

Guia básico proto2

Carlos, Jean, Mateus, Victor, Ruan

1

O que é *protocol buffers*?



Protocol Buffers

- Representação externa de dados estruturados.
- Criado pelo *Google*
- Formato de arquivos **.proto** e compilados para linguagens específicas.



Protocol Buffers

- Usado como IDL do google para seus processos de *remote procedure Call* (RPC).
- Compilador protoc gera código C++, java e python.
- Salva os dados em formato binário – Não interpretável por humanos.

2

**Criação de arquivos .proto
com sintaxe proto2**



Criar arquivos .proto

- Tipos de dados: bool, int32, float, double, string.
- Valores padrão: falso para bool, 0 para int e vazio para string.
- É possível aninhar tipos de dados criados.



Criar arquivos .proto

```
syntax = "proto2";
```

```
package tutorial;
```

```
option java_package = "com.example.tutorial";
```

```
option java_outer_classname = "AddressBookProtos";
```

Define a sintaxe do arquivo. Há também a sintaxe proto3



Criar arquivos .proto

```
syntax = "proto2";
```

```
package tutorial;
```

```
option java_package = "com.example.tutorial";
```

```
option java_outer_classname = "AddressBookProtos";
```

Define o pacote em
que o arquivo
gerado fará parte



Criar arquivos .proto

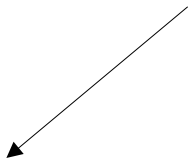
```
syntax = "proto2";
```

```
package tutorial;
```

```
option java_package = "com.example.tutorial";
```

```
option java_outer_classname = "AddressBookProtos";
```

Define o pacote java
da classe gerada





Criar arquivos .proto

```
syntax = "proto2";
```

```
package tutorial;
```

```
option java_package = "com.example.tutorial";
```

```
option java_outer_classname = "AddressBookProtos";
```

Define o nome da
classe gerada





Criar arquivos .proto

```
message Person {  
  required string name = 1;  
  required int32 id = 2;  
  optional string email = 3;  
  
  enum PhoneType {  
    MOBILE = 0;  
    HOME = 1;  
    WORK = 2;  
  }  
  
  message PhoneNumber {  
    required string number = 1;  
    optional PhoneType type = 2 [default = HOME];  
  }  
  
  repeated PhoneNumber phones = 4;  
}
```



Criar arquivos .proto

```
message Person {  
  required string name = 1;  
  required int32 id = 2;  
  optional string email = 3;
```

```
  enum PhoneType {  
    MOBILE = 0;  
    HOME = 1;  
    WORK = 2;  
  }  
}
```

```
message PhoneNumber {  
  required string number = 1;  
  optional PhoneType type = 2 [default = HOME];  
}
```

```
repeated PhoneNumber phones = 4;  
}
```

Tag: Todos os campos tem uma tag associada. É utilizada para codificar os dados para binário. Utilize valores menores que 16 quando possível.



Criar arquivos .proto

```
message Person {  
  required string name = 1;  
  required int32 id = 2;  
  optional string email = 3;
```

```
enum PhoneType {  
  MOBILE = 0;  
  HOME = 1;  
  WORK = 2;  
}
```

```
message PhoneNumber {  
  required string number = 1;  
  optional PhoneType type = 2 [default = HOME];  
}
```

```
repeated PhoneNumber phones = 4;  
}
```

Modificadores: Todos os campos tem um modificador associado: **required**, **optional** ou **repeated**.



Criar arquivos .proto

```
message Person {  
  required string name = 1;  
  required int32 id = 2;  
  optional string email = 3;
```

```
  enum PhoneType {  
    MOBILE = 0;  
    HOME = 1;  
    WORK = 2;  
  }
```

```
  message PhoneNumber {  
    required string number = 1;  
    optional PhoneType type = 2 [default = HOME];  
  }
```

```
  repeated PhoneNumber phones = 4;  
}
```

required: Campo de preenchimento obrigatório. Segundo o *google* é muito difícil tornar um campo obrigatório como opcional.

Quando um campo **required** não é informado é disparada uma exceção



Criar arquivos .proto

```
message Person {  
  required string name = 1;  
  required int32 id = 2;  
  optional string email = 3;
```

```
  enum PhoneType {  
    MOBILE = 0;  
    HOME = 1;  
    WORK = 2;  
  }
```

```
  message PhoneNumber {  
    required string number = 1;  
    optional PhoneType type = 2 [default = HOME];  
  }
```

```
  repeated PhoneNumber phones = 4;  
}
```

optional: Campo de preenchimento facultativo. Se um campo é atualmente obrigatório, porém pode ser opcional no futuro, declare-o como **optional**.

Para checar se o campo opcional foi preenchido: `Person.has_email()`



Criar arquivos .proto

```
message Person {  
  required string name = 1;  
  required int32 id = 2;  
  optional string email = 3;
```

```
enum PhoneType {  
  MOBILE = 0;  
  HOME = 1;  
  WORK = 2;  
}
```

```
message PhoneNumber {  
  required string number = 1;  
  optional PhoneType type = 2 [default = HOME];  
}
```

```
repeated PhoneNumber phones = 4;  
}
```

enum: É possível definir os valores aceitos pelo campo com enum.



Criar arquivos .proto

```
message Person {  
  required string name = 1;  
  required int32 id = 2;  
  optional string email = 3;
```

```
  enum PhoneType {  
    MOBILE = 0;  
    HOME = 1;  
    WORK = 2;  
  }
```

```
  message PhoneNumber {  
    required string number = 1;  
    optional PhoneType type = 2 [default = HOME];  
  }
```

```
  repeated PhoneNumber phones = 4;  
}
```

default: É possível definir um valor padrão para o campo. Caso o usuário não informe um valor, o campo é preenchido com o valor **default**.



Criar arquivos .proto

```
message Person {  
  required string name = 1;  
  required int32 id = 2;  
  optional string email = 3;
```

```
  enum PhoneType {  
    MOBILE = 0;  
    HOME = 1;  
    WORK = 2;  
  }
```

```
  message PhoneNumber {  
    required string number = 1;  
    optional PhoneType type = 2 [default = HOME];  
  }
```

```
  repeated PhoneNumber phones = 4;  
}
```

repeated: Indica que o campo pode ser repetido nenhuma ou várias vezes. É similar a uma lista dinâmica.

Número de itens de um campo
repeated: Person.phones_size()



Criar arquivos .proto

```
message Person {  
  required string name = 1;  
  required int32 id = 2;  
  optional string email = 3;
```

```
  enum PhoneType {  
    MOBILE = 0;  
    HOME = 1;  
    WORK = 2;  
  }  
}
```

```
message PhoneNumber {  
  required string number = 1;  
  optional PhoneType type = 2 [default = HOME];  
}
```

```
repeated PhoneNumber phones = 4;  
}
```

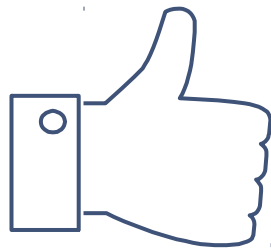
message: Para cada **message** será gerado um objeto, uma classe que pode ser instanciada.



Tipos de dados aceitos – Arquivos .proto

Tipo .proto	Tipo C++	Tipo Java	Tipo Python
double	double	double	float
float	float	float	float
int32	int32	int	int
int64	int64	long	int/long
uint32	uint32	int	int/long
uint64	uint64	long	int/long
sint32	int32	int	int
sint64	int64	long	int/long
fixed32	uint32	int	int
fixed64	uint64	long	int/long
bool	bool	boolean	bool
string	string	String	str/unicode
bytes	string	ByteString	str

Adaptado de <https://developers.google.com/protocol-buffers/docs/proto#scalar>



Tutorial proto3 nos slides!

Documentação proto2:

<https://developers.google.com/protocol-buffers/docs/proto>