

# OkHttp

Download v3.2.0

(<http://github.com/square/okhttp>)

An HTTP & HTTP/2 client for Android and Java applications  
(<http://square.github.io/>)

## Overview

HTTP is the way modern applications network. It's how we exchange data & media. Doing HTTP efficiently makes your stuff load faster and saves bandwidth.

OkHttp is an HTTP client that's efficient by default:

- HTTP/2 support allows all requests to the same host to share a socket.
- Connection pooling reduces request latency (if HTTP/2 isn't available).
- Transparent GZIP shrinks download sizes.
- Response caching avoids the network completely for repeat requests.

OkHttp perseveres when the network is troublesome: it will silently recover from common connection problems. If your service has multiple IP addresses OkHttp will attempt alternate addresses if the first connect fails. This is necessary for IPv4+IPv6 and for services hosted in redundant data centers. OkHttp initiates new connections with modern TLS features (SNI, ALPN), and falls back to TLS 1.0 if the handshake fails.

Using OkHttp is easy. Its request/response API is designed with fluent builders and immutability. It supports both synchronous blocking calls and async calls with callbacks.

OkHttp supports Android 2.3 and above. For Java, the minimum requirement is 1.7.

Overview

Examples

Download

Contributing

License

Wiki  
(<https://github.com/square/okhttp/wiki>)

Javadoc  
([3.x/okhttp/](http://3.x/okhttp/))

StackOverflow  
(<http://stackoverflow.com/questions/tagged/okhttp?sort=active>)

## Examples

### GET A URL

This program downloads a URL and print its contents as a string. Full source (<https://raw.github.com/square/okhttp/master/samples/guide/src/main/java/okhttp3/guide/GetExample.java>).

```
OkHttpClient client = new OkHttpClient();
```

Download v3.2.0

```
String run(String url) throws IOException {
```

```
    Request request = new Request.Builder()
```

(<http://github.com/square/okhttp>)

```
        .url(url)
```

```
        .build();
```

(<http://square.github.io/>)

```
    Response response = client.newCall(request).execute();
```

```
    return response.body().string();
```

```
}
```

## POST TO A SERVER

This program posts data to a service. Full source (<https://raw.githubusercontent.com/square/okhttp/master/samples/guide/src/main/java/okhttp3/guide/PostExample.java>).

```
public static final MediaType JSON
```

```
    = MediaType.parse("application/json; charset=utf-8");
```

```
OkHttpClient client = new OkHttpClient();
```

```
String post(String url, String json) throws IOException {
```

```
    RequestBody body = RequestBody.create(JSON, json);
```

```
    Request request = new Request.Builder()
```

```
        .url(url)
```

```
        .post(body)
```

```
        .build();
```

```
    Response response = client.newCall(request).execute();
```

```
    return response.body().string();
```

```
}
```

*Download*

↓ v3.2.0 JAR (<http://repo1.maven.org/maven2/com/squareup/okhttp3/okhttp/3.2.0/okhttp-3.2.0.jar>)

You'll also need Okio (<http://github.com/square/okio>), which OkHttp uses for fast I/O and resizable buffers. Download the latest JAR ([https://search.maven.org/remote\\_content?g=com.squareup.okio&a=okio&v=LATEST](https://search.maven.org/remote_content?g=com.squareup.okio&a=okio&v=LATEST)).

The source code to OkHttp, its samples, and this website is available on  
 GitHub (<http://github.com/square/okhttp>).

# OkHttp

Download v3.2.0

## MAVEN

(<http://github.com/square/okhttp>)

```
<dependency>
  <groupId>com.squareup.okhttp3</groupId>
  <artifactId>okhttp</artifactId>
  <version>3.2.0</version>
</dependency>
```

(<http://square.github.io/>)

## GRADLE

```
compile 'com.squareup.okhttp3:okhttp:3.2.0'
```

## Contributing

If you would like to contribute code you can do so through GitHub by forking the repository and sending a pull request.

When submitting code, please make every effort to follow existing conventions and style in order to keep the code as readable as possible. Please also make sure your code compiles by running `mvn clean verify`.

Before your code can be accepted into the project you must also sign the Individual Contributor License Agreement (CLA) (<http://squ.re/sign-the-cla>).

## License

Copyright 2016 Square, Inc.

Download v3.2.0

Licensed under the Apache License, Version 2.0 (the "License");

you may not use this file except in compliance with the License.

You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an "AS IS" BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.

See the License for the specific language governing permissions and limitations under the License.

(<http://github.com/square/okhttp>)

(<http://square.github.io/>)

 Square (<https://squareup.com>)