

# Android

-

## komunikácia



Peter Borovanský

KAI, I-18

MS-Teams: [2sf3ph4](#), [List](#), [github](#)

borovan 'at' ii.fmph.uniba.sk

- **triedy z org.apache.\***
  - http(s) GET, POST, cookies
  - static Google Maps V2,
- **Retrofit** - GET/POST
- **formáty json (gson) a xml**
  - Google Directions

# MFF Keška 2019

48.151901 17.068422



AndreaS

- "I invented the term 'Object-Oriented', and I can tell you I did not have C++ in mind." - Alan Kay

TatianaG

- "Beware of bugs in the above code; I have only proved it correct, not tried it." - Donald E. Knuth.

PeterT

- Most software today is very much like an Egyptian pyramid with millions of bricks piled on top of each other, with no structural integrity, but just done by brute force and thousands of slaves." - Alan Kay

TomášM

- "Measuring programming progress by lines of code is like measuring aircraft building progress by weight." - Bill Gates





# MFF Keška 2020

## 48.151901 17.068422



Jozef B

- Co sa rado mava, aj sa rado hada...

Jakub T, Miriam C

- Vytiahli sme kešku s číslom BP 3/11 a bol na nej nahratý text
- "Kto druhému jamu kope, asi sám do nej spadne..."

Richard N

- Kon ma styri nohy a predsa sa aj tak potkne...

Lukáš V

- Ked nejde hora k Mohamedovi, ide Mohamed k hore...

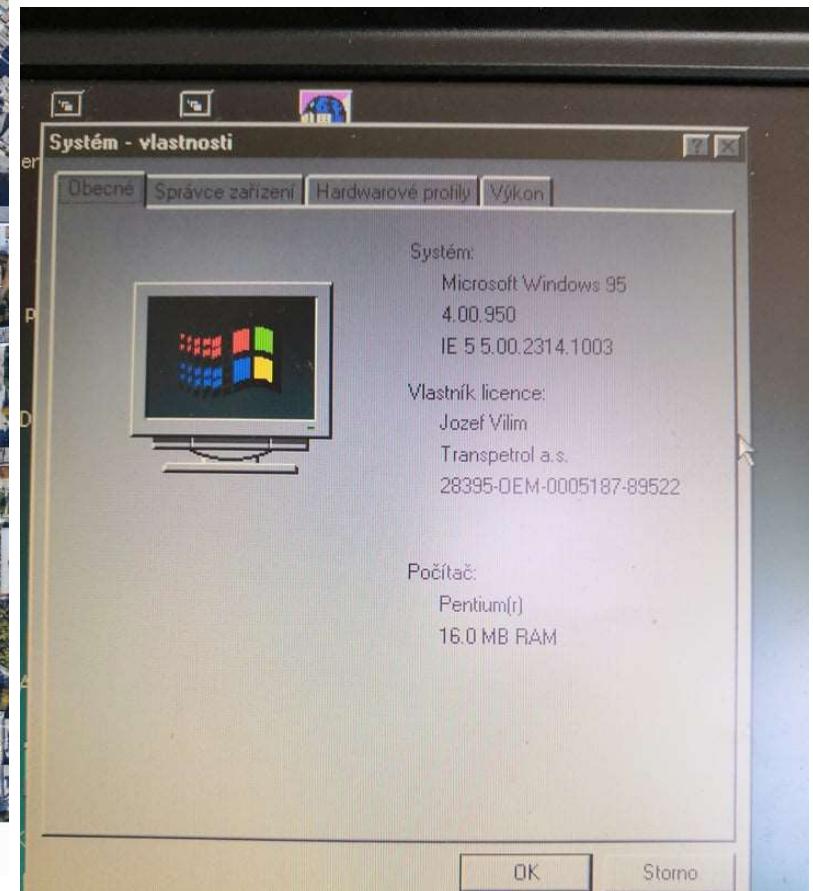
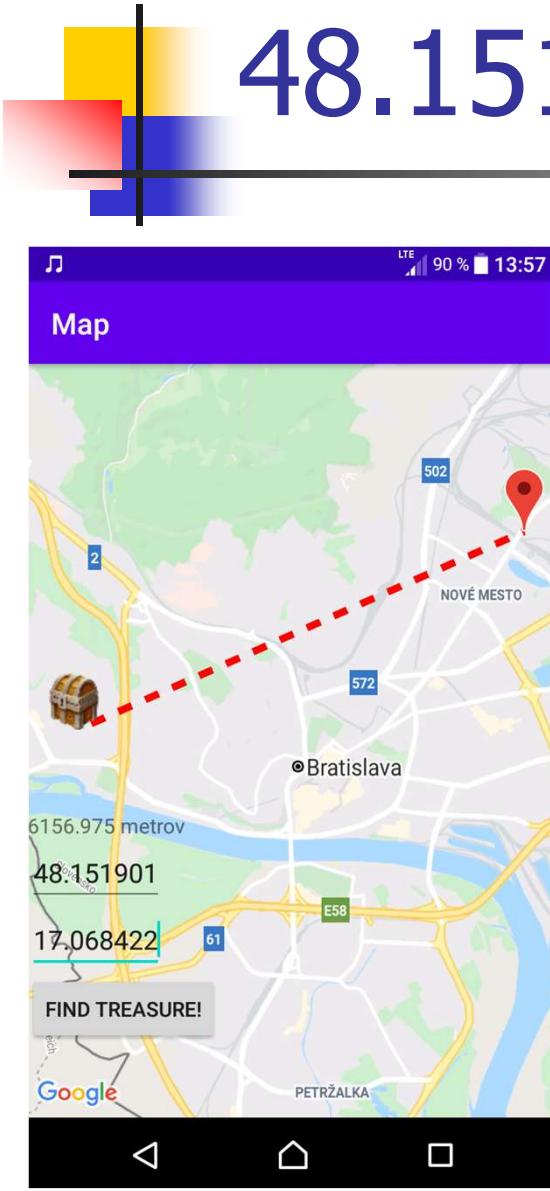
JožoB

- Co sa rado mava, aj sa rado hada...

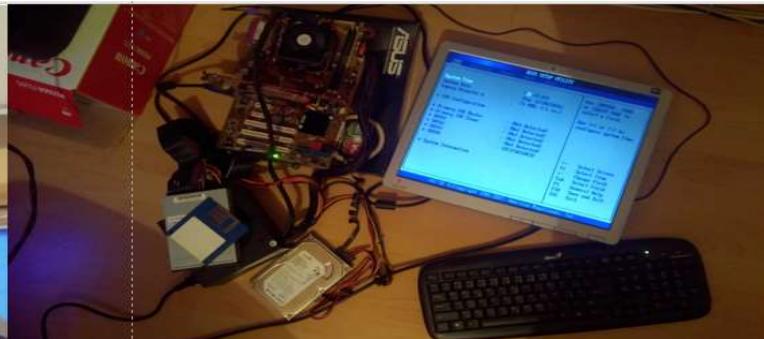


# MFF Keška 2020

## 48.151901 17.068422



# LukášG JurajH





# Aký klient...

dnes to bude viac o org.apache ako o androide

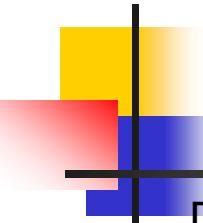
<http://hc.apache.org/httpcomponents-client-ga/tutorial/html/>

Klient koho, resp. kto je server ?

- server je len náš (ale nepoužívame http protokol na komunikáciu s ním):
  - môže to byť aj socket-socket komunikácia, ale vyvoláva to veľa otáznikov ...  
ako napr. bezpečnosť, robustnosť, multi-vlákno pre obsluhu viacerých klientov, ...
  - priamy prístup do databázy, napr. cez jdbc, iný komunikačný protokol
- server nie je náš, ale máme tam neadminovský účet (davinci.fmph.uniba):
  - najčastejšie provider poskytne rozhrania (okrem webservera/Apache) php, mysql, ...
  - najčastejšie, jediný otvorený port je http/https,
  - najčastejšie vznikne tzv. AMP riešenie (Apache-MySQL-Php/Perl/Python) – minulá dekáda
- server vôbec nie je náš
  - môžeme odtiaľ čítať,
  - resp. máme špeciálne API na prístup k dátam (Parse fy.Facebook, Firebase fy.Google)

Čo so ním chceme robiť (so serverom):

- download všeobecne prístupných, resp. zaheslovaných dát,
- upload (malé resp. veľké dáta)
- run/exec (RPC – Remote Procedure Call)



# Obsah prednášky

## Download:

- HTTP GET – primárne chceme dotiahnuť (veľké) dátu zo servera
  - malé dátu – *všetko jedno ...*
  - príklad: select/update mojej gps pozície (lat, long) z databázy na serveri
  - veľké dátu – potrebujeme *extra vlákno*, aby sa *nehryzla* apká,
    - *AsyncTask*
    - *corutina*
    - *Retrofit async-callback*
  - autorizácia (Basic Authorization)

Upload – primárne chceme poslať (veľké dátu na server)

- HTTP POST
  - veľké dátu (max. veľkosť HTTP GET requestu ~8kB, podľa http web-servera)  
problém, ak chceme uploadovať napr. snímka z kamery, video, ...

## Interpretované dátu:

- Bitmapa - Google Static Maps – príde statická bitmapa
- JSON – REST API, JSON parser (com.google.gson, alternatíva: org.json.JSONObject)
  - LocationApi.org príklad už bol minule
  - Google Directions – získanie cesty-navigácie od služby Google
- XML – SAXParser, alternatíva: org.w3c.dom

<http://dai.fmph.uniba.sk/courses/VMA/>  
[http://dai.fmph.uniba.sk/courses/VMA/ISLAND.JPG \(851kB\)](http://dai.fmph.uniba.sk/courses/VMA/ISLAND.JPG)  
[http://dai.fmph.uniba.sk/courses/VMA/ISLAND2.JPG \(5.5MB\)](http://dai.fmph.uniba.sk/courses/VMA/ISLAND2.JPG)  
[http://dai.fmph.uniba.sk/courses/VMA/android/03Http/KOZA.JPG \(1MB\)](http://dai.fmph.uniba.sk/courses/VMA/android/03Http/KOZA.JPG)  
[http://dai.fmph.uniba.sk/courses/VMA/android/03Http/KOZA2.JPG \(5MB\)](http://dai.fmph.uniba.sk/courses/VMA/android/03Http/KOZA2.JPG)  
<http://dai.fmph.uniba.sk/~borovan/rosnicka/>

# HttpClient - GET

Zobrazí obsah nejakej URI (Uniform Resource Identifier)

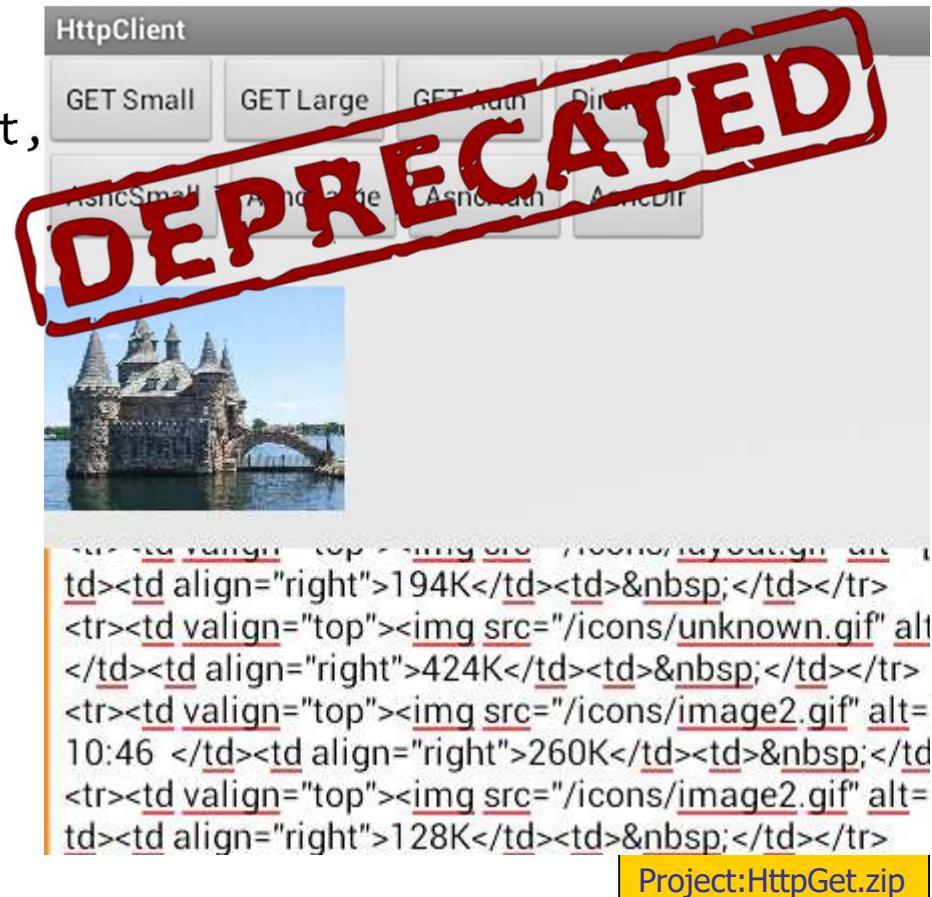
```
val uri= URI("http://dai.fmph.uniba.sk/courses/VMA/index.php")
```

Ilustrujte triedy:

- HttpClient, DefaultHttpClient,
- HttpGet
- HttpResponse
  
- prečíta obsah ako InputStream z HttpResponse
- zobrazí ako textový súbor a zobrazí v EditText

Problémy:

- ak je súbor veľký, "hryzne" sa hlavné vlákno aplikácie
- ak je zaheslovaný

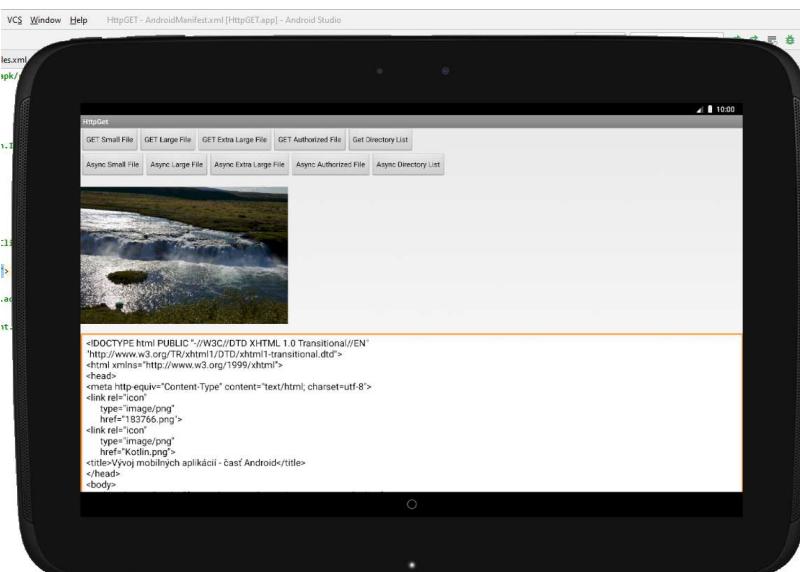


# Problém: API 29+

D/NetworkSecurityConfig:

No Network Security Config specified, using platform default

- na problém ma upozornili ErikK, AdamO,
- “*Works on my hardware*”, (Android 7, Emulátor 8.1)
- java.net.UnknownServiceException: CLEARTEXT communication to permitted by network security policy**
- ... aj našli riešenie (ľahké a funguje)



AndroidManifest.xml

```
<application  
    android:networkSecurityConfig="@xml/network_security_config"
```

res/xml/network\_security\_config.xml

```
<?xml version="1.0" encoding="utf-8"?>  
<network-security-config>  
    <base-config cleartextTrafficPermitted="true">  
        <trust-anchors>  
            <certificates src="system" />  
        </trust-anchors>  
    </base-config>  
</network-security-config>
```

Project:HttpGet.zip

**DEPRECATED**

# Problém: Apache HTTP client

## Apache HTTP client deprecation

With Android 6.0, we removed support for the Apache HTTP client. Beginning with Android 9, that library is removed from the bootclasspath and is not available to apps by default.

To continue using the Apache HTTP client, apps that target Android 9 and above can add the following to their `AndroidManifest.xml`:

```
<uses-library android:name="org.apache.http.legacy" android:required="false"/>
```

★ Note: The `android:required="false"` attribute is required for apps that have a minimum SDK of 23 or lower, because on devices with API levels lower than 24, the `org.apache.http.legacy` library is not available. (On those devices, the Apache HTTP classes are available on the bootclasspath.)

### AndroidManifest.xml

```
<application
    <activity
        ...
    </activity>
    <uses-library android:name="org.apache.http.legacy"
                  android:required="false"/>
</application>
```

```
build.gradle
android {
    useLibrary 'org.apache.http.legacy'
```

Project:HttpGet.zip

```
org.apache.http.HttpResponse  
org.apache.http.client.methods.HttpGet  
org.apache.http.impl.client.DefaultHttpClient
```

# HttpClient - GET

```
val httpget = HttpGet()  
httpget.uri = URI("http://dai.fmph.uniba.sk/courses/VMA/")  
val response = httpClient.execute(httpget)
```



```
val br = BufferedReader(InputStreamReader(  
                      response.entity.content))  
val result = StringBuilder() //!!! val result:String = ""  
while (true) {  
    val line = br.readLine() ?: break  
    result.append(line + "\n") //!!!result += line;  
}  
return result.toString()
```

Project:HttpGet.zip

```
org.apache.http.auth.AuthScope  
org.apache.http.auth.UsernamePasswordCredentials  
org.apache.http.client.CredentialsProvider  
org.apache.http.impl.client.BasicCredentialsProvider
```

# 3 Problémy



## Autorizácia // Basic Authorization, alternatíva:Digest (MD5)

```
val credentialProvider = BasicCredentialsProvider()  
credentialProvider.setCredentials(      // AuthScope("ip", 443)  
    AuthScope(AuthScope.ANY_HOST, AuthScope.ANY_PORT),  
    UsernamePasswordCredentials("java", "vaja")) //login,pass  
httpClient.credentialsProvider = credentialProvider
```

### ■ Interpretácia dát – obrázky

```
inputStream = httpResponse.entity.content
```

```
try {
```

```
    bitmap = BitmapFactory.decodeStream(inputStream, ..., options) }
```

- velké obrázky => BitmapFactory OutOfMemory
- riešenie BitmapFactory.Options.inSampleSize = 4

```
val options = BitmapFactory.Options()  
options.inSampleSize = 4  
options.inJustDecodeBounds = false
```

inSampleSize = 4 returns an image that is 1/4 the width/height of the original, and 1/16 the number of pixels

Project:HttpGet.zip

org.apache.http.auth.AuthScope  
org.apache.http.auth.UsernamePasswordCredentials  
org.apache.http.client.CredentialsProvider  
org.apache.http.impl.client.BasicCredentialsProvider

# 3 Problémy



- **Čakanie, dlhotrvajúce operácie**, ktoré blokujú hlavné vlákno
  - ak sa aktivita neozýva > 5sec (lebo pracuje), automaticky sa stopne, ...
  - riešenie (extra vlákno):
    - AsyncTask (deprecated),
    - Kotlin-corutiny
- **Aplikácia sama detektuje network operáciu v hlavnom vlákne**
  - a padne na chybe
- **Alebo jej to zakážete, ale problém ste nevyriešili**
  - nastavením thread policy (network oprácia však bude trvať rovnako dlho)

```
val policy = StrictMode.ThreadPolicy.Builder()  
    .detectAll()  
    .penaltyLog()  
    .build()  
StrictMode.setThreadPolicy(policy)
```

Project:HttpGet.zip

# AsyncTask

```
varargs params = pole vstupných parametrov  
progress = hodnota progresu, pre ProgressDialog  
result = typ výsledku
```

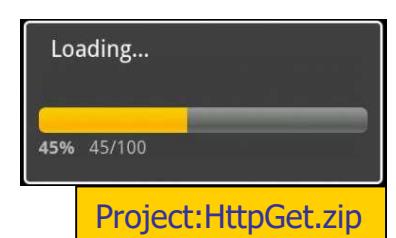
```
val ast = object : AsyncTask<String, Integer, String>() {  
    override fun doInBackground(varargs String params?):String {  
        var count = 0;  
        var result = StringBuilder()  
        try {  
            httpClient = DefaultHttpClient()  
            val uri = URI(params[0])      // moje http://....  
            val httpresponse = httpClient.execute(httpget) ←  
            val br = BufferedReader(InputStreamReader(  
                httpresponse.entity.content))  
            while (true) {  
                var line = br.readLine():break  
                result.append(line + "\n")  
                publishProgress(count++)  
            } catch (e : Exception) { ... }  
        return result.toString()  
    }
```

android.os.AsyncTask

AsyncTask<String, Integer, String>

je zložitá generická trieda

DEPRECATED



Project:HttpGet.zip

```
AsyncTask<String, Integer, String>
AsyncTask<String, Integer, Bitmap>
```

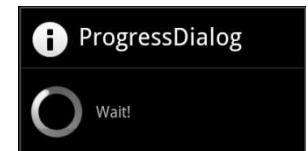
# AsyncTask

Bežia v GUI vlákne

```
override fun onPostExecute(result: String) {
    edttxt.append(result) // edttxt je nahradený parametrom kam
    pd.dismiss()
}

override fun onPreExecute() {
    pd = ProgressDialog(this@MyHttpClient)
    pd.setProgressStyle(ProgressDialog.STYLE_SPINNER)
    pd.max = 100
    pd.show()
}
override fun onProgressUpdate(vararg values: Int?) {
    pd.incrementProgressBy(values[0]?:0)
} _____ tu končí definícia AsyncTasku
ast.execute("http://dai.fmph.uniba.sk/courses/VMA/index.php")
```

DEPRECATED



v kóde najdete:

- **fun HttpGetAsyncString(uristr: String, kam: EditText)**  
implementuje AsyncTask<String, Integer, String>
- **fun HttpGetAsyncBitmap(uristr: String, kam: ImageView)**  
implementuje AsyncTask<String, Integer, Bitmap>

Project:HttpGet.zip

# varargs sú v Jave tri bodky ...

jazykové miniokienko

```
method(params : String[]) {}  
    sa volá takto  
method(arrayOf("first", "second"));
```



Kotlin:

```
method(vararg params : String) {} // variabilný počet Stringov
```

Java:

```
method(String... params) {}  
    sa volá takto  
method("first", "second");
```

ale je to len syntax sugar, verzia s varargs/... sa skompiluje ako pole []

```
override fun onProgressUpdate(vararg values: Int?) {  
    pd.incrementProgressBy(values[0]?:0)  
}
```

Project:HttpGet.zip



# Retrofit interface

Retrofit - type-safe HTTP client for Android, Java, Kotlin,...(Square)

REST-endpoint (obsahuje JSON converter GSON, XML, ...)   
GSON je knižnica na konverziu JSON na/z JAVA/Kotlin class

BASE URL <http://dai.fmph.uniba.sk>

```
interface APIService {  
    @GET("/users")  
    fun getUsers(@Query("per_page") int per_page,  
                @Query("page") int page)  
        Call<List<User>>  
  
    @GET("/users/{username}")  
    fun getUser(@Path("username") String username)  
        Call<User>  
}
```

# Retrofit builder



Retrofit - type-safe HTTP client for Android, Java, Kotlin,...(Square)

```
val httpClient = OkHttpClient.Builder()
```

OkHttpClient

- je implementácia HTTP & HTTP/2 klienta
- umožňuje nám pridávať interceptory, napr. na autentifikáciu

```
val retrofit = Retrofit.Builder()  
    .baseUrl("http://dai.fmph.uniba.sk")  
    .addConverterFactory(GsonConverterFactory.create())  
    .client(httpClient.build())  
    .build()
```

Retrofit builder integruje:

- komunikačný protokol, httpclient,
- konvertor, GSON medzi JSON a java-class, alebo XML

# Retrofit

## asynchronnost'



Retrofit - type-safe HTTP client for Android, Java, Kotlin,...(Square)

Retrofit builder integruje:

- komunikačný protokol, httpclient,
- konvertor, GSON medzi JSON a java-class, alebo XML

```
val service = retrofit.create(UserService.class::java)
```

```
val callAsync : Call<User> = service.getUser("borovan")
```

```
callAsync.enqueue(Callback<User>() {  
    override fun onResponse(call: Call<User>, response: Response<User>) {  
        user = response.body()  
    }  
    override fun onFailure(call: Call<User>, throwable: Throwable) {  
        println(throwable)  
    }  
})
```

<http://dai.fmph.uniba.sk/courses/VMA/>  
<http://dai.fmph.uniba.sk/courses/VMA/ISLAND.JPG>  
<http://dai.fmph.uniba.sk/courses/VMA/ISLAND2.JPG>  
<http://dai.fmph.uniba.sk/courses/VMA/android/03Http/KOZA.JPG>  
<http://dai.fmph.uniba.sk/~borovan/rosnicka/>



# Retrofit Download Image

```
interface RetrofitInterface {  
    @GET  
    fun getImage(@Url url: String): Call<ResponseBody?>  
}
```

```
val retrofit = Retrofit.Builder()  
    .baseUrl("http://dai.fmph.uniba.sk/courses/VMA/")  
    .build()  
val retrofitInterface = retrofit.create(  
    RetrofitInterface::class.java)  
val request = retrofitInterface.getImage("ISLAND2.JPG")  
try {  
    val body: ResponseBody? = request.execute().body()  
    downloadImage(body)  
} catch (e : IOException) {  
    e.printStackTrace()  
    Toast.makeText(applicationContext, e.message,  
        Toast.LENGTH_SHORT).show()  
}
```

Project:RetrofitDownloadImage.zip



# Retrofit Download Image

## Basic Authentication

```
class BasicAuthInterceptor(username:String, password:String): Interceptor {  
    private var credentials: String = Credentials.basic(username, password)  
  
    override fun intercept(chain: Interceptor.Chain): okhttp3.Response {  
        var request = chain.request()  
        request = request.newBuilder()  
            .header("Authorization", credentials).build()  
        return chain.proceed(request)  
    }  
}
```

```
val okclient = OkHttpClient.Builder()  
    .addInterceptor(BasicAuthInterceptor("java", "vaja")) ←  
    .build()  
  
val retrofit = Retrofit.Builder()  
    .baseUrl("http://dai.fmph.uniba.sk/courses/VMA/")  
    .client(okclient)  
    .build()  
  
val retrofitInterface = retrofit.create(RetrofitInterface::class.java)  
val request = retrofitInterface.getImage("ISLAND2.JPG")
```

Project:RetrofitDownloadImage.zip



# Retrofit Download Image

Save response body to file

```
private fun downloadImage(body: ResponseBody?) {
    var count = 0
    val data = ByteArray(1024 * 4)
    val fileSize = body?.contentLength()
    val inputStream = BufferedInputStream(body?.byteStream(), 1024 * 8)
    val outputFile = File(Environment
        .getExternalStoragePublicDirectory(Environment.DIRECTORY_DOWNLOADS),
        "downloaded.jpg")
    val outputStream = FileOutputStream(outputFile)
    var total = 0L
    var downloadComplete = false
    while (inputStream.read(data).also({ count = it }) != -1) {
        total += count.toLong()
        val progress = ((total*100).toDouble()/(fileSize?:100).toDouble()).toInt()
        updateNotification(progress)
        outputStream.write(data, 0, count)
        downloadComplete = true
    }
    onDownloadComplete(downloadComplete)
    outputStream.flush()
    outputStream.close()
    inputStream.close()
}
```

```
Android-10
<application
    android:requestLegacyExternalStorage="true"
```

Project:RetrofitDownloadImage.zip



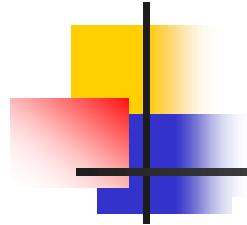
# Retrofit Download Image

Load image from downloaded file

```
val file = File(Environment.getExternalStoragePublicDirectory(  
    Environment.DIRECTORY_DOWNLOADS)  
    .getPath() + File.separator.toString() +  
    "downloaded.jpg")  
Picasso.get().load(file).into(imgView)
```

```
Android-10  
<application  
    android:requestLegacyExternalStorage="true"
```

Project:RetrofitDownloadImage.zip



# Static Google Maps

(život pred Google Maps API...)



Google Maps poskytujú API pre download statickej mapy, príklad.

<http://maps.googleapis.com//maps/api/staticmap?center=48.152177,17.07153&zoom=15&size=500x500&maptype=mobile&markers=48.152177,17.07153,ref/&key=AIzaSyCUUeYuihYGL8&sensor=false&>

Povinné polia v HTTP Requeste:

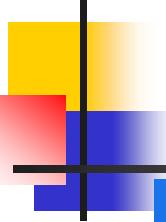
- center – stred mapy,  
napr. "48.152177,17.07153", resp. "Mlynska dolina, Bratislava"
- zoom – 0..21
- size – veľkosť obrázku
- formát – PNG (default), jpg, gif, ...
- key (25kAccess/day ☺) si vygenerujete tu: <https://console.developers.google.com/apis/>  
<https://developers.google.com/maps/documentation/staticmaps/>  
<https://developers.google.com/maps/documentation/maps-static/get-api-key>

iné, nepovinné polia:

- markers,
- path, ...

Project: StaticMaps.zip

# Google Platform Console



Google Cloud Platform My Project

Google Maps APIs

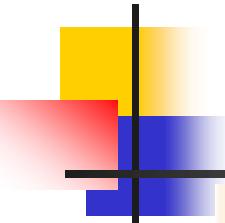
Overview APIs Metrics Support

**In use APIs**

Select an API to view details. Figures are for the last 30 days.

API ↑	Requests	Errors
Directions API	4	3
Distance Matrix API	0	0
Maps Embed API	0	0
Maps JavaScript API	0	0
Maps SDK for Android	0	0
Maps SDK for iOS	0	0
Maps Static API	139	127
Roads API	0	0

**Reminder:** To use the Maps Static API, you must enable billing. You can enable billing when you get your API key (see the [Quick guide](#)) or as a separate process (see [Usage and Billing](#)).



# Disclaimer statement

(tento kurz neberie žiadnu zodpovednosť za straty na vašich účtoch)



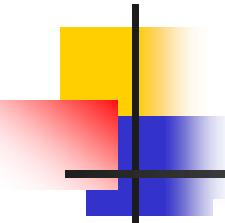
Requests per Day (QPD) limits have ended, effective June 11, 2018

**For Existing Customers:**

- If you are an existing customer (using the Google Maps Platform before June 11, 2018), you were billed under the previous plan until July 16, 2018.
- **Note:** Starting on June 11, 2018, Google began to roll out the removal of the default QPD limits on existing billing accounts. This process may take up to six weeks.
- If you rely on the default QPD limits to help you manage your cost of use, we recommend you [set your own daily limits](#) in the Google Cloud Platform Console, to override the default QPD limits set by Google. This will ensure that your preferred QPD limits remain in place.

**For New Customers:**

- If you are a new customer (using the Google Maps Platform starting on or after June 11, 2018), you were billed under the previous plan until July 16, 2018.
- **Note:** New customers began receiving unlimited QPD starting on June 11, 2018.
- To help you manage your cost of use, you can [set your own QPD limits](#) in the Google Cloud Platform Console.



# Pricing



Maps Static API

Google

OVERVIEW

PRICING

DOCUMENTATION

SUPPORT

Category: [Maps](#)

Service name: static-maps-backend.go

## About Google

Google's mission is to organize the world's information and make it universally accessible and useful. Through products and platforms like Search, Maps, Gmail, Android, Google Play, Chrome and YouTube, Google plays a meaningful role in the daily lives of billions of people.

## Pricing

Flexible pricing that scale to fit your needs. Plus, get [USD200.00](#) in free usage for Maps, Routes, and Places every month.

Static Maps [?](#)

TIER 1

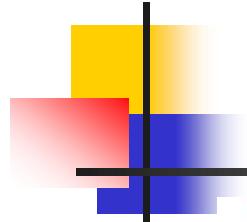
**USD2.00**

/1K requests

TIER 2

**USD1.60**

/1K requests



Google™

## Invoice

Invoice number: 3833153931

Google Commerce Limited

Gordon House

Barrow Street

Dublin 4

Ireland

VAT number: SK4020458739

Bill to

Peter Borovansky

Invoice number .....	31
Invoice date .....	20
Billing ID .....	-5589
Account ID .....	IFCE4-B5792B

Google Cloud

Total in EUR	€0.00
--------------	-------

Summary for Nov 1, 2020 - Nov 30, 2020

Subtotal in EUR	€0.00
VAT (20%)	€0.00
Total in EUR	€0.00

# Static Maps API v2

(V2 Upgrade Guide)

## Tiež prišlo **API V2 Static Maps**

- <https://developers.google.com/maps/documentation/staticmaps/upgrade>

Static map creator:

- vygeneruje request <http://staticmapmaker.com/>, len si pridajte API Key:  
[http://maps.google.com/maps/api/staticmap?  
center=48.160020,17.075810&zoom=13&  
markers=mlynska+dolina, bratislava&size=400x400&  
sensor=TRUE OR FALSE](http://maps.google.com/maps/api/staticmap?center=48.160020,17.075810&zoom=13&markers=mlynska+dolina, bratislava&size=400x400&sensor=TRUE OR FALSE)



# Ako vygenerovať httpRequest s parametrami

```
org.apache.http.NameValuePair;
org.apache.http.message.BasicNameValuePair;

val httpClient = DefaultHttpClient()
var url = staticGoogleMap
if (!url.endsWith("?")) url += "?"
val httpParams = mutableListOf<NameValuePair>()
val latlngString = "${loc.latitude}, ${loc.longitude}"
httpParams.add(BasicNameValuePair("center", latlngString))
httpParams.add(BasicNameValuePair("zoom", "15"))
httpParams.add(BasicNameValuePair("size", "480x480"))
httpParams.add(BasicNameValuePair("markers", latlngString))
httpParams.add(BasicNameValuePair("key", "AIzaS*****YGL8"))
url += URLEncodedUtils.format(httpParams, "utf-8")
val httpget = HttpGet()
httpget.uri = URI(url)
val httpResponse = httpClient.execute(httpget)
val inputStream = httpResponse.entity.content
result = BitmapFactory.decodeStream(inputStream)
```

**DEPRECATED**

urlEncoding: [http://www.w3schools.com/tags/ref\\_urlencode.asp](http://www.w3schools.com/tags/ref_urlencode.asp)

Project:StaticMaps.zip



# Retrofit request s parametrami

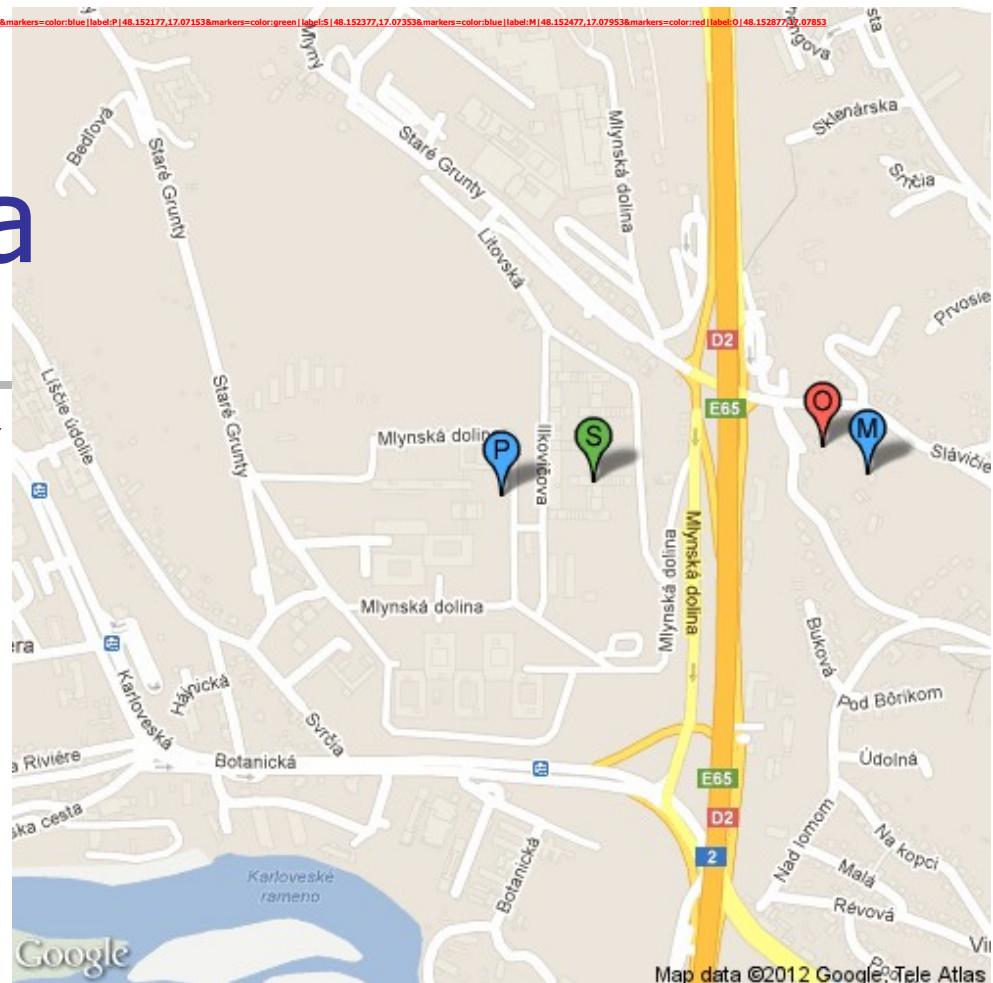
```
@GET  
fun getStaticMap(@Url url: String,  
                 @Query("center")  lslng : String,  
                 @Query("zoom")    zoom : String,  
                 @Query("size")    size : String,  
                 @Query("markers") markers : String,  
                 @Query("key")     key : String): Call<ResponseBody?>
```

```
val retrofit = Retrofit.Builder()  
    .baseUrl("http://maps.google.com/")  
  
    ...  
val request = retrofitInterface.getStaticMap(  
    "maps/api/staticmap",  
    "48.160020,17.075810",  
    "13",  
    "480x480",  
    "48.160020,17.075810",  
    "AIzaS*****EeYuihYGL8"  
)
```

# Domáca úloha

(kamaráti na mape)

- napíšte aplikáciu, ktorá po zapnutí uploaduje súradnice užívateľa (lat,long) na server pomocou priloženého php scriptu.
- tabuľka obsahuje ID, DateTime, Latitude, Longitude,
- zobrazte posledné záznamy všetkých aktívnych (max.60 min. starý záznam) užívateľov na statickej mape farebne odlišených markermi s iniciálkou
- GMAPs alebo Static GMAPs



[maps.googleapis.com/maps/api/staticmap?center=48.152177,17.07153&zoom=15&size=500x500&maptype=mobile&key=AIzaSyC3yT-5chHoknBR0kR-F7xmuPjzTakcJE&sensor=false&markers=color:blue|label:P|48.152177,17.07153&markers=color:green|label:S|48.152377,17.07353&markers=color:blue|label:M|48.152477,17.07953&markers=color:red|label:O|48.152877,17.07853](http://maps.googleapis.com/maps/api/staticmap?center=48.152177,17.07153&zoom=15&size=500x500&maptype=mobile&key=AIzaSyC3yT-5chHoknBR0kR-F7xmuPjzTakcJE&sensor=false&markers=color:blue|label:P|48.152177,17.07153&markers=color:green|label:S|48.152377,17.07353&markers=color:blue|label:M|48.152477,17.07953&markers=color:red|label:O|48.152877,17.07853)

# Potrebné detaily k DÚ

<http://dai.fmph.uniba.sk/courses/VMA/android/php/PositionUpdate.php?name=Pete&lat=48.152177&long=17.07153>

```
<?php
$dbhost = 'kempelen.ii.fmph.uniba.sk';
$dbuser = 'androids';
$dbpass = 'HotelDiamantDunakility';
$conn = mysql_connect($dbhost, $dbuser, $dbpass);
if(! $conn )
    die('Could not connect: ' . mysql_error());
$sql = "insert into myfriends (name, time, lati, longi) values ('".
    $_GET['name'].'', NOW(),".$_GET['lat'].",".$_GET['long'].")";
mysql_select_db('androids');
$retval = mysql_query( $sql, $conn );
if(! $retval )
    die('Could not update data: ' . mysql_error());
echo "Insert successfull\n";
mysql_close($conn);
?>
```

<http://kempelen.ii.fmph.uniba.sk/phpmyadmin/>

# Potrebné detaily k DÚ

<http://dai.fmph.uniba.sk/courses/VMA/android/php/PositionSelect.php>

```
<?php  
...  
$sql = "select * from myfriends where time > DATE_SUB( now() ,  
           INTERVAL 1 DAY);";  
mysql_select_db('androids');  
$retval = mysql_query( $sql, $conn );  
if(! $retval ) {  
    die('Could not select data: ' . mysql_error());  
}  
$rows = array();  
while($r = mysql_fetch_assoc($retval)) {  
    $rows[] = $r;  
}  
print json_encode($rows);  
mysql_close($conn);  
?>
```

<http://kempelen.ii.fmph.uniba.sk/phpmyadmin/>



```
[  
  {"name": "Peter", "time": "2012-11-28  
15:41:50", "lati": "48.3443", "longi": "17.2322"},  
  {"name": "Peter", "time": "2012-11-28  
19:27:20", "lati": "48.1522", "longi": "17.0715"},  
  {"name": "Silvia", "time": "2012-11-28  
19:27:42", "lati": "48.1522", "longi": "17.0415"},  
  {"name": "Peter", "time": "2012-11-28  
19:28:44", "lati": "48.1522", "longi": "17.0715"}]
```

# HttpClient – POST

Encode  
Decode  
Base64

Ak potrebujem uploadovať väčšie dáta (napríklad fotku), použijeme POST

- vytvoríme aplikáciu, ktorá zosníma obrázok z kamery,
- zobrazí na display, kde ju môžeme pomenovať,
- pomocou HTTP-POST pošleme na server
- tam ju pomocou malého php-scriptu ukladáme do „galéria“ (adresára),

```
<?php
$base=$_REQUEST['image'];
$iname=$_REQUEST['iname'];
if ($iname != "") {
    $binary=base64_decode($base); // dekóduje z MIME base64
    $file = fopen($iname.'.jpg', 'wb');
    fwrite($file, $binary);
    fclose($file);
    echo "... OK ...";
} else
    echo "... NULL NAME ...";
?>
```

<http://dai.fmph.uniba.sk/courses/VMA/galeria/>

download 

[ShotUploaderPhp=  
http://dai.fmph.uniba.sk/courses/VMA/upload.php](http://dai.fmph.uniba.sk/courses/VMA/upload.php)

Project:HttpPOST.zip

# HttpClient - POST

Encode  
Decode  
Base64

HttpPostAsyncBitmap vytvorí AsyncTask<String, Integer, Boolean>

```
val baos = ByteArrayOutputStream()  
// Bitmap.CompressFormat.PNG, Bitmap.CompressFormat.WEBP, 90%  
→ bmp.compress(Bitmap.CompressFormat.JPEG, 90, baos) // fotka je v baos  
val img_string = Base64.encodeBytes(baos.toByteArray())  
val variableValue = mutableListOf<NameValuePair>()  
variableValue.add(BasicNameValuePair("image", img_string)) //fotka  
variableValue.add(BasicNameValuePair("iname", params[0])) // meno  
try {  
    val httpclient = DefaultHttpClient()  
    val httppost = HttpPost("http://dai.fmph.uniba.sk/courses/VMA/galeria/upload.php")  
    httppost.entity = UrlEncodedFormEntity(variableValue)  
    val response = httpclient.execute(httppost)  
    Log.d("HttpClient", response.statusLine().toString())  
    . . .
```



upload

DEPRECATED

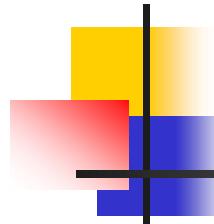
Project:HttpPOST.zip

# Ako odfotit'



```
override fun onClick(v: View) {
    if (v.id == R.id.btnShot)
        startActivityForResult(
            Intent(MediaStore.ACTION_IMAGE_CAPTURE),
            IMAGE_CAPTURE_RESULT)
    if (v.id == R.id.btnUpload)
        HttpPostAsyncBitmap(edtPictureName.text.toString(), bmp)
-----
override fun onActivityResult(requestCode:Int, resultCode:Int,
                                data: Intent) {
    // fotka prída ako časť intentu data
    if (requestCode==IMAGE_CAPTURE_RESULT && resultCode==RESULT_OK){
        val bundle = data.extras // extrahovanie bitmapy
        bmp = bundle["data"] as Bitmap
        val iview = findViewById(R.id.imageView) as ImageView
        iview.setImageBitmap(bmp) // zobrazenie bitmapy
    }
}
```

Project:HttpPOST.zip



# Prémia

(len krátko-trvajúca)

Napíšte malú jednoúčelovú androidovskú aplikáciu,  
hoc aj bez GUI,..., ale ktorá z adresára

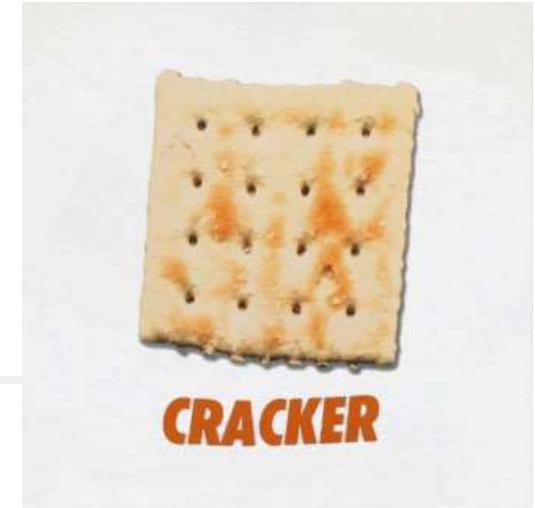
<http://dai.fmph.uniba.sk/courses/VMA/android/crackme/>

- zmažte súbor crack.me
- vytvorí tam súbor VašeMeno.VašePriezvisko (kvôli bodom za prémiu),  
ktorý ale obsahuje vašu hackerskú prezývku, teda žiadnen php-kód...
- všetky stopy (napr. pomocné súbory) po sebe uprakte.

Poznámka: zvyšok webu prosím nechajte tak ☺

Deadline: čím skôr...

bodovanie 3 body (zlato), 2 (striebro), 1.5 (bronz), 1 (za účasť')



Project:Crackmi.zip

# Break ----- Retro POST

- <https://medium.com/@normanaspx/android-retrofit-post-5c84d62ec24d>
- <https://medium.com/@normanaspx/android-retrofit-post-5c84d62ec24d>



# Cookies

Pokúsime sa prihlásiť do LISTu a zapamätať si vytvorenie cookies, obsahujúce session-id.  
Otvoríme si Developer Tools (FF) a pozeráme, čo sa deje, aké requesty-responsy chodia...

Nový studentský účet

- Zobrazit / skryť formulár

Celé meno: *	Mobilný matfyzák
E-mail: *	mobilny@matfyzak.sk
Heslo: *	androidovy

## Vitajte v LIST-e

E-mailová adresa študenta:	mobilny@matfyzak.sk
Heslo študenta:	*****

Navigácia

- Zapisť sa na kurz
- Zmeniť svoju skupinu
- Úlohy a hodnotenie**
- Tabuľka bodov

Úlohy a hodnotenie

Ste úspešne prihlásený(á) do systému.

Nie je zvolený žiadny aktívny kurz

Headers	Cookies	Params	Response	Timings
Filter cookies				
<input checked="" type="checkbox"/> list_session: "a:4:{s:10:"session_id";s:32:"...73023ba47002f549f31580de54f28"				
<input checked="" type="checkbox"/> list_session: "a:4:{s:10:"session_id";s:32:"...73023ba47002f549f31580de54f28"				
<input checked="" type="checkbox"/> list_session: "a:4:{s:10:"session_id";s:32:"...73023ba47002f549f31580de54f28"				
Request cookies				
__utma: "108429333,423508135,1348676542,1382609048,1384800468,58"				
__utma: "265282635,1670959162,1352189289,1352194877,1371904886,3"				
__utmc: "108429333"				
__utmz: "108429333,1380036106,54,8,utmcs...=organic utmctr=(not provided)"				
__utmz: "265282635,1371904886,3,1,utmcsr...d=referral utmccn=/kontakt.html"				
list_session: "a:4:{s:10:"session_id";s:32:"...73023ba47002f549f31580de54f28"				

Method File Domain

- POST aHR0cDovL2NhcGVrLmlpLm2tcGgu... capek.ii.fmph.uniba.sk
- GET / capek.ii.fmph.uniba.sk
- GET jquery.fancybox.pack.js?v=2.1.4 capek.ii.fmph.uniba.sk
- GET jquery.fancybox.css?v=2.1.4 capek.ii.fmph.uniba.sk
- GET jquery.fancybox-buttons.css?v=2... capek.ii.fmph.uniba.sk
- GET jquery.fancybox-buttons.js?v=2.... capek.ii.fmph.uniba.sk

# Advent of Code

<https://adventofcode.com/>

- Json URI
- <https://adventofcode.com/2019/leaderboard/private/view/229344.json>
- Ale chce to vypátrat' cookies od servera

Name	Value	Domain
_gid	GA1.2. 4.1575 [REDACTED]	adventofcode.co m
_ga	GA1.2. .15727 [REDACTED]	adventofcode.co m
session	53616c7465645f5 f9d4157c5dfffe18 1 [REDACTED] ad b [REDACTED] 8 d [REDACTED] 7f4 7e1156110085d70 b869676	adventofcode.co m

▼ 229344:

local\_score: 325  
id: "229344"  
stars: 8  
completion\_day\_level:  
  ▼ 1:  
    ▼ 1:  
      get\_star\_ts: "1575176691"  
    ▼ 2:  
      get\_star\_ts: "1575177020"  
      ▼ 2:  
        get\_star\_ts: "1575264081"  
        get\_star\_ts: "1575264839"  
      ▼ 3:  
        ▼ 1:  
          get\_star\_ts: "1575350735"  
        ▼ 2:  
          get\_star\_ts: "1575351770"  
      ▼ 4:  
        ▼ 1:  
          get\_star\_ts: "1575436158"  
        ▼ 2:  
          get\_star\_ts: "1575440183"  
global\_score: 0  
last\_star\_ts: "1575440183"  
name: "Peter BOROVANSKY"

# Advent of Code

<https://adventofcode.com/>

## ■ Softvér tretích strán - Postman

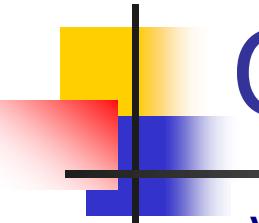


A screenshot of the Postman application interface. The top bar shows a GET request to <https://adventofcode.com/2019/leaderboard/private/view/229344.json>. The 'Params' tab is selected, showing a single entry: 'Key' with 'Value'. Below the table are tabs for 'Body', 'Cookies (3)', 'Headers (7)', and 'Test Results'. The status bar at the bottom right indicates a successful response: Status: 200 OK, Time: 690ms, Size: 10.54 KB.

KEY	VALUE	DESCRIPTION
Key	Value	Description

**Cookies (3)**

Name	Value	Domain	Path	Expires	HttpOnly	Secure
_gid	GA1.2.4.1575	adventofcode.com	/	Never	false	false
_ga	GA1.2..15727	adventofcode.com	/	Never	false	false
session	53616c7465645f5f9d..1812a..cadb2e..28de1..7f47c4430f4f085a70b869676	adventofcode.com	/	Never	false	false



# Cookie store

```
val cookieStore = BasicCookieStore()
val bc1 = BasicClientCookie("_gid", "GA1.2.126736...")
bc1.domain = "adventofcode.com"
bc1.path = "/"      bc1.isSecure = false
cookieStore.addCookie(bc1)

val bc2 = BasicClientCookie("_ga", "GA1.2.11794137...")
bc2.domain = "adventofcode.com"
bc2.path = "/"      bc2.isSecure = false
cookieStore.addCookie(bc2)

val bc3 = BasicClientCookie("session", "53616c7...")
bc3.domain = "adventofcode.com"
bc3.path = "/"      bc3.isSecure = false
cookieStore.addCookie(bc3)

val ctx = BasicHttpContext()
ctx.setAttribute(ClientContext.COOKIE_STORE, cookieStore)

val httpResponse = httpClient.execute(httpget, ctx)
```

Project:HttpAoC.zip

# JSON

(JSON je vrah čitatel'nejší ako xml)

{ "owner\_id": "229344", "members": [ "633850", { "stars": 2, "id": "633850", "completion\_day\_level": { "1": { "1": { "get\_star\_ts": "1575177325" }, "2": { "get\_star\_ts": "1575232476" } }, "local\_score": 64, "name": null, "last\_star\_ts": "1575232476", "global\_score": 0 }, "50608": { "id": "50608", "completion\_day\_level": { "2": { "1": { "get\_star\_ts": "1575323440" }, "2": { "get\_star\_ts": "1575324582" } }, "1": { "2": { "get\_star\_ts": "1575234400" }, "1": { "get\_star\_ts": "1575233779" } }, "local\_score": 99, "name": null, "last\_star\_ts": "1575324582", "global\_score": 0, "stars": 4 }, "387689": { "completion\_day\_level": {}, "id": "387689", "last\_star\_ts": 0, "global\_score": 0, "name": "beowulf11", "local\_score": 0, "stars": 0 }, "284148": { "local\_score": 116, "name": "Jana Lachova", "last\_star\_ts": "157517389", "global\_score": 0, "stars": 4 }, "284148": { "completion\_day\_level": { "1": { "1": { "get\_star\_ts": "1575190775" }, "2": { "get\_star\_ts": "1575191559" } }, "2": { "1": { "get\_star\_ts": "1575316384" }, "2": { "get\_star\_ts": "1575317389" } }, "stars": 4 }, "289473": { "stars": 0, "completion\_day\_level": {}, "id": "289473", "last\_star\_ts": 0, "name": "Michal Vido", "local\_score": 0 }, "381890": { "completion\_day\_level": {}, "id": "381890", "last\_star\_ts": 0, "name": "Zoltan Kluka", "global\_score": 0, "local\_score": 0, "stars": 0 }, "133669": { "stars": 6, "local\_score": 160, "last\_star\_ts": "1575400260", "global\_score": 0, "name": "Zoltan Onody", "id": "133669", "completion\_day\_level": { "2": { "2": { "get\_star\_ts": "1575333802" }, "1": { "get\_star\_ts": "1575331323" } }, "1": { "1": { "get\_star\_ts": "1575202384" }, "2": { "get\_star\_ts": "1575204753" } }, "3": { "1": { "get\_star\_ts": "1575394347" }, "2": { "get\_star\_ts": "1575400260" } } }, "440747": { "name": "Ivan Martynovskyl", "last\_star\_ts": "1575401728", "global\_score": 0, "local\_score": 177, "completion\_day\_level": { "2": { "2": { "get\_star\_ts": "1575310108" }, "1": { "get\_star\_ts": "1575308668" } }, "1": { "2": { "get\_star\_ts": "1575188229" }, "1": { "get\_star\_ts": "1575187596" } }, "3": { "2": { "get\_star\_ts": "1575401728" }, "1": { "get\_star\_ts": "1575400101" } }, "id": "440747", "stars": 6 }, "374071": { "stars": 6, "id": "374071", "completion\_day\_level": { "3": { "2": { "get\_star\_ts": "15753767678" } }, "1": { "get\_star\_ts": "1575375301" } }, "1": { "2": { "get\_star\_ts": "1575185453" } }, "1": { "get\_star\_ts": "1575184775" }, "2": { "2": { "get\_star\_ts": "1575271779" } }, "1": { "get\_star\_ts": "1575271476" }, "local\_score": 202, "global\_score": 0, "last\_star\_ts": "1575376678", "name": "Gyorgy Tomcsanyi", "id": "635829", "completion\_day\_level": { "3": { "1": { "get\_star\_ts": "1575350543" } }, "2": { "get\_star\_ts": "1575350619" } }, "1": { "2": { "get\_star\_ts": "1575176838" } }, "1": { "get\_star\_ts": "1575176482" }, "2": { "2": { "get\_star\_ts": "1575265176" }, "1": { "get\_star\_ts": "1575265098" } }, "id": "635829", "name": "Jozef Kubik", "last\_star\_ts": "1575350619", "global\_score": 15, "local\_score": 246, "stars": 6 }, "425420": { "stars": 6, "name": "Peter Gergej", "last\_star\_ts": "1575359599", "global\_score": 0, "local\_score": 210, "completion\_day\_level": { "3": { "2": { "get\_star\_ts": "1575359599" } }, "1": { "1": { "get\_star\_ts": "1575274108" } }, "2": { "get\_star\_ts": "1575274499" } }, "1": { "1": { "get\_star\_ts": "1575184467" }, "2": { "2": { "get\_star\_ts": "1575184888" } }, "id": "425420", "stars": 0, "last\_star\_ts": 0, "name": "OHUSAR", "global\_score": 0, "local\_score": 0, "completion\_day\_level": {}, "id": "432600", "last\_star\_ts": 0, "name": "Maroš Malý", "global\_score": 0, "local\_score": 0, "stars": 0 }, "205219": { "id": "205219", "completion\_day\_level": {}, "local\_score": 0, "last\_star\_ts": 0, "global\_score": 0, "name": "yoyko", "stars": 0 }, "413457": { "stars": 0, "name": "kkikus", "last\_star\_ts": 0, "global\_score": 0, "local\_score": 0, "completion\_day\_level": {}, "id": "413457", "last\_star\_ts": "1575379220", "name": "F Kerak", "global\_score": 0, "local\_score": 143, "completion\_day\_level": { "2": { "2": { "get\_star\_ts": "1575376842" } }, "1": { "get\_star\_ts": "1575376511" } }, "1": { "2": { "get\_star\_ts": "1575375755" }, "1": { "get\_star\_ts": "1575375433" } }, "3": { "1": { "get\_star\_ts": "1575378173" } }, "2": { "get\_star\_ts": "1575379220" }, "id": "387446", "stars": 0 }, "382973": { "completion\_day\_level": {}, "id": "382973", "last\_star\_ts": 0, "name": "Boris Silný", "global\_score": 0, "local\_score": 0, "stars": 0 }, "633816": { "stars": 6, "completion\_day\_level": { "3": { "2": { "get\_star\_ts": "1575355116" } }, "1": { "get\_star\_ts": "1575353986" } }, "2": { "1": { "get\_star\_ts": "1575264967" }, "2": { "get\_star\_ts": "1575265404" } }, "1": { "2": { "get\_star\_ts": "1575177501" } }, "1": { "get\_star\_ts": "1575177229" }, "id": "633816", "last\_star\_ts": "1575355116", "global\_score": 0, "name": null, "local\_score": 230 }, "357988": { "last\_star\_ts": "1575384913", "name": null, "global\_score": 0, "local\_score": 181, "completion\_day\_level": { "1": { "2": { "get\_star\_ts": "1575191610" } }, "2": { "2": { "get\_star\_ts": "1575280368" } }, "1": { "get\_star\_ts": "1575279807" } }, "3": { "2": { "get\_star\_ts": "1575384913" } }, "1": { "get\_star\_ts": "1575384157" }, "id": "357988", "stars": 6 }, "442294": { "stars": 0, "local\_score": 0, "last\_star\_ts": 0, "name": "mrkvrek", "global\_score": 0, "id": "442294", "completion\_day\_level": {}, "id": "727545", "stars": 0 }, "275369": { "completion\_day\_level": {}, "id": "727545", "last\_star\_ts": 0, "global\_score": 0, "name": null, "local\_score": 0, "stars": 0 }, "406017": { "id": "406017", "completion\_day\_level": { "2": { "1": { "get\_star\_ts": "1575264301" } }, "2": { "2": { "get\_star\_ts": "1575176687" } }, "1": { "1": { "get\_star\_ts": "1575177031" } }, "3": { "2": { "get\_star\_ts": "1575350314" } }, "1": { "get\_star\_ts": "1575349931" } }, "local\_score": 249, "global\_score": 0, "last\_star\_ts": "1575350314", "name": "Erik Szalay", "stars": 6 }, "633833": { "stars": 0, "id": "633833", "completion\_day\_level": {}, "local\_score": 0, "global\_score": 0, "last\_star\_ts": 0, "name": null }, "49729": { "stars": 0, "name": "Seluwin", "last\_star\_ts": 0, "global\_score": 0, "local\_score": 0 }, "452327": { "stars": 4, "local\_score": 89, "global\_score": 0, "last\_star\_ts": "1575359071", "name": "Andrea Spišáková", "id": "452327", "completion\_day\_level": { "2": { "1": { "get\_star\_ts": "1575318972" } }, "2": { "get\_star\_ts": "1575350104" } }, "1": { "2": { "get\_star\_ts": "1575316042" } }, "1": { "get\_star\_ts": "1575315442" }, "id": "382474", "completion\_day\_level": { "3": { "1": { "get\_star\_ts": "1575367655" } }, "2": { "get\_star\_ts": "1575380238" } }, "1": { "1": { "get\_star\_ts": "1575367655" } }, "2": { "get\_star\_ts": "1575297391" }, "1": { "2": { "get\_star\_ts": "1575294741" } }, "1": { "get\_star\_ts": "1575290505" }, "id": "382474", "last\_star\_ts": "1575380238", "global\_score": 0, "name": "Linda Jurkasova", "local\_score": 170, "stars": 6 }, "230440": { "local\_score": 196, "last\_star\_ts": "1575355079", "name": "Tomáš Bočinec", "global\_score": 0, "id": "230440", "completion\_day\_level": { "3": { "2": { "get\_star\_ts": "1575355063" } }, "1": { "1": { "get\_star\_ts": "1575355079" } }, "2": { "get\_star\_ts": "1575222029" } }, "2": { "2": { "get\_star\_ts": "1575222656" } }, "1": { "1": { "get\_star\_ts": "1575222029" } }, "2": { "get\_star\_ts": "1575222656" }, "id": "257369", "completion\_day\_level": { "1": { "2": { "get\_star\_ts": "1575375683" } }, "2": { "get\_star\_ts": "1575272186" } }, "1": { "1": { "get\_star\_ts": "1575272186" } }, "2": { "get\_star\_ts": "1575279106" }, "1": { "1": { "get\_star\_ts": "1575184259" } }, "2": { "get\_star\_ts": "1575184876" }, "3": { "1": { "get\_star\_ts": "1575369676" } }, "2": { "get\_star\_ts": "1575375683" }, "local\_score": 207, "last\_star\_ts": "1575375683", "global\_score": 0, "name": null, "stars": 6 }, "289854": { "last\_star\_ts": "1575375697", "name": "Miki Hermann", "local\_score": 206, "completion\_day\_level": { "2": { "2": { "get\_star\_ts": "1575267648" } }, "1": { "get\_star\_ts": "1575265752" } }, "1": { "2": { "get\_star\_ts": "1575185929" } }, "1": { "1": { "get\_star\_ts": "1575185448" } }, "id": "289854", "stars": 6 }, "677144": { "local\_score": 196, "last\_star\_ts": "1575367533", "name": "Vsetky Borufky", "global\_score": 0, "id": "677144", "completion\_day\_level": { "3": { "2": { "get\_star\_ts": "1575367533" } }, "1": { "get\_star\_ts": "1575357449" } }, "1": { "2": { "get\_star\_ts": "1575241741" } }, "1": { "get\_star\_ts": "1575241062" }, "2": { "1": { "get\_star\_ts": "1575264300" } }, "2": { "get\_star\_ts": "1575267832" }, "stars": 6 }, "340516": { "stars": 6, "id": "340516", "completion\_day\_level": { "3": { "1": { "get\_star\_ts": "1575351837" } }, "2": { "get\_star\_ts": "1575376586" } }, "1": { "1": { "get\_star\_ts": "1575351837" } }, "2": { "get\_star\_ts": "1575376586" }, "id": "1575182013", "local\_score": 233, "last\_star\_ts": "1575376586", "name": "Lukáš Gajdošec", "global\_score": 0, "id": "440518", "last\_star\_ts": "1575376586", "name": "Marek Stachera", "local\_score": 0, "completion\_day\_level": {}, "id": "440518", "stars": 0 }, "394723": { "local\_score": 0, "last\_star\_ts": 0, "global\_score": 0, "name": null, "id": "394723", "completion\_day\_level": {}, "stars": 0 }, "229344": { "stars": 6, "id": "229344", "completion\_day\_level": { "3": { "2": { "get\_star\_ts": "1575351770" } }, "1": { "get\_star\_ts": "1575176691" } }, "2": { "2": { "get\_star\_ts": "1575177020" } }, "1": { "get\_star\_ts": "1575264839" }, "name": "Alexka Nyitraiová", "global\_score": 0, "id": "133724", "completion\_day\_level": { "1": { "1": { "get\_star\_ts": "1575210606" } }, "2": { "get\_star\_ts": "1575211974" } }, "2": { "2": { "get\_star\_ts": "1575328931" } }, "1": { "get\_star\_ts": "1575327359" }, "id": "644573", "completi on\_day\_level": { "1": { "1": { "get\_star\_ts": "1575176824" } }, "2": { "get\_star\_ts": "1575171727" } }, "2": { "1": { "get\_star\_ts": "1575263803" } }, "2": { "get\_star\_ts": "1575264339" }, "3": { "2": { "get\_star\_ts": "1575351620" } }, "1": { "get\_star\_ts": "1575351021" }, "local\_score": 245, "last\_star\_ts": "1575351620", "name": "Peter BOROVANSKY", "id": "133724", "stars": 4 }, "229344": { "stars": 6, "id": "229344", "completion\_day\_level": { "3": { "2": { "get\_star\_ts": "1575351770" } }, "1": { "get\_star\_ts": "1575176691" } }, "2": { "2": { "get\_star\_ts": "1575177020" } }, "1": { "get\_star\_ts": "1575264081" } }, "local\_score": 245, "last\_star\_ts": "1575351770", "name": "Matej Magat", "global\_score": 0, "local\_score": 106, "completion\_day\_level": { "2": { "1": { "get\_star\_ts": "1575315504" } }, "2": { "get\_star\_ts": "1575318359" } }, "1": { "2": { "get\_star\_ts": "1575318359" } }, "1": { "2": { "get\_star\_ts": "1575229904" } }, "1": { "get\_star\_ts": "1575228543" }, "id": "668371", "stars": 4 }, "event": "2019"

# Domáca úloha

AoC

- <https://adventofcode.com/>
- Leaderboardu 229344-861e5094.

Vašou úlohou je v aplikácii prečítať tento JSON a interpretovať.

Čo sa chápe pod interpretáciou:

- v master view zobrazíte v jednom riadku ListView mená (ID ak name=null) s počtom hviezd,
- po klinutí na konkrétny riadok (hráča) sa zobrazí detail view, ktorý obsahuje dni a časy, kedy tento hráč vyriešil ktorú úlohu. JSON obsahuje timestampy vo formate long (napr. 1575234400), tie samozrejme prevedťte na čitateľný Date-Time formát. Pre jednoduchosť, dni sú dni adventu 1..25 a úlohy sú v každom dni len 2, teda 1..2.

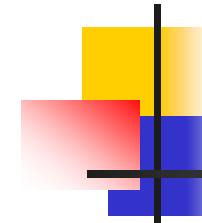
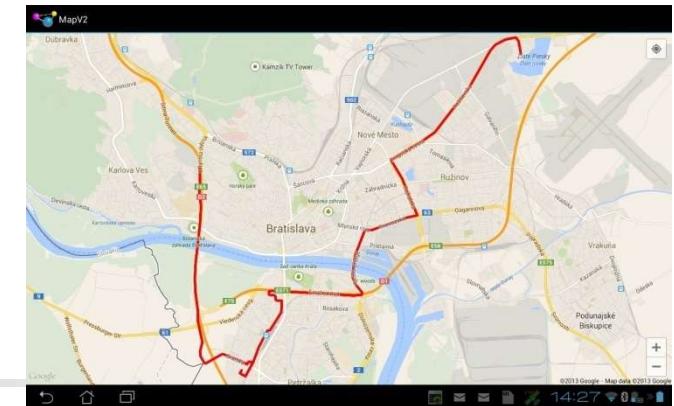
# GoogleDirections

(ako ide matfyzák na pláž)

■ <https://developers.google.com/maps/documentation/directions/>

```
val startLocation = "Mlynska dolina, Bratislava"
                    .replace(" ", "+")
val endLocation = "Zlate piesky, Bratislava"
                    .replace(" ", "+")
val via = "AGEM, Kopcianska, Bratislava|
          Alza, Prievozska, Bratislava".replace(" ", "+")
val urlString =
    "http://maps.googleapis.com/maps/api/directions/json?"+
        "origin=" + startLocation +
        ",+destination=" + endLocation +
        ",+mode=bicycle,&avoid=highways,&waypoints=" +     via
+
        ",+sensor=false"
```

Project:GMapDirections.zip



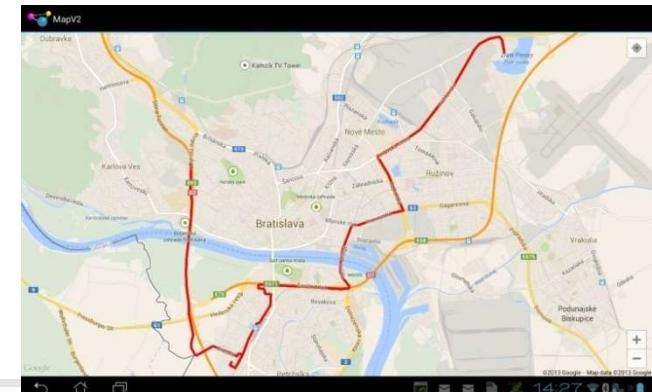
<http://maps.googleapis.com/maps/api/directions/json?origin=Mlynska+dolina,+Bratislava,+&destination=Zlate+piesky,+Bratislava,+&mode=bicycle,+&avoid=highways,+&waypoints=AGEM,+Kopcianska,+Bratislava+Alza,+Prievozska,+Bratislava,+&sensor=false>

# Hľadaná trasa

vo formáte json, resp. xml, na vykreslenie trasy potrebujeme dekódovať

„overview\_polyline“

```
"maneuver" : "keep-right",
"polyline" : {
    "points" : "urbeHyj(gBRAdDm@fGeA"
},
"start_location" : {
    "lat" : 48.187468,
    "lng" : 17.1846087
},
"travel_mode" : "DRIVING"
}
],
"via_waypoint" : []
},
"overview_polyline" : {
    "points" :
"cg) dHybfgBvCWbDEpQSn@B`B`@|AV|@Fb@AjAGdCg@1C_@bBQr@OzGH|I
O|BcA|FoAxKkBrASn@?t@Fr@PbDvA1A@jBF~AQh@S1@a@zCgCfA)@PUJS
AdA`DfcCiCrJsC_CrC~BhCsJaDgCkAeA(AiAuB_BuDqCqE_CiJeEsAk@i@[
Wj@EZKnI_DGCMAEIOIK?MBsCHuGzsQb@s^PmNJ}Lb@yZ
\\oWVsMP(O@oCDwDA]@e@?s@Ca@UgAS]MM[M_@@[JUL(@x@cEpG)p@s@f@
iAmHaEcYoAiIqAcJc@oEaAyLGgAG(@MD]NWJWL)Aj@kAj@uBvAc@
\\CTkCtAgCpAyFtC_CjAgL`G)D|BiF1EkF`FyExDgC~BkCeJiAsDc@gB_C
gB(A_@a@q@)@_AoAk@iAq@uAlmA@)A)GUy@e@)Cc@kDY_DOyBO)FRILQjA
),
"summary" : "Route 61",
"warnings" : [],
"waypoint_order" : [ 0, 1 ]
```



```
- <DirectionsResponse>
  <status>OK</status>
  - <route>
    <summary>Route 61</summary>
    + <leg></leg>
    + <leg></leg>
    + <leg></leg>
    <copyrights>Map data ©2013 Google</copyrights>
  - <overview_polyline>
    - <points>
      cg) dHybfgBvCWbDEpQSn@B`B`@|AV|@Fb@AjAGdCg@1C_@bBQr@OzGH|I
      O|BcA|FoAxKkBrASn@?t@Fr@PbDvA1A@jBF~AQh@S1@a@zCgCfA)@PUJS
      AdA`DfcCiCrJsC_CrC~BhCsJaDgCkAeA(AiAuB_BuDqCqE_CiJeEsAk@i@[
      Wj@EZKnI_DGCMAEIOIK?MBsCHuGzsQb@s^PmNJ}Lb@yZ
      \\oWVsMP(O@oCDwDA]@e@?s@Ca@UgAS]MM[M_@@[JUL(@x@cEpG)p@s@f@
      iAmHaEcYoAiIqAcJc@oEaAyLGgAG(@MD]NWJWL)Aj@kAj@uBvAc@
      \\CTkCtAgCpAyFtC_CjAgL`G)D|BiF1EkF`FyExDgC~BkCeJiAsDc@gB_C
      gB(A_@a@q@)@_AoAk@iAq@uAlmA@)A)GUy@e@)Cc@kDY_DOyBO)FRILQjA
    ),
    <summary>Route 61</summary>
    <warnings>[]</warnings>
    <waypoint_index>0</waypoint_index>
    <waypoint_index>1</waypoint_index>
  + <bounds></bounds>
  </route>
</DirectionsResponse>
```

Project:GMapDirections.zip

<http://maps.googleapis.com/maps/api/directions/xml?origin=Mlynska+dolina,+Bratislava,+&destination=Zlate+piesky,+Bratislava,+&mode=bicycle,+&avoid=highways,+&waypoints=AGEM,+Kopciamska,+Bratislava+Alza,+Prievozska,+Bratislava,+&sensor=false>

# Ako sa dostať k ceste

```
val jsonOutput = response.toString()
val jsonObject = JSONObject(jsonOutput)
val routesArray = // z routes berieme prvú alternatívu
                  jsonObject.getJSONArray("routes")
val route = routesArray.getJSONObject(0)
val poly = // pod route je uzol overview_polyline
           route.getJSONObject("overview_polyline")
polyline = poly.getString("points")
decodePoly(polyline)
```

decodePoly Prémia

ako dekódovať reťazec na zoznam bodov cesty ???

**private List<LatLng> decodePoly(String encoded)**  
je mágia mimo rozsahu tejto prednášky

<http://stackoverflow.com/questions/15924834/decoding-polyline-with-new-google-maps-api>



Project:GMapDirections.zip