

サルベジオン問題

© 2014 結城浩

<http://www.hyuki.com/codeiq/>

2014 年 11 月～12 月



1 概要

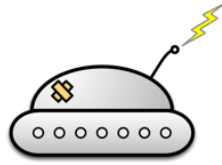
あなたは サルベジオン 社 (Salvageon) にプログラマとして雇われました。サルベジオン社は、難破した無人宇宙船に格納されているデータ救出を行う会社で、遠い星の無人宇宙船と通信を行ってデータ解析を行うのが得意としています。

あなたに与えられるのは無人宇宙船のデータベースにアクセスする Web API ひとつ。この Web API を使って データを見つけ出す のがあなたのミッションなのです！

2 会話

あなた「データを見つけ出す仕事と聞きましたが、いったいどういうことでしょうか……」

依頼者「順を追って説明しましょう。わがサルベジオン社は、難破した無人宇宙船に格納されているデータ救出を行う会社です。最近、遠い星に無人宇宙船を飛ばすことが多くなってきましたが、経費削減のありで事故も多い。せっかく飛ばした無人宇宙船が難破してしまうこともしばしばあります」



難破した無人宇宙船

あなた「はい」

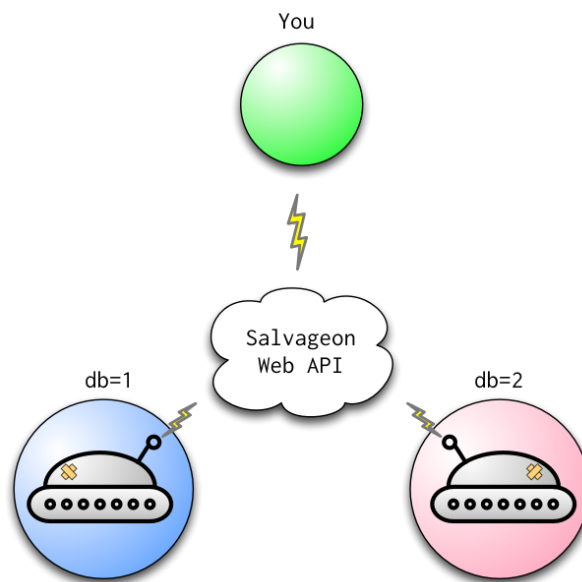
依頼者「無人宇宙船全体を修理してまた地球まで飛ばすのは費用も掛かりますから、せめてデータベースだけでもサルベージ（救出）したいという要望が多く、わが社はそこでビジネスをしているわけです」

あなた「なるほど。御社の社名、サルベージオンもそこから？」

依頼者「そういうことです。現在も二台分のデータベース・サルベージ作業を行っていて、あなたにはその業務を手伝っていただきます」

あなた「わかりました。しかし、私には無人宇宙船の知識はありませんが……」

依頼者「それはご心配なく、すでに無人宇宙船の間には通信チャンネルが確立していますので、あなたには、サルベージオン社の **Web API** を使ってデータベースにアクセスしていただきます。そして、与えられたキーを持つ **データを見つけ出す** のがお仕事となります」



サルベージオン社 Web API

あなた「たとえば、SQL のような言語を使うことになりますか？」

依頼者「違います。遠い星で故障している無人宇宙船ですから、そんなに高度なアクセスはできません。アクセス方法は以下の通りです」

- アクセスには Web API を用いる。
- 個々のデータベースは、データベース番号 (db) で区別する。
- 個々のデータベースは配列のように見える。
- 非負整数のインデクス (index) を与えると、キー (key) とバリュー (value) のペアが得られる。

あなた「なるほど。データベース番号とインデクスが入力で、キーとバリューのペアが出力ということですね」

依頼者「そういうことです。データベースが生きていたらキーを与えてバリューが得られるはずなのですが、あちこち壊れているために、このような原始的な形でアクセスしかできないのです」

あなた「概要はわかりました」

依頼者「あなたにお願いしたいのは、データベースの中から、与えられたキーに対応するデータを見つけ出すことです」

あなた「与えられたキーを持つデータは、データベースの中に必ず存在するのでしょうか？」

依頼者「はい、存在します。また、一つのデータベースの中ではキーに重複はありません。バリューには重複があるかもしれませんが」

あなた「了解です。あとは具体的な情報をいただければすぐに仕事にかかれると思います」

依頼者「では具体的な例で Web API の説明をしましょう。たとえば、このようなデータベースがあります」

index	key	value
0	K6742	V02176333
1	K5499	V05677882
2	K3668	V79713413
3	K9396	V29970163
4	K0929	V60983729

db が 0 の例

依頼者「このデータベースのデータベース番号 (db) は 0 です。このデータベースには全部で 5 個のデータ、すなわちインデクスとバリューのペアが 5 組格納されています」

あなた「よくわかります。たとえばインデクスが 2 のデータは、キーが K3668 で、バリューが V79713413 ということですね？」

依頼者「その通りです。このデータベースからキーが K9396 であるバリューを探してほしいといわれれば、V29970163 を答えていただければよろしいということになります」

あなた「なるほど。ところで、キーは K の後に数字列、バリューは V の後に数字列が続く形式ですね」

依頼者「そうです。それはどのデータベースも同じです」

あなた「キーの順番はどうなっているのでしょうか」

依頼者「データベースごとにルールがあるらしいのですが、ドキュメントが残されていないので判明していません」

あなた「(そんなあ!) ……な、なるほど。そうですか。そこは調査が必要になりますね」

依頼者「データベース番号 (db) が 0 であるデータベースへアクセスして、インデクス (index) が 2 であるデータを得るとき、Web API はこう呼び出します」

`http://salvageon.textfile.org/?db=0&index=2`

データベース番号が 0 で、インデクスが 2 の呼び出し

あなた「なるほど。素直な呼び出しですね」

依頼者「この結果はテキストで以下のように返されます」

`0 2 K3668 V79713413 5`

データベース番号が 0 で、インデクスが 2 の呼び出し結果

あなた「0 はデータベース番号、2 は与えたインデクス、K3668 は得られたキーで、V79713413 は得られたバリューですね。最後の 5 は何ですか？」

依頼者「ああ、データベースのサイズですよ。つまりこのデータベースに与えることができる最大のインデクスに 1 加えた値ですね」

あなた「なるほど、わかりました」

依頼者「一般的に書けば、入力はこちらになります」

`http://salvageon.textfile.org/?db=DB&index=INDEX`

Web API 呼び出しの形式

依頼者「この URL で DB の部分にはデータベース番号を与え、INDEX にはインデクスを与えます。インデクスに与えるのは十進数の非負整数です」

あなた「はい」

依頼者「この URL にアクセスすると結果は、HTTP のレスポンスとしてデータベース番号 (DB)、インデクス (INDEX)、キー (KEY)、バリュー (VALUE)、そしてデータベースサイズ (SIZE) がスペース区切りで返されてきます。Content-type は text/plain です。ただしエラーの場合には ? という一文字が返されます」

`DB INDEX KEY VALUE SIZE`

Web API 呼び出し結果の形式

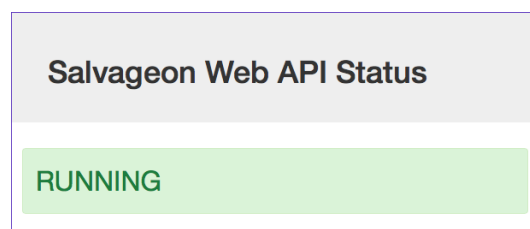
あなた「有効なインデックスの範囲は、0 以上 SIZE 未満の整数すべてですね？」

依頼者「その通りです。ただし、SIZE はデータベース番号ごとに異なります」

あなた「整理させてください。こういうことでしょうか」

- データベース番号ごとに、探すべきデータのキーが与えられる。
- Web API を使ってキーを見つけ出し、それに対応するバリューを得る。

依頼者「その通りです。なお、Web API にアクセスできないときには、以下の URL にアクセスしてみてください。通常は RUNNING とのみ表示されていますが、非常時の連絡事項がそこに表示される場合があります」



<http://salvageon.textfile.org/status.html>

あなた「わかりました。では、プログラムを作りますので、データベース番号とキーの一覧をいただけますか」

依頼者「探すデータは二件です。ファイル `keys.txt` としてお渡しします」

db	key
1	K208050656559285601386927895421059705239114932023754
2	K2023636070998557444542586045

データベース番号と探すべきデータのキー (`keys.txt`)

あなた「ありがとうございます」

依頼者「そうだ、最後にもう一つ。Web API へのアクセスは過負荷にならないようにしてください。具体的には一回のアクセスごとに約 1 秒、時間をおいてくださいね」

あなた「あつ、はい」

ミッションはこのようにして始まりました……。