

# International TEDays

## HOMEWORK 3

Link utili:



Andrea Appiani – 1057683

Luca Mazzoleni – 1054271

Cristian Sacco – 1063539

Alexandr Rucodainii – 1045954

Nunzio Marco Bisceglia – 1046319

# LAMBDA FUNCTIONS

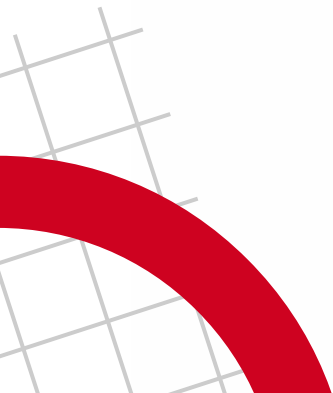


## `getWatchNextbyId`

Ritorna l'elenco dei Watch Next del TEDx Talk con l'ID indicato

## `getTEDxByInternationalDay`

Ritorna l'elenco dei TEDx Talks riguardanti un tema trattato o una ricorrenza celebrata durante una delle giornate internazionali per le Nazioni Unite





**getWatchNextById**

# LF getWatchNextById

Lo schema dei TEDx Talks usati per la Lambda Function viene definito nel file Talk.js

```
const mongoose = require("mongoose");

const talk_schema = new mongoose.Schema(
  {
    _id: String,
    title: String,
    url: String,
    watch_next: Array,
  },
  { collection: "tedx_data" }
);

module.exports = mongoose.model("talk", talk_schema);
```

# LF getWatchNextById

Dopo aver importato i file db.js e Talk.js viene definita la struttura della richiesta

```
const connect_to_db = require("../db");
const talk = require("../Talk");

module.exports.get_watch_next_by_id = (event, context, callback) => {
  context.callbackWaitsForEmptyEventLoop = false;

  console.log("Received event:", JSON.stringify(event, null, 2));

  let body = {};
  if (event.body) {
    body = JSON.parse(event.body);
  }

  if (!body.id) {
    callback(null, {
      statusCode: 500,
      headers: { "Content-Type": "text/plain" },
      body: "Could not fetch the watch next talks. Id is null.",
    });
  }
}
```

# LF getWatchNextById

Si cerca nel database il TEDx Talk con l'id specificato nella richiesta e viene ritornato il titolo e l'array dei Watch Next nella risposta.

```
connect_to_db().then(() => {
  console.log("=> get_all watch next talks");
  talk
    .find({ _id: body.id })
    .then((response) => {
      callback(null, {
        statusCode: 200,
        body: JSON.stringify({
          name: response[0].title,
          watch_next: response[0].watch_next,
        }),
      });
    })
    .catch((err) => {
      callback(null, {
        statusCode: err.statusCode || 500,
        headers: { "Content-Type": "text/plain" },
        body: "Could not fetch the watch next talks.",
      })
    });
});
```

# LF getWatchNextById

La Lambda Function viene esposta tramite AWS API Gateway e il risultato è il seguente:

```
{  
  "name": "The intangible effects of walls",  
  "watch_next": [  
    "https://www.ted.com/talks/alex\_honnold\_how\_i\_climbed\_a\_3\_000\_foot\_vertical\_cliff\_without\_ropes",  
    "https://www.ted.com/talks/will\_hurd\_a\_wall\_won\_t\_solve\_america\_s\_border\_problems",  
    "https://www.ted.com/talks/ronald\_rael\_an\_architect\_s\_subversive\_reimagining\_of\_the\_us\_mexico\_border\_wall",  
    "https://www.ted.com/talks/megan\_campisi\_and\_pen\_pen\_chen\_what\_makes\_the\_great\_wall\_of\_china\_so\_extraordinary",  
    "https://www.ted.com/talks/anna\_heringer\_the\_warmth\_and\_wisdom\_of\_mud\_buildings",  
    "https://www.ted.com/talks/julia\_dhar\_how\_to\_disagree\_productively\_and\_find\_common\_ground"  
  ]  
}
```

L'utente ha così la possibilità di visualizzare i Watch Next e il nome del TEDx Talk a partire dall'ID inserito.



**getTEDxByInternationalDay**



# LF getTEDxByInternationalDay



## IDEA ALLA BASE

Effettuare un **match** tra le parole chiave (**keywords**) all'interno dell'evento celebrato (dato ottenuto dal dataset **international\_days**) e i **tags** dei TEDx Talks



## ESEMPIO

Il 22 marzo si celebra la **World Water Day**  
La Lambda Function, trovata la giornata nel database, filtra la stringa, estrae la parola chiave **water** e restituisce tutti i TEDx Talks che includono il tag **water**

# LF getTEDxByInternationalDay

## Implementazione:

- Importazione dei files Talk.js e Day.js che fungono da modello dati e del file db.js per la connessione al database
- Ricerca del giorno, passato come parametro, all'interno del dataset delle giornate internazionali
- Creazione di un array contenente sole parole chiave tramite la funzione **getKeywords**. Questa effettua una serie di trasformazioni a partire dal nome dell'evento e da un array di parole da rimuovere caricato da un file .txt.

## Esempio:

"World Water Day" -> ["World", "Water", "Day"] -> ["world", "water", "day"] -> filtro -> ["water"]

```
const getKeywords = (event, wordsToRemove) =>
  event
    .split(" ")
    .map((el) => el.toLowerCase())
    .filter((el) => !wordsToRemove.includes(el));
```

# LF getTEDxByInternationalDay

Successivamente si cercano nel database tramite l'operatore **\$in** i TEDx Talks che hanno almeno un tag corrispondente alle keywords trovate; infine, viene restituito il risultato.

```
talk
  .find({ tags: { $in: keywords } })
  .then((response) => {
    let teds = [];
    response.forEach((ted) => teds.push(ted.url));

    let result = {};
    result["days"] = days;
    result["tags"] = keywords;
    result["teds"] = teds;

    callback(null, {
      statusCode: 200,
      body: JSON.stringify(result),
    });
  })
```

# LF getTEDxByInternationalDay

La risposta della Lambda Function tramite AWS API Gateway è composta da:

- **days:** le giornate internazionali celebrate nel giorno inserito
- **tags:** le parole chiave che hanno permesso il match keywords-tags
- **teds:** gli URL dei TEDx Talks

```
{  
  "day": [  
    "World Water Day"  
  ],  
  "tags": [  
    "water"  
  ],  
  "teds": [  
    "https://www.ted.com/talks/alex_gendler_how_the_world_s_longest_underwater_tunnel_was_built",  
    "https://www.ted.com/talks/stefan_al_why_isn_t_the_netherlands_underwater",  
    "https://www.ted.com/talks/yves_behar_how_surfboards_connect_us_to_nature",  
    "https://www.ted.com/talks/angelicque_white_what_ocean_microbes_reveal_about_the_changing_climate",  
    "https://www.ted.com/talks/kelsey_leonard_why_lakes_and_rivers_should_have_the_same_rights_as_humans",  
  ]  
}
```

L'utente in questo modo, inserita una data, può conoscere se ci sono giornate internazionali e guardare TEDx Talks riguardanti l'argomento.

# CRITICITÀ

- Per alcuni giorni vengono restituiti molti TEDx Talks, a causa di **tag troppo comuni** (es. war). Di conseguenza alcuni TEDx Talks potrebbero essere **non inerenti** all'argomento esatto della giornata internazionale;
- In termini di prestazioni, restituire molti TEDx Talks significa **rallentare il tempo di risposta** dell'API;
- Al contrario, alcune giornate internazionali **non hanno TEDx Talks associati**.

# POSSIBILI EVOLUZIONI

- Le parole chiave sono ottenute da un filtro che utilizza un set di parole da file di testo. L'elenco delle parole da rimuovere è sicuramente **ampliabile** per garantire un filtro più efficace, senza però rimuovere parole rilevanti per il match con i tags dei TEDx Talks;
- Due o più giornate internazionali possono essere celebrate nello **stesso giorno**: in questo caso il servizio restituisce un unico array di parole chiave. Si potrebbe pensare di restituire TEDx Talks per ciascuna giornata in array diversi



# TEAM

Andrea Appiani 1057683

Luca Mazzoleni 1054271

Cristian Sacco 1063539

Alexandr Rucodainii 1045954

Nunzio Marco Bisceglia 1046319

