

Dataset Pipeline Client

Rafael Campos Nunes

<2019-03-01 sex>

Conteúdo

A elaboração de uma API que disponibiliza um serviço de upload de arquivos sobre o protocolo HTTP se completa com a criação de um cliente onde as seguintes funções foram determinadas:

- Leitura de um diretório de arquivos contendo arquivos JSON
- Envio dos dados ao serviço onde serão persistidos de alguma maneira
- Consulta dos dados através do atributo PK ao serviço disponibilizado

Ao realizar as tarefas denotadas anteriormente a linguagem Go foi escolhida para o desenvolvimento visto que sua biblioteca padrão fornece vários recursos que permitem a escrita da aplicação de maneira simples e fácil além de ser uma linguagem de compilação rápida, contrastando essas características com linguagens como C e C++. A serialização de extração de dados dos arquivos no formato mencionado anteriormente é realizada utilizando o pacote *encoding/json*, o envio destes dados pode ser feita utilizando o pacote *net/http* e a leitura de diretórios/arquivos é realizada utilizando os pacotes *os*, *io* e *io/ioutil*.

Ao ler o diretório bastou usar a função contida no pacote *io/ioutil* denominada *LoadDir* onde é passada uma cadeia de caracteres representando o caminho do diretório no fs relativo ao ambiente de execução do programa. A partir disso é retornado um vetor de *os.FileInfo* onde é possível verificar algumas características do arquivo como se é um diretório e o nome deste.

Ao filtrar os arquivos a serem lidos por extensão e também pela característica de arquivo é denotada uma estrutura que armazena as informações do arquivo lido

Os arquivos são simplesmente arquivos de texto (a extensão não foi determinada) que armazenam, a cada linha deste, um JSON (Javascript Object Notation) de tal forma que a estrutura desse arquivo é tal como abaixo:

```
1 { "pk": "1", "score": "321" }
2 { "pk": "2", "score": "435" }
3 ...
4 { "pk": "101", "score": "13" }
```

Observa-se que o objeto contém dois atributos, o atributo *pk* e o *score*. É fácil, portanto, definir uma estrutura em Go (linguagem utilizada para desenvolver a aplicação) para serialização,

e vice versa com serialização reversa também conhecida neste contexto como unmarshalling¹, de dados.

A extração da informação contida nos arquivos é feita utilizando

O envio dos dados é realizado utilizando a API REST e o console. Ao utilizar a função *send* é executado o carregamento de arquivos dentro do diretório definido previamente na variável *dir* e, após isso, envia-se requisições GET ao ponto da API com as informações necessárias.

¹Serialization in Computer Science available on: <https://en.wikipedia.org/wiki/Serialization>