

Sentry in Flutter



Automatic Error Reporting





What is the Sentry in flutter ?

Sentry is a powerful tool for real-time **error tracking** and performance monitoring, to capture and report errors, crashes, and performance bottlenecks. This helps developers track issues and bugs in production and provides valuable insights into what's going wrong with the app.



```
1  dependencies:
2    sentry_flutter: ^8.8.0
3
```





1- Initialize Sentry in Your App

You'll need to initialize Sentry in the main.dart file.

```
1  Future<void> main() async {
2    await SentryFlutter.init(
3      (options) {
4        options.dsn = 'your-dsn-url-here'; // Replace with your DSN from Sentry
5        options.environment = kReleaseMode ? 'production' : 'development'; // Set the environment
6        options.tracesSampleRate = 1.0; // For automatic tracking of performance
7        // there are more options to set
8      },
9      appRunner: () => runApp(const MyApp()),
10    );
11 }
```





2- Capturing Errors Manually

Sentry **automatically** captures unhandled errors, but you can also **manually** report specific exceptions or errors within your app.



```
1  try {  
2    // Some code that might throw an exception like a network request  
3  } catch (error, stackTrace) {  
4    Sentry.captureException(  
5      error,  
6      stackTrace: stackTrace,  
7    );  
8  }
```





3- Capturing Performance Data

Sentry supports performance tracing in Flutter apps, such as tracking transaction spans like app startup time or specific HTTP requests.

For **automatic tracking** of performance, set the `tracesSampleRate` option when initializing Sentry:
`options.tracesSampleRate = 1.0;`

```
1 final transaction = Sentry.startTransaction('app-start', 'navigation');
2 try {
3   // App initialization logic
4 } finally {
5   transaction.finish();
6 }
```





Ahmed Hussein
Flutter Developer

THANK YOU



Was This Helpful?

Like, Share and Follow
for more Content



Like



Comment



Share



Save

