# Json与Protocolbuf相互转换

Example serializing to JSON string from Message  
**std::string json\_string = google::protobuf::util::MessageToJsonString(startMovie); // startMovie is a google::protobuf::Message**

Example parsing from JSON string to Message  
**Movie startMovie = google::protobuf::util::JsonToMessage("type.googleapis.com/movie.pbuf.Movie", json\_string);**

proto file: (message.proto)

syntax = "proto2";

package pb;

message Person

{

optional string name = 1;

}

main.cpp:

#include <iostream>

#include <memory>

// protobuf includes

#ifdef \_MSC\_VER

#pragma warning(push)

#pragma warning(disable: 4100 4127 4244 4267 4512)

#endif

#include "messages.pb.h"

#include <google/protobuf/util/json\_util.h>

#include <google/protobuf/util/type\_resolver\_util.h>

#ifdef \_MSC\_VER

#pragma warning(pop)

#endif

int main(int argc, char \*\*argv)

{

class pb::Person person;

std::string binary\_string;

for (char i = 0; i < 10; i++)

{

binary\_string.push\_back(i);

}

person.set\_name("Maria");

person.set\_binary(binary\_string.c\_str(), binary\_string.length());

std::string serialized = person.SerializeAsString();

std::string json\_string;

google::protobuf::util::JsonOptions options;

options.always\_print\_primitive\_fields = true;

std::shared\_ptr<google::protobuf::util::TypeResolver> resolver\_;

resolver\_.reset(google::protobuf::util::NewTypeResolverForDescriptorPool(

"", google::protobuf::DescriptorPool::generated\_pool()));

auto status = google::protobuf::util::BinaryToJsonString(resolver\_.get(), "/pb.Person", serialized, &json\_string, options);

std::cout << json\_string << std::endl;

std::string res;

std::cin >> res;

return(0);

}