

Branch: master logger / README.md

Find file Copy path

orhanobut

 Bump version to 2.1.1

d847dea on 6 Jun

3 contributors

137 lines (110 sloc) 4.26 KB

Android Arsenal

Logger

AndroidWeekly

#147

glitter

join chat

Methods and size

198 | 18 KB

build

passing

Logger

Simple, pretty and powerful logger for android

Setup

Download

```
compile 'com.orhanobut:logger:2.1.1'
```

Initialize

```
Logger.addLogAdapter(new AndroidLogAdapter());
```

And use

```
Logger.d("hello");
```

Output

D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
W/PRETTY_LOGGER:
W/PRETTY_LOGGER:
W/PRETTY_LOGGER:
W/PRETTY_LOGGER:
W/PRETTY_LOGGER:
I/PRETTY_LOGGER:
I/PRETTY_LOGGER:
I/PRETTY_LOGGER:
E/PRETTY_LOGGER-tag:
E/PRETTY_LOGGER-tag:
E/PRETTY_LOGGER-tag:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
D/PRETTY_LOGGER:
W/MyTag:
W/MyTag:
W/MyTag:

Thread: main

Activity.performCreate (Activity.java:6679)
MainActivity.onCreate (MainActivity.java:29)

Thread info, method info and message

Activity.performCreate (Activity.java:6679)
MainActivity.onCreate (MainActivity.java:43)

no thread info and only 1 method

no thread info and method info

Custom tag for only one use

{
"key": 3,
"value": "something"

[foo, bar]

{key=value, key1=value2}

my log message with my tag

Thread info
Method info
Message

Method info
Message

Only message

Message with 'one-time-use' tag

Json and Xml support

List support

Map and Set support

Global tag

https://github.com/orhanobut/logger/blob/master/README.md

1/4

Options

```
Logger.d("debug");
Logger.e("error");
Logger.w("warning");
Logger.v("verbose");
Logger.i("information");
Logger.wtf("wtf!!!!");
```

String format arguments are supported

```
Logger.d("hello %s", "world");
```

Collections support (only available for debug logs)

```
Logger.d(MAP);
Logger.d(SET);
Logger.d(LIST);
Logger.d(ARRAY);
```

Json and Xml support (output will be in debug level)

```
Logger.json(JSON_CONTENT);
Logger.xml(XML_CONTENT);
```

Advanced

```
FormatStrategy formatStrategy = PrettyFormatStrategy.newBuilder()
    .showThreadInfo(false) // (Optional) Whether to show thread info or not. Default true
    .methodCount(0)        // (Optional) How many method line to show. Default 2
    .methodOffset(7)       // (Optional) Hides internal method calls up to offset. Default 5
    .logStrategy(customLog) // (Optional) Changes the log strategy to print out. Default LogCat
    .tag("My custom tag")   // (Optional) Global tag for every log. Default PRETTY_LOGGER
    .build();

Logger.addLogAdapter(new AndroidLogAdapter(formatStrategy));
```

Loggable

Log adapters checks whether the log should be printed out or not by checking this function. If you want to disable/hide logs for output, override `isLoggable` method and put the condition.

```
Logger.addLogAdapter(new AndroidLogAdapter() {
    @Override public boolean isLoggable(int priority, String tag) {
        return BuildConfig.DEBUG;
    }
});
```

Save logs to the file

//TODO: More information will be added later

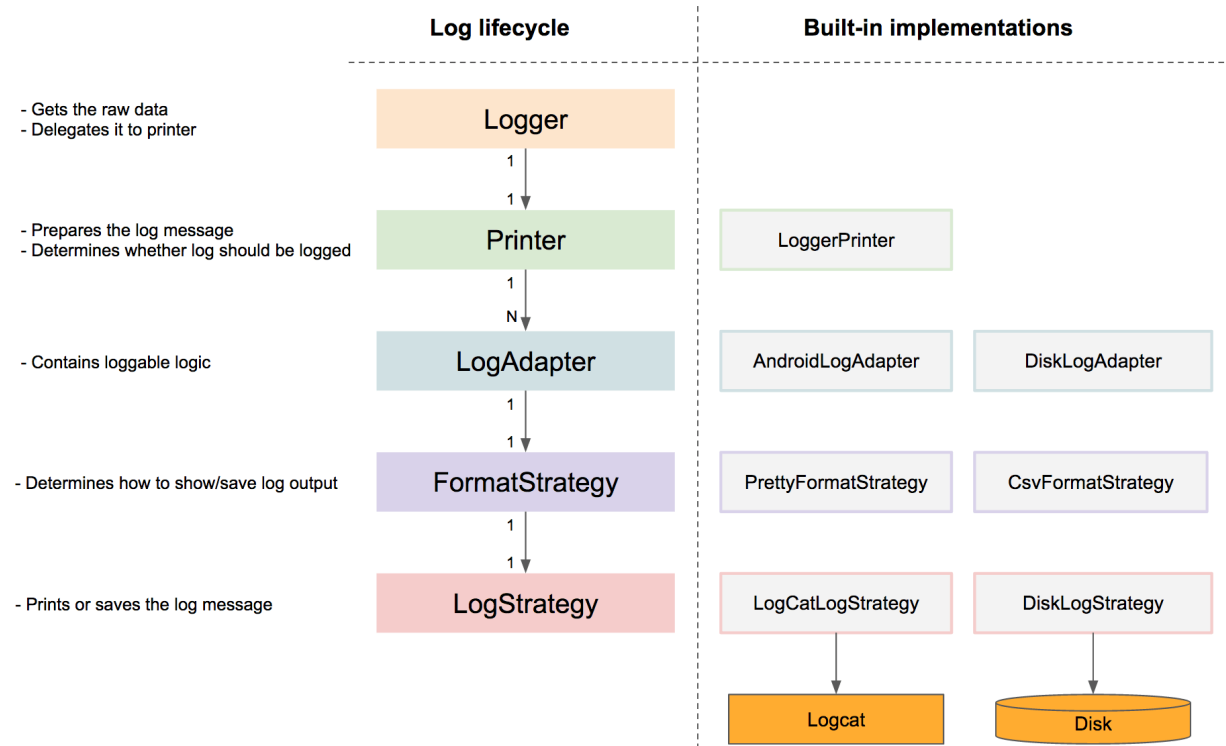
```
Logger.addLogAdapter(new DiskLogAdapter());
```

Add custom tag to Csv format strategy

```
FormatStrategy formatStrategy = CsvFormatStrategy.newBuilder()
    .tag("custom")
    .build();

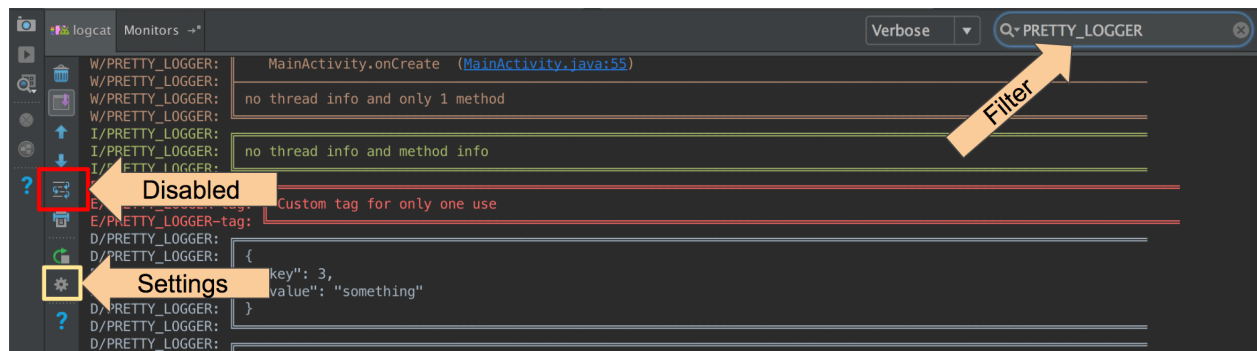
Logger.addLogAdapter(new DiskLogAdapter(formatStrategy));
```

How it works



More

- Use the filter for a better result. PRETTY_LOGGER or your custom tag
- Make sure that wrap option is disabled
- You can also simplify output by changing settings.



- Timber Integration

```
// Set methodOffset to 5 in order to hide internal method calls
Timber.plant(new Timber.DebugTree() {
    @Override protected void log(int priority, String tag, String message, Throwable t) {
        Logger.log(priority, tag, message, t);
    }
});
```

Breaking changes

- Initialization is changed. No backward compatibility support. Use `Logger.addLogAdapter`
- `LogLevel` is removed. Use the new `isLoggable` approach

License

Copyright 2017 Orhan Obut

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.