
ハンズオン Step4:

TOP ページを APL で動かしてみる

目次

目次.....	2
【Step4】 TOP ページを APL で動かしてみる	4
Step4 のゴール	4
Step4 で作る対話モデル	4
APL Document の改修.....	5
Lambda の改修	7
テスト	8
クイズ.....	8

【Step4】 TOP ページを APL で動か してみる

Step4 のゴール

Step4 では TOP ページを一定時間ごとにスライドショー形式で動かしてみます。

以下のことができるようになります。

- APL ページによるスライドショーの仕組みを理解する

Step4 で作る対話モデル

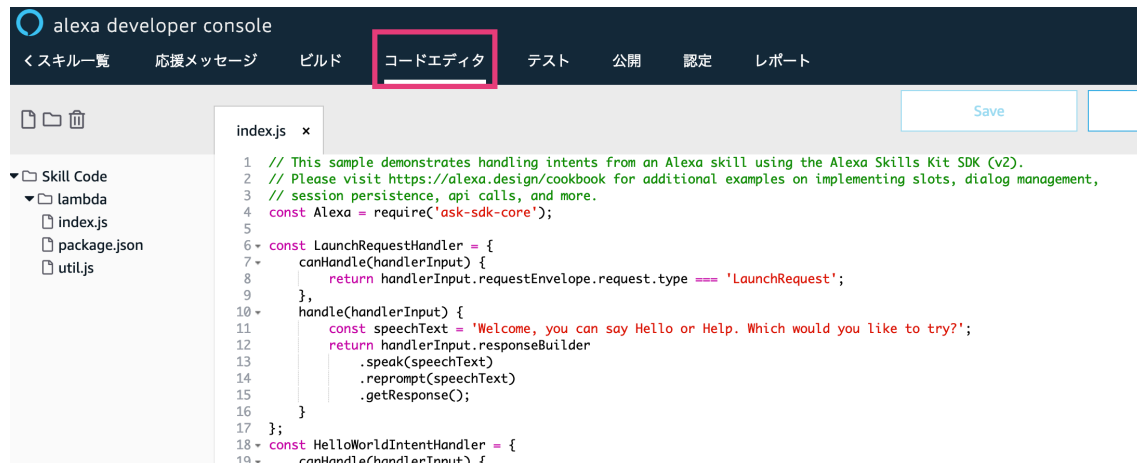
※Step1 と同じです

ポイント

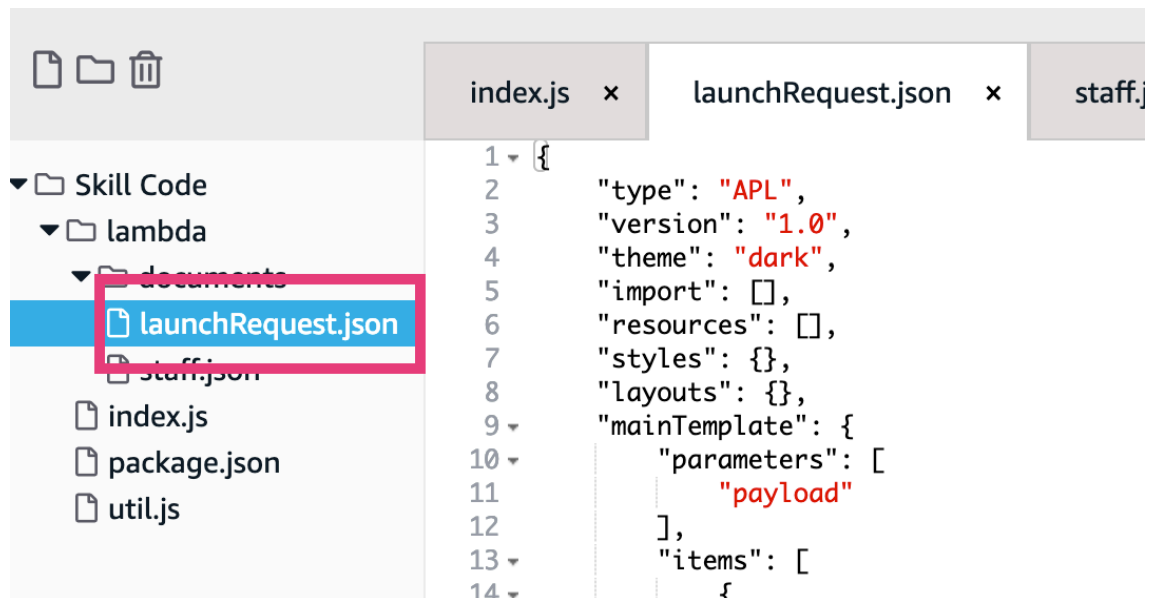
- APL Command を使用して自動的にページング処理を行います。

APL Document の改修

1. 上部メニューより「コードエディタ」をクリックし、Lambda のコード画面を表示します。

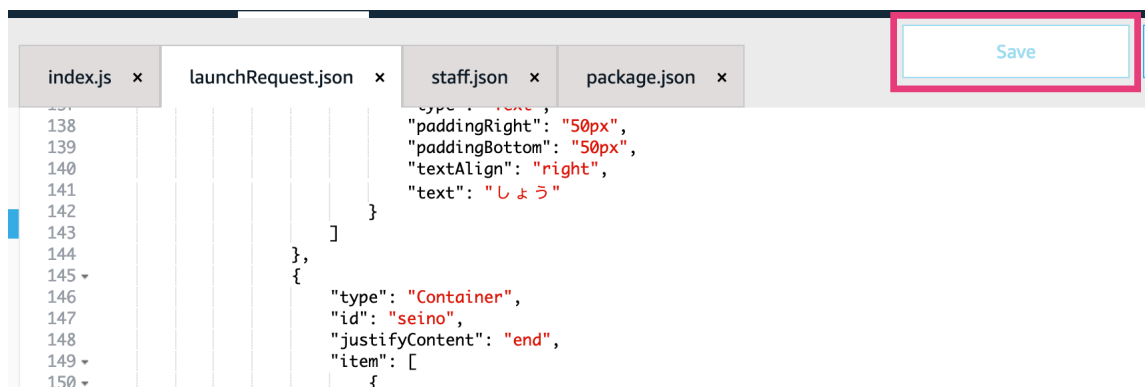


2. documents フォルダの中の launchRequest.json ファイルをダブルクリックして画面上に開きます。



4. 画面上の launchRequest.json にダウンロードしたハンズオン資料より STEP4 -> lambda -> launchRequest.json のファイルを開いてコピー&ペーストします。終わりましたら画面右上部にある「Save」をクリックします。

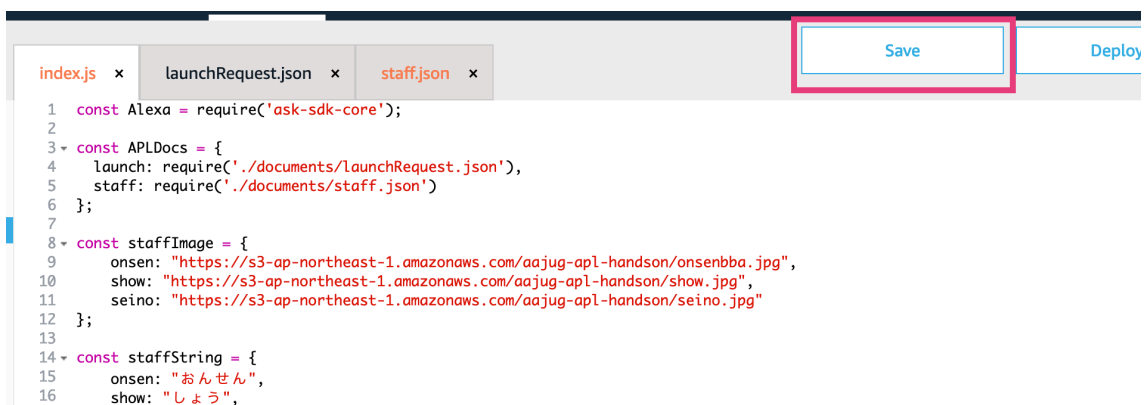
※日本語が文字化けしている方は launchRequest-shift-jis.json を開くか、エディターの文字コードを「UTF-8」にして launchRequest.json を開いてみてください。



これで APL Document の改修は完了です。次は Lambda を改修します。

Lambda の改修

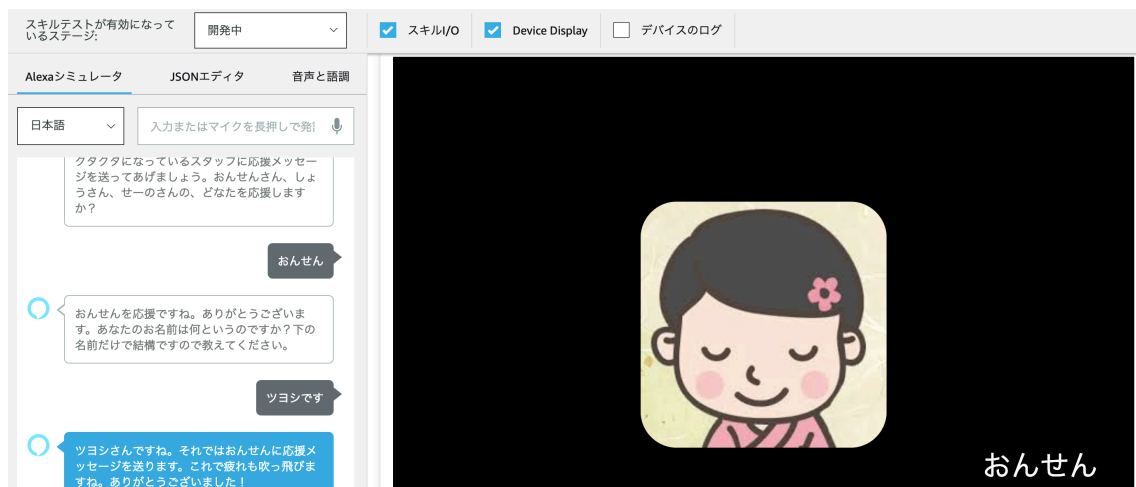
1. index.js を画面上に開き、ダウンロードしたハンズオン資料より STEP4 -> lambda -> index.js のファイルを開いてコピー&ペーストします。終わりましたら画面右上部にある「Save」をクリックし、その横の「Deploy」をクリックします。



```
1  const Alexa = require('ask-sdk-core');
2
3  const APLDocs = {
4    launch: require('./documents/launchRequest.json'),
5    staff: require('./documents/staff.json')
6  };
7
8  const staffImage = {
9    onsen: "https://s3-ap-northeast-1.amazonaws.com/aajug-apl-handson/onsenbba.jpg",
10   show: "https://s3-ap-northeast-1.amazonaws.com/aajug-apl-handson/show.jpg",
11   seino: "https://s3-ap-northeast-1.amazonaws.com/aajug-apl-handson/seino.jpg"
12 };
13
14 const staffString = {
15   onsen: "おんせん",
16   show: "しょう",
```

テスト

1. STEP1 同様テストシミュレーターから「応援メッセージ」でスキルを開いて、動いているかどうか確認しましょう。メッセージの途中で画面が変わったら成功です。



クイズ

STEP3 と STEP4 の index.js は具体的にどこが違うでしょう？該当する箇所にコメントを入れてみてください。