Ex : Basic Wordle Backend API using Express

This is what is being called when the Lambda function is triggered.

index.ts

src > index.ts > app.get("/word") callback > word

```ts
import express, { Request, Response } from 'express'
import cors from 'cors';
import serverless from 'serverless-http';
import {pickRandomWord, checkWord} from "./words";

const app  = express()
app.use(express.json()); // for parsing application/json
app.use(cors()); // adds cors headers

// Gets a word form the daily wordle set
app.get("/word", (req : Request, res : Response) => {
    const word = pickRandomWord()
    res.send({
        word : word
    });
});

// Checks if a word is correct
app.post('/word', (req : Request, res : Response) => {
    const body = req.body;
    if (body === undefined){
        res.status(401).send({
            anwser : "Invalid body"
        });
    }
    const word = body.word;
    if(word === undefined || word.length !== 5){
        res.status(401).send({
            anwser : "Invalid word"
        });
    }
    res.send({
        anwser : checkWord(word)
    });
}
)

export const handler = serverless(app)
```

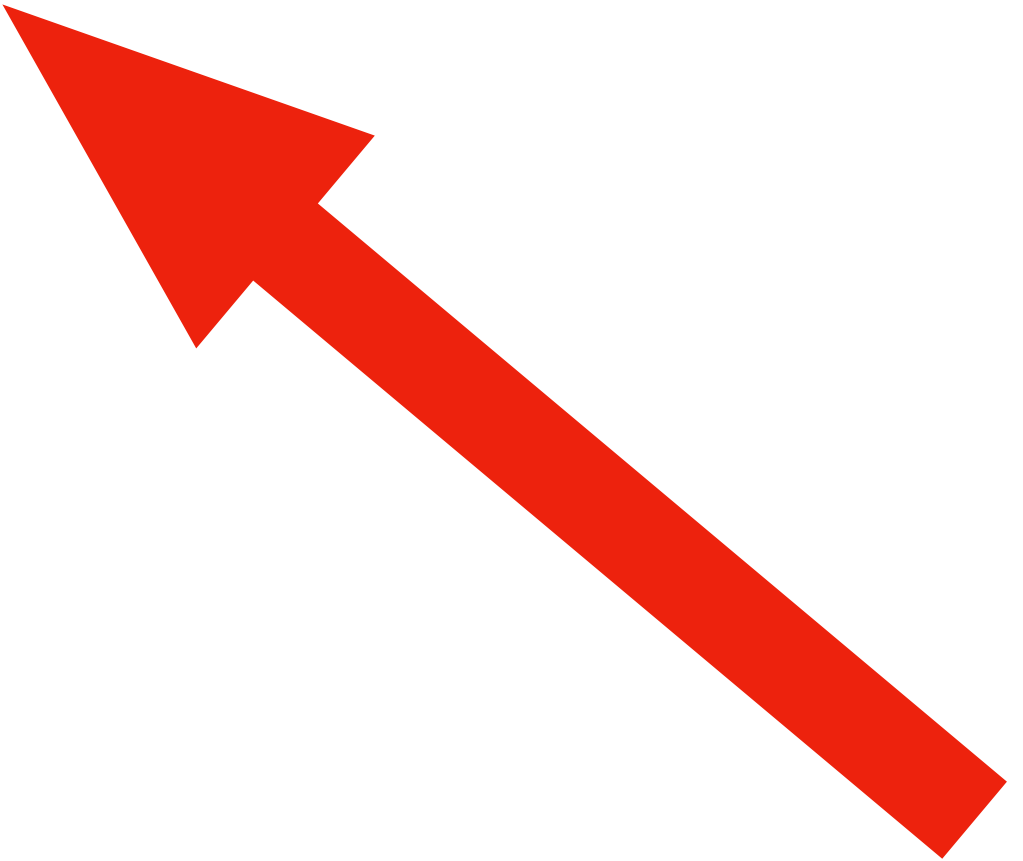
words.ts

src > words.ts > checkWord

```ts
    "zebra", "zesty", "zonal"]

// Picks a random word from the list based on Today
export function pickRandomWord() : string{
    // Applies a hashing function for the current day to get a word
    const now = new Date(Date.now()); // Uses UTC time
    const day = now.getDate();
    const month = now.getMonth();
    const year = now.getFullYear();
    // Djb2 hash function
    let hash = 5381
    hash = hash * 33 + day;
    hash = hash * 33 + month;
    hash = hash * 33 + year;
    return words[hash % words.length];
}

// Checks the word against the word of the day
export function checkWord(word : string){
    const todayWord = pickRandomWord();
    if (words.includes(word) === false) { // Is it a valid word?
        return "Not a word";
    }
    return word.split("").map((letter, index) => {
        if(letter === todayWord[index]){
            return "Correct"
        }
        else if (todayWord.includes(letter)){
            return "Misplaced"
        }
        else{
            return "Not in word"
        }
    })
}
```



Ex : Basic Wordle Backend API using Express

index.ts

```ts
import express, { Request, Response } from 'express'
import cors from 'cors';
import serverless from 'serverless-http';
import {pickRandomWord, checkWord} from "./words";

const app  = express()
app.use(express.json()); // for parsing application/json
app.use(cors()); // adds cors headers

// Gets a word form the daily wordle set
app.get("/word", (req : Request, res : Response) => {
    const word = pickRandomWord()
    res.send({
        word : word
    });
});

// Checks if a word is correct
app.post('/word', (req : Request, res : Response) => {
    const body = req.body;
    if (body === undefined){
        res.status(401).send({
            anwser : "Invalid body"
        });
    }
    const word = body.word;
    if(word === undefined || word.length !== 5){
        res.status(401).send({
            anwser : "Invalid word"
        });
    }
    res.send({
        anwser : checkWord(word)
    });
}
)

export const handler = serverless(app)
```

words.ts

```ts
// Picks a random word from the list based on Today
export function pickRandomWord() : string{
    // Applies a hashing function for the current day to get a word
    const now = new Date(Date.now()); // Uses UTC time
    const day = now.getDate();
    const month = now.getMonth();
    const year = now.getFullYear();
    // Djb2 hash function
    let hash = 5381
    hash = hash * 33 + day;
    hash = hash * 33 + month;
    hash = hash * 33 + year;
    return words[hash % words.length];
}

// Checks the word against the word of the day
export function checkWord(word : string){
    const todayWord = pickRandomWord();
    if (words.includes(word) === false) { // Is it a valid word?
        return "Not a word";
    }
    return word.split("").map((letter, index) => {
        if(letter === todayWord[index]){
            return "Correct"
        }
        else if (todayWord.includes(letter)){
            return "Misplaced"
        }
        else{
            return "Not in word"
        }
    })
}
```

This is what is being called when the Lambda function is triggered.

Compiling & Project Structure