</td><td></td>	67	*	Θ	
	アカウントを保護する目的で使用されます 年 月 生年月日 性別 ● ごの情報が必要な理由				

9. 電話番号とメールアドレスはセキュリティ保護のために基本的には入力するが, 今回は未入力で登録。

G Google アカウントの作成 × +	-			\times
$\leftarrow \rightarrow \ \ C \ \ \ \ \ \ \ \ \ \ \ \ \$	07	☆	Θ	:
Google				^
Google へようこそ				
<u>®</u>				
@gmail.com				
電話番号 (省略可)				
Google では、アカウントのセキュリティ 検護に電話番号を使用します。電話番号が他のユーザーに公開されることはありせん。				
再設定用のメールアドレス(省略可)				
アカウントを保護する目的で使用されます				
年月 - 日				
生年月日				
個人情報は非公開であり、安全で 性別 す				
この情報が必要な理由				
				Ţ

10. 生年月日, 性別に誤りがなければ, 「次へ」ボタンを押下。

G Google アカウントの作成	× +	-]	×
\leftrightarrow \rightarrow C \triangleq http:	s://accounts.google.com/signup/v2/webpersonaldetails?continue=https%3A%2F%2Fwww.google.co.jp%2F&hl=ja&flowNa	07	☆	θ	:
	Google				^
	Google へようこそ				
	<u> </u>				
	@gmail.com				
	Google では、アカウントのセキュリティ保護に電話番号を使用します。電話番号が他のユーザーに公開されることはありせん。				
	再設定用のメールアドレス(省略可)				
	アカウントを保護する目的で使用されます				
	生年月日 個人情報は非公開であり 安全で				
	7. 小学業和15人と1回+51回+5				
	この有報が必要な理由				
	戻る 次へ				

11. 利用規約を確認し問題なければ、「同意する」ボタンを押下。



12. 作成したユーザでログインされていることを確認。

G Google × +		-		×
← → C			☆	***
	Gmail 画像		0	太郎
	Google			
	Ų			
	Google 検索 I'm Feeling Lucky			
日本				
広告 ビジネス Googleについて	プライバ	≥— ;	現約	設定

これで Google のアカウントが作成できました。

2.2.2 Google Developers Console内で設定

Gmail API の設定を有効にします。

 URL 入力バーに"<u>https://console.developers.google.com/apis/dashboard</u>"を入力し、 「Enter」キーを押下。

※1 上記 URL (Google APIs) にアクセスする場合, Internet Explorer では正常に表示されないことがあるため, Google Chrome を使用することを推奨。

※2 Google Chrome をインストールされていない方は,下記の URL よりダウンロードし インストールしてください。

https://www.google.com/chrome/

G Google	× +					-		×
$\leftarrow \ \rightarrow \ G$	5 https://console.developers.g	google.com/apis/dashboard) 0	:
					Gmail 画像		コグイン	
		Goo	ogle					
				Ļ				
		Google 検索	I'm Feeling Lucky					
日本								
広告 ビジ	ネス Googleについて				プライバミ	— 規約	句 設定	Ē

2. <u>3.2.1 Google アカウント作成</u>で決めたパスワードを入力。

※既にログイン済の場合は、項番4に進む。

G Google Cloud ブラットフォーム × +					-	-		×
\leftrightarrow \rightarrow C $$ https://accounts.google.c	m/signin/v2/sl/pwd?service=cloudcons	sole&passive=	1209600&osid	=1&continue=https%	3A%2F%2Fconsol	☆	Θ	÷
	Goo 会津; () パスワードを入力 パスワードをお忘れの場合	gle 太郎 @gmail.com	× &					*
		1111	ノブイハンニ	为元不当				*

3. パスワードに誤りがなければ、「次へ」ボタン押下。

→ C 🌲 https://accounts.google.co	m/signin/v2/sl/pwd?service=cloudconsole&passive=1209600&osid=1&cor	ntinue=https%3A%2F%2Fc	6	. 0) :
	Google 会津太郎 @gmail.com ~ パスワードを入力 パスワードをお忘れの場合				

4. 利用規約を確認し問題なければ、"同意します"にチェックし、"最新情報をメールで通知"を 希望する場合は"はい"、希望しない場合は"いいえ"を選択。



5. 内容に誤りがなければ、「承諾する」ボタンを押下。



6. Google APIs の画面が表示されたら、「プロジェクトを作成」ボタンを押下。



7. 「作成」ボタンを押下。



8. プロジェクト名を入力し,「作成」ボタンを押下。

※プロジェクト名は作成した後で"プロジェクトの設定"より変更可。

G 新しいプロジェクトーGoogle API Co × +		-]	×
← → C		0-	☆	太郎	:
無料トライアルを試してみませんか。有効にしていただくと、\$300 分のクレジットを獲得して Google Cloud サービスをお試しいただけます。詳細		閉じる	5	有効(ſĿ
E Google APIs Q	0	۰	:		
新しいプロジェクト					
▲ 割り当て内の残りの projects 数は 12 件です。割り当てを増やすようリク エストするか、プロジェクトを削除してください。 詳細 MANAGE QUOTAS プロジェクト&*- My Project 57373 ②					

9. Google APIs のロゴの右にあるプロジェクト選択リストボックスで作成したプロジェクト が選択されていることを確認し, Gmail API を選択。



10. 「有効にする」ボタンを押下。

G	API とサービス - My Project 57373 - 🗙	+		-]	×
÷	\rightarrow C https://console.d	velopers.google.com/apis/library/gmail.googleapis.com?id=869e4b9c-0da4-4cbe-8b8d-c77f7ae060cc&proj	9C	07	☆	太郎	:
=	Google APIs 🔹 My Proj	ect 57373 🗸 🍳 🖬 🖗	0	1	:		B
÷	API ライブラリ						
	M	Gmail API Google Flexible, RESTful access to the user's inbox 有効にする この API を試す ご					
	タイプ API とサービス 最終更新 2018/12/27 8:10 カテゴリ メールアドレス G Suite	概要 The Gmail API lets you view and manage Gmail mailbox data like threads, messages, and labels. Google の概要 Google's mission is to organize the world's information and make it universally accessible and useful. Through products and platforms like Search, Maps, Gmail, Android, Google Play, Chrome and YouTube, Google plays a meaningful role in the daily lives of billions of people.					

11. 「認証情報を作成」ボタンを押下。

G AP	리 とサービス — My Project 57373 - 🗙	+				-]	×
\leftarrow	C https://console.dev	velopers.google.com/apis/api/gmail.googleapis.	com/overview?project=second-casing-226800&folde	r&organization	d	0-7	☆	太郎	:
≡	Google APIs : My Proje	ct 57373 👻 Q		ii (0	0	:		
Μ	API とサービス Gmail API	概要 ■ API を無効にする フィ	ィードバックを送信						
₩ ₩	概要 指標 認証情報		品が必要になる可能性があります。開始するには、[認証()	青報を作成] をクリ	リックしてく	ください	<i>۱</i> °		•
		III III III III III III III IIII IIII	M レスポンス コード別のトラフィッ 要求数/秒 (2時間の平均)	ク			1	D/s	l
		提供者: Google サービス名 gmail.googleapis.com 概要 The Gmail API lets you view and manage Gmail malibox data like threads, messages, and labels. 有効化のステータス 有効	▲ 選択した時間枠で使用で 	できるデータがあ Dec 16	りません。 Dec 23		0.1 0.1 0.1 0.1	8/s 5/s 4/s 2/s	
۲I									•

12. 「使用する API」をコンボボックスより"Gmail API"を選択。

G 認証情報ウィザード - My Project 57 ×	+			-	□ ×			
← → C								
	ct 57373 👻 Q	ñ	Ø	0) : 😩			
API API とサービス	認証情報							
ダッシュボード	プロジェクトへの認証情報の追加							
豊 ライブラリ	1 必要な認証情報の種類を調べる							
O→ 認証情報	正しい認証情報の設定をお手伝いします この手順をスキップして、APIキー、クライアントID、サービスアカウントを作成することもできます 使用するAPI APIによって使用する認証ブラットフォームが異なります。また、一部の認証情 報を制限して、特定のAPIのみを呼び出すことができます。 選択.							
۲I								

13. 「API を呼び出す場所」をコンボボックスより"その他の UI (Windows, CLI ツールなど) "を選択し,「アクセスするデータの種類」は"ユーザーデータ"を選択。

G 認証情報ウィザード - My Project 57 ×	+				-		×
← → C	velopers.google.com/apis/credentials/wizard?project=second-casing-226800				0- 7	*	:
	ct 57373 👻 Q	Ĩ	ø	0	1	: (
API APIとサービス	記言正代青年段						
 ダッシュボード サイブラリ 	プロジェクトへの認証情報の追加						*
O→ 認証情報							
۲I	82/安/A-664世间甲牧						-

14. 選択内容に誤りがなければ、「必要な認証情報」ボタンを押下。

G 認証情報ウィザード - My Project 57 ×	+				-		×
\leftarrow \rightarrow C \triangleq https://console.d	evelopers.google.com/apis/credentials/wizard?project=second-casing-226800				o- ☆	大郎	:
	ect 57373 🗸 Q	ñ	ø	0	1	:	
API APIとサービス	認証情報						
ダッシュボード	プロジェクトへの認証情報の追加						•
豊 ライブラリ	1 必要な認証情報の種類を調べる						
⊶ 認証情報	正しい認証情報の設定をお手伝いします この手順をスキップして、APIキー、クライアント ID、サービス アカウントを作成することもできます						
	使用する API API によって使用する認証プラットフォームが異なります。また、一部の認証情 報を制限して、特定の API のみを呼び出すことができます。 Gmail API ◆ API を呼び出しれてあるコンテキストの詳細情報を使用して制限する ことができます。認証情報によっては、特定のコンテキストで安全に使用できま せん。 その他の UI (Windows、CLIツールなど) ◆ アクセスするデータの確想 リクエストするデータの存列 いの選びです。 ● ユーザーデータ Google ユーザーに属するデータにアクセスします(ユーザー許可あり) ● プリケーションデータ 自分のアプリケーションに属するデータにアクセスします						
<1	COSC/GROUDLIN W						•

15. OAuth 2.0 クライアント ID を入力。

G 認証情報ウィザード - My Project 57 ×	+		- 🗆 X
← → C 🔒 https://console.de	velopers.google.com/apis/credentials/wizard?project=second-casing-226800		॰ 🌣 🚥 :
	kt 57373 ✔ Q	ii () 0 🚺 : 😩
API APIとサービス	認証情報		
ダッシュボードガイブラリ	プロジェクトへの認証情報の追加		
o- 認証情報	 ② 必要な認証情報の/種類を認べる Gmail APIを U/ベースのブラットフォームから呼び出します 2 OAuth 2.0 クライアント ID を作成する 名前 ◎ その他のクライアント I OAuth 2.0 内息画面を設定する 4 認証情報をダウンロードする キャンセル 		
<1			

クライアントの ID の名前を今回は「GmailTest」にしてください。

16. OAuth クライアント ID に誤りがなければ、「OAuth クライアント ID を作成」ボタンを 押下。

G 認証情報ウィザード - My Project 57 ×	+			- 🗆 X
← → C 🔒 https://console.de	velopers.google.com/apis/credentials/wizard?project=second-casing-226800			or ☆ 🚳 i
≡ Google APIs 🔹 My Proje	ct 57373 👻 🔍	Ĩ	9 0	0 : 😩
API APIとサービス	認証情報			
🔹 ダッシュボード	プロジェクトへの認証情報の追加			
₩ ライブラリ	✓ 必要な認証情報の種類を調べる			
O→ 認証情報	Gmail API を UI ペースのブラットフォームから呼び出します 2 OAuth 2.0 クライアント ID を作成する 名前 ◎ <u>GmailTest</u> 3 OAuth 2.0 同意画面を設定する 4 認証情報をダウンロードする キャンセル			
<۱				

17. メールアドレスはデフォルトとし、サービス名に任意の名称を入力。

G 認証情報ウィザード - My Project 57 ×	+			-		\times
\leftarrow \rightarrow C \triangleq https://console.de	velopers.google.com/apis/credentials/wizard?project=second-casing-226800			0- ☆	*) :
⊟ Google APIs \$• My Proje	ct 57373 🗸 Q.	ø	Ø	0	:	
API APIとサービス	認証情報					
 ◆ ダッシュポード 曲 ライブラリ ● 認証情報 	プロジェクトへの認証情報の追加 ② 必要な認証情報の種類を調べる Gmail API を UIベースのブラットフォームから呼び出します ③ OAuth 2:0 クライアント ID を作成する OAuth 2:0 の見意画面を設定する ③ OAuth 2:0 同意画面を設定する					4
<۱	に登録されているすべてのアプリケーション で表示されます。 OAuth が磯能するためには、メールアドレス とサービス名を指定する必要があります。					

サービス名の名前を今回は「GmailTest」にしてください。

18. 入力内容に誤りがなければ、「次へ」ボタンを押下。

G 認証情報ウィザード - My Project 57 ×	+			-			×
\leftrightarrow \rightarrow C $($ https://console.de	velopers.google.com/apis/credentials/wizard?project=second-casing-226800			0- 7	2	太郎	:
Google APIs 🔹 My Proje	ect 57373 ✔ Q #	Ø	0	1	:		
API APIとサービス	記名語正作書報						
 ・ダッシュボード ・ライブラリ ・ ・ ・	プロジェクトへの認証情報の追加 ② 必要な認証情報の種類を調べる Gmail API を UIベースのブラットフォームから呼び出します ③ OAuth 20 クライアント ID を作成する OAuth 20 同意画面を設定する メールアドレス ◎ ③ ogmail.com ユーザーに表示するサービス名 ◎ GmailTest ※ その他のカスタマイズオブション						*
<1	で表示されます。 OAuth が隔絶するためには、メールアドレス とサービス名を指定する必要があります。						

19. 「ダウンロード」ボタンを押下。

G 認証情報ウイザード - My Project 57 ×	+			- 0	×
\leftarrow \rightarrow C \triangleq https://console.dev	elopers.google.com/apis/credentials/wizard?project=second-casing-226800			or 🕁	
	t 57373 👻 Q	ñ	Ø 6) 🚺 :	
API APIとサービス	記記時報				
∲ ダッシュボード	プロジェクトへの認証情報の追加				
豊 ライブラリ	必要な認証情報の種類を調べる Cmail APLを IIIベースのブラットフォーム から呼び出します				
⊶ 認証情報					
	OAuth 2.0 クライアント ID を作成する OAuth クライアント「その他のクライアント 1」を作成しました				
	⊘ OAuth 2.0 同意画面を設定する				
	4 認証情報をダウンロードする				
	Client ID				
	この認証情報を JSON 形式でダウンロードします。認証情報ページでいつでも行えます。 <mark>ダウンロード</mark> 後で				
4	完了キャンセル				

20. "client_id.json	"ファイルが正常にダウンロードされたら,	「完了」	ボ	タン	ノを	押下。
G 認証情報ウィザード - My Project 57 ×	+				-	
\leftarrow \rightarrow C $$ https://console.de			o- ☆	* :		
≡ Google APIs 🔹 My Proje	ct 57373 ✔ Q	Ũ	ø	0	1	:
API APIとサービス	記念正情報					
💠 ダッシュボード	プロジェクトへの認証情報の追加					•
豊 ライブラリ	必要な認証情報の種類を調べる Gmail APIをUKペースのブラットフォームから呼びせいます					- 1
⊶ 認証情報	OAuth 2.0 クライアント ID を作成する OAuth クライアント 「その他のクライアント 1」を作成しました					
	✓ OAuth 2.0 同意画面を設定する					- 1
	4 認証情報をダウンロードする					
	Client ID					
	この認証情報を JSON 形式でダウンロードします。認証情報ページでいつでも行えます。					- 1
	ダウンロード 後で					
<1	第7 キャンセル					
Client_id.json ^					<u>ৰ</u> ম	て表示 ×

21. 項番 20 でダウンロードし忘れた場合, 「完了」ボタン押下後に表示される下記の画面の 「≛」ボタンよりダウンロード可。

G 認証情報 - My Project 57373 - G ⊂ ×	+						-			
← → C ♠ https://console.dev	velopers.google.com/ap	is/credentials?proje	ct=second-casir	ng-226800			0-7 ☆	r 🗶 :		
	et 57373 👻	۹			ĩ	Ø	0	:		
API API とサービス	認証情報									
💠 ダッシュボード	認証情報 OAuth 同	意画面 ドメインの	確認							
豊 ライブラリ	認証情報を作成 ▼	证情報を作成 > 削除								
⊶ 認証情報	Create credentials to a	ccess your enabled AP	'ls. For more infor	mation, see the authentication documentation.						
	0Auth 2.0 クライア	ントロ								
	2 名前	作成日 ~	タイプ	クライアントID						
	GmailTest	2018/12/27	その他				6			
								_		
<1										
Client_id.json ^							すべ	べて表示 ×		

22. ファイル名は"client_secret_….json"という名称でダウンロードされる。

G 認証情報 - My Project 57373 - G ⊂ ×	+						-			
\leftarrow \rightarrow C \triangleq https://console.dev	velopers.google.com/ap	is/credentials?proje	ct=second-casin	g-226800			o- 🌣	🚥 :		
Google APIs • My Project	ct 57373 🔻	۹			Ĩ	9 0	0	:		
API APIとサービス	認証情報									
ダッシュボード # ライブラリ	認証情報 OAuth 同	意画面 ドメインの	確認							
o- 認証情報	Create credentials to a	情報を作成 🚽 削除 te credentials to access your enabled APIs. For more information, see the authentication documentation.								
	OAuth 2.0 クライア	ントロ								
	名前 GmailTest	作成日 ~ 2018/12/27	タイプ その他	クライアントロ			6	/ 1 ±		
<1										
Client_secret_1053json							すべて	表示 ×		

ダウンロードされた.json ファイルは[client_secret.json]に名前を変えてください。

2.2.3 Raspberry Pi上で GmailAPI を使える様にインストール

1. 「client_secret.json」ファイルを Raspberry Pi にコピー。



Terterm 画面右上のファイル(F)->[SSH SCP]を使いファイルをコピーします。

TTSSH: S	ecure File Copy			:	×			
<u>F</u> rom: To:	client secret	ison 格納フォルダ		<u>S</u> end <u>C</u> ancel				
	You can drag th	ne file to this window.						
Fr <u>o</u> m:				<u>R</u> eceive				
To:	C:#Program Files	s (x86)¥teraterm						
I rasp	A2.local - pi@raspA	\2: ~ VT						×
ファイル(F)	編集(E) 設定(S)	コントロール(O) ウィンドウ(W)	ヘルプ(I	H)				
Piercop	<u>12: ° \$ lo</u>							^
client_ Desktop Documen	secret.json	Downloads Music omninames-raspA2.bak	omni Pict Publ	names-ras :ures ic	pA2.log	python⊥ Templat Videos	games es	
client_ Desktop Documen pi@rasp	secret.json nts A2:~ \$	Downloads Music omninames-raspA2.bak	omni Pict Rubl	names-ras ures ic	pA2.log	python_ Templat Videos	games es	
client_ Desktop Documen pi@rasp	secret.json nts A2:~ \$	Downloads Music omninames-raspA2.bak	omni Pict x Publ	names-ras sures lic	pA2.log	python_, Templati Videos	games es	
client_ Desktop Documen pi@rasp	secret.json nts A2:~~\$	Downloads Music omninames-raspA2.bak	omni Pict x Publ	names-ras sures lic	pA2.log	python⊥ Templati Videos	games es	
client_ Desktop Documen pi@rasp	secret.json nts A2:~~\$	Downloads Music omninames-raspA2.bak	omni Pict K Pub	names-ras sures ic	pA2.log	python⊥ Templati Videos	games es	
client_ Desktop Documen pi@rasp	secret.json nts A2:~~\$	Downloads Music omninames-raspA2.bak	omni Pict K Pub	names-ras sures ic	pA2.log	python_J Templati Videos	games es	

2. パッケージを最新に更新。

\$ sudo apt-get update

 $\$ sudo apt-get upgrade

3. Gmail API に必要な Python パッケージをインストール。

\$ sudo pip install google-api-python-client
\$ sudo pip install oauth2client
\$ sudo pip install apiclient

2.2.4 正常に動作するか確認するために Quickstart を実行

1. 下記ソースコードを「quickstart.py」として保存し, Raspberry Piの 「client_secret.json」ファイルと同一ディレクトリにコピー。

```
#Raspberry Pi
from googleapiclient.discovery import build
from httplib2 import Http
from oauth2client import file, client, tools
# Setup the Gmail API
SCOPES = 'https://mail.google.com/'
store = file.Storage('credentials.json')
creds = store.get()
if not creds or creds.invalid:
   flow = client.flow_from_clientsecrets('client_secret.json', SCOPES)
   creds = tools.run_flow(flow, store)
service = build('gmail', 'v1', http=creds.authorize(Http()))
# Call the Gmail API
results = service.users().labels().list(userId='me').execute()
labels = results.get('labels', [])
if not labels:
   print"No labels found."
else:
   print"Labels:"
   for label in labels:
       #print"label["+name+"]"
       print label['name']
# [END gmail quickstart]
```

コピーが出来ない方は以下 URL からダウンロードしてください。

https://rtc-fukushima.jp/wp/wp-content/uploads/2019/01/quickstart.zip

このプログラムは Google が確認用に配布しているプログラムを Raspberry Pi 様に編集したものです。

Python Quickstart

https://developers.google.com/gmail/api/quickstart/python ※参照したのが 6 月でしたので現在のものとは異なります。

2. 下記のコマンドを実行。

\$ python quickstart.py --noauth_local_webserver

[--noauth_local_webserver]はこの[quickstart.py]を実行したコンピュータとは別のブラウザ を使う必要があるときに付けるオプションです。今回は TerTerm で Raspberry Pi にアクセス し、プログラムを実行しています。ブラウザは PC 側で起動する必要があるのでこのオプショ ンを付けます。

3. 表示された URL を Windows で起動しているブラウザに入力。



G Google	× +		-	пх
\leftrightarrow \rightarrow G	G https://accounts.google.com/o/oa	uth2/auth?scope=https%3A%2F%2Fwww.googleapis.com%2Fauth%2Fgmail.readonly&redirect_uri=urn%3Aietf	⁶ %3A	*® :
		Gmail 画像		太郎
		Google		
		Ļ		
		Google 検案 I'm Feeling Lucky		
日本				
広告 ビミ	ネス Googleについて	プライバシー	- 規約	設定
client_sec	ret_1053json ^		すべて	表示 X

4. 自分のアカウントを選択。

G ログイン - Google アカウント × +			-		×	
\leftarrow \rightarrow C $($ https://accounts.google.com/signi	n/oauth/oauthchooseaccount?client_id=10533130)89373-apmhnegg912r6na9ndr89p2tjndo6ekk.apps.goog	euser	<u>ل</u>) :	
	G Google にログイン アカウントの選打 「GmailTest」に移動 @gmail.com ② 別のアカウントを使用	R の権限の作 M Gmailのすべてのメールの 作成、送信、完全な削除 拒	サチの閲覧、	許可	、 出) レ	のウィンドウが たら[許可]を押す
□ client_secret_1053json ^	L					

5. 「許可」ボタンを押下。

G Google にログイン	
選択内容を確認してください @gmail.com	
gmailtest に次のスコープを許可しています。	
✓ Gmall のすべてのメールの閲覧、作成、送信、完 全な削除	
gmailtest を信頼できることを確認	
機密情報をこのサイトやアプリと共有する場合があり ます。gmailtestの利用規約とプライバシーポリシー で、ユーザーのデータがどのように取り扱われるかを ご確認ください。アクセス権の確認、削除は、 Google アガウントでいつでも行えます。	
リスクの詳細	
キャンセル 許可	

6. 表示されたコードをコピーし, TeraTerm の"Enter verification code:"に張り付ける。



_			
raspA2.local - pi@raspA2: ~ VT	_		×
ファイル(E) 編集(E) 設定(S) コントロール(Q) ウィンドウ(W) ヘルプ(H)			
pi@raspA2: \$ python quickstart.pynoauth_local_webserver /usr/local/lib/python2.7/dist-packages/oauth2client/_helpers.py: : Cannot access credentials.json: No such file or directory warnings.warn(_MISSING_FILE_MESSAGE.format(filename))	255: l	lser₩arı	ning
Go to the following link in your browser:			
https://accounts.google.com/o/oauth2/auth?scope=https%3A%2F% .com%2Fauth%2Fgmail.readonly&redirect_uri=urn%3Aietf%3Awg%3Aoautl sponse_type=code&client_id=819553718199-okgmh0458t4ub0ouu5mkv7jd ogleusercontent.com&access_type=offline	2Fwww. h%3A2. ubmdr9	<u>google</u> 0%3Aool 4p.apps	apis o&re s.go
Enter verification code: 4/3ADQ0XqyFXyJhy0-JQZjjV8ne7dJst1-6uYzlo 9o	qOhLCK	sQQaJw	bB0S
			I
			\checkmark



プログラムが実行されました。

これは登録した Gmailのラベルを表示しております。

プログラムが実行されると同時に「credentials.json」が作成されます。このファイルには認 証情報が登録されます。以降はこのファイルを参照するので、ブラウザを使用した認証はなし で実行が出来ます。

2.3 Python でメールを出す

1. MailSend.py を作成し, Raspberry Pi にコピーしてください。

```
import httplib2
import os
#Raspberry Pi
import googleapiclient
from googleapiclient.discovery import build
from oauth2client import file, client, tools
from httplib2 import Http
import base64
from email.mime.text import MIMEText
from email.utils import formatdate
SCOPES = "https://www.googleapis.com/auth/gmail.send"
FROM = # 自分のメールアドレス#
TO = # 自分のメールアドレス#
def create_message():
   message = MIMEText("Gmail body: Hello world!")
   message["from"] = FROM
   message["to"] = TO
   message["subject"] = "gmail api test"
   return {'raw': base64.urlsafe_b64encode(message.as_string())}
def main():
   store = file.Storage('credentials.json')
   creds = store.get()
   if not creds or creds.invalid:
       flow = client.flow_from_clientsecrets('client_secret.json', SCOPES)
       creds = tools.run_flow(flow, store)
   service = build('gmail', 'v1', http=creds.authorize(Http()))
   try:
       result = service.users().messages().send(
       userId=FROM, body=create_message()).execute()
       print("Message Id: {}".format(result["id"]))
   except:
       print("----errer-----")
if __name__ == "__main__":
   main()
```

[MailSend.py]でファイルを作成してください。メールアドレスの欄は先ほど設定の時に使用 した Gmail のアドレスを入れてください。文字列なのでダブルクォートで囲む必要がありま す。

コピーが出来ない方は以下 URL からダウンロードしてください。

https://rtc-fukushima.jp/wp/wp-content/uploads/2019/01/MailSend.zip

メールを出すプログラムは以下のページを参考に作成されています。

• Users.messages: send

https://developers.google.com/gmail/api/v1/reference/users/messages/send

• Sending Email

https://developers.google.com/gmail/api/guides/sending

2. プログラムの実行
 下記コマンドを実行してください。

\$ python MailSend.py --noauth_local_webserver

下図の様に結果が出た場合は、[5.]で自分の Gmail を確認してください。

🔳 raspA	2.local - pi	@raspA2	= ~ VT			-	×
7711UE	編集(E)	設定(5)	コントロール(Q)	ウィンドウ(図)	ヘルブ(圧)		
pi@rasp Message pi@rasp	A2: \$ 1 Id: 168 A2: \$	oython 37ebcbf	MailSend.r 184fb7dc	ynoaut	h_local_webserver		^

URL が表示された人は[3.]以降の作業をしてください。

3. 表示された URL を Windows で起動しているブラウザに入力。



G Google	×	+						- []	×
← → C ()	https://accounts.g	google.com/o/oa	auth2/auth?scope=https%3A%2F%	62Fwww.googleapis.com%2	Prauth%2Fgmail.send&redired	ct_uri=urn%3Ai	etf%3Awg	1%	* #	:
						Gmail 運	i像	0	太魚	5
			G							
				ogie						
					Ŷ					
			Google 検索	I'm Feeling Lucky						
日本										
広告 ビジネス	Google(こついて	5				プライ	イバシー	規約	設力	Ē

「quickstart.py」の時と同じように認証をしてください。コードを表示してください。

4. 表示されたコードをコピーし, TeraTerm の"Enter verification code:"に張り付ける。



以下の様に結果が表示されます。



5. Gmail にログインし、メールを受信したことを確認。

G Success code=	/wQD1dC0888Fz × 新しいタブ × M gmail api test - aizutaro.1227@g × +				-	- 🗆	×
← → C I	https://mail.google.com/mail/u/0/#inbox/FMfcgxwBTsWRDwstHZLTNmJtdLTpTGkC				۲	☆	*® :
≡ M (Gmail Q メールを検索		Ŧ			0	太郎
÷	0 0 1 0 b •	1/2	<	> 3	あ -	\$	31
	gmail api test 受信トレイ×				ē	ß	Ø
* 🔺	@gmail.com To 自分 ¥	17:43 (0 分前)	☆	*	:	0
	Cのメールにはご注意ください Gmailでは、このメールが本当に aizutaro.1227@gmail.com から送信されたものであることを確認できませんで メールに含まれるリンクのクリックや添付ファイルのダウンロード、または返信に個人情報を記載することは過 ださい。 スパムを報告 フィッシングを報告	でした。 計てく			0		+
最近 ヤツ	Gmail body: Hello worldl						
あり	 ▲ 返信 ➡ 転送 						>

今回は自分から自分に送付しているので,スパム扱いになっていますが,別のメールアドレス 宛てにすれば問題はございません。

3 温度を取得してメールで送信するプログラムの作成

Raspberry Pi に温度センサの回路を接続し、取得した温度をメールで送信するプログラムを作成します。先ほど作成したプログラムに温度センサから値を取得するプログラムを組み合わせて、実行してください。

3.1 温度センサを接続

前日の内容を参考に、Raspberry Pi に温度センサを接続してください。

3.2 温度を取得して送信するプログラムを作成

温度を取得して送信するプログラムを作成。 プログラム名は[MailSend_Raspberry_temp.py]で作成してください。

3.3 プログラムの実行

\$ python MailSend_Raspberry_temp.py --noauth_local_webserver

4 温度を取得してメールを1分毎に送信するプログラム

温度をメールで送付するプログラムを作成しました。次はこのプログラムを1分毎に起動する ように設定しましょう。

動かす方法はシェルスクリプトを使用します。

4.1 シェルスクリプトとは

シェルで実行するコマンドを連続で実行するためのプログラムのことです。 このプログラムは上から順に実行されるので、複数のコマンドを実行する作業などをシェルス クリプト内に書き込むと便利です。

4.2 シェルの書き方

4.2.1 作り方・実行の仕方

シェルファイルの作り方と実行の仕方を説明します。 [test.sh]で保存して Raspberry Pi に転送してください。

#!/bin/sh

echo 'hello world'

転送後以下のコマンドで実行してください。

```
$ chmod 755 test.sh
$ sed -i 's/¥r//' test.sh
$ ./test.sh
```

実行結果として[hello world]と表示されます。

シェルプログラムのファイルは[.sh]の形式で保存されます。そのファイルを[./ファイル名]で 実行します。

[#!/bin/sh]はシェルスクリプトであるという意味になります。[echo]は出力を表します。

4.2.2 変数

シェルスクリプトでも変数を使うことが出来ます。数字や文字列を代入することができます。

変数への代入は[=]で行います。その時スペースを入れないようにしてください。変数を echo で表示する場合は変数の前に\$を付けます。

#!/bin/bash
num=10
echo \$num
word="hello"
echo \$word

4.2.2.1 変数での計算

変数同士の計算を行う場合は式を[\$(())]で囲む。

#!/bin/bash
a=1
b=4
c=\$((a+b))
echo \$c

4.2.3 条件分

If 文の構文は以下になります。 if 条件文1 then 処理1 elif 条件文2 処理2 else 処理3 fi

条件文は[]でくくります。値の比較は以下の演算子を使います。

演算子	書き方	意味
eq	a -eq b	a=b
ge	a -ge b	a>=b
gt	a -gt b	a>b
le	a-le b	a<=b
lt	a-It b	a≺b
ne	a -ne b	a!=b

条件文の[]や比較演算の間にはスペースを入れてください。

```
#!/bin/bash
a=1
b=2
if [ $a -eq $b ]
then
    echo True
else
    echo False
fi
```

4.2.4 ループ文

while文

While文の構文は以下になります。

```
while 条件文
do
処理を書く
done
```

条件式が真ならばループを続けます。

無限ループの場合は条件文の所を[:]を指定します。

#!/bin/bash
a=1
while :
do
 echo \$a
 a=\$((a+1))
done

ループを抜ける時は[CTR+C]で抜けてください。

4.2.5 コマンドの実行

シェルスクリプト内ではLinuxのコマンドを実行することが出来ます。これを利用してディレクトリの移動やファイルの実行が来ます。

print "Hello World"

[test.py]で保存して Raspberry Pi にホームディレクトリに転送してください。

以下のプログラムを実行してください。

#!/bin/bash
pwd
cd /home/
pwd
cd
python test.py

コマンド[pwd]の結果が2回表示された後、Pythonのファイルが実行されます。

4.2.5.1 Sleep コマンド

sleep コマンドを使うと指定した時間だけ処理を遅延させることが出来ます。

#!/bin/bash
echo 'start'
sleep 10s
echo 'end'

[start]から 10 秒後[end]と表示されます。s は秒, m は分, h は時間を表します。

4.3 シェルスクリプト作成

1分毎に python プログラムを実行するシェルを作成してください。