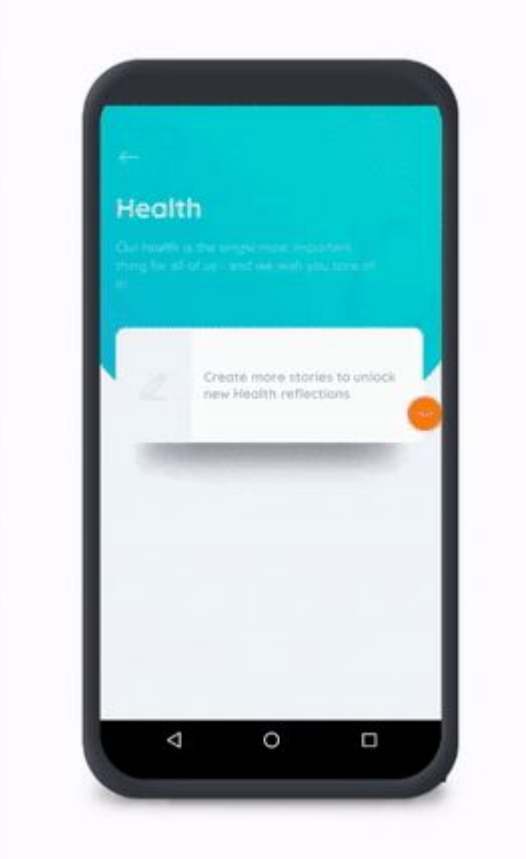
Vince Varga

Full-stack Dart Development

Part I (?)

Flutter Munich
Tuesday, December 3, 2019



Reflectly: An award winning mindfulness app built with Flutter.
Source: flutter.dev

Flutter is Google's UI toolkit for building beautiful, natively compiled applications for mobile, web, and desktop from a single codebase.

You should never have to say no to your designer

Matt Carroll

Flutter vs Web with Matt Carroll

1,887 views • Jan 25, 2019

70 0

SFHTML5
7.27K subscribers

youtube.com/watch?v=5O795a02L94

If you write Flutter apps, you

write Dart code

Dart

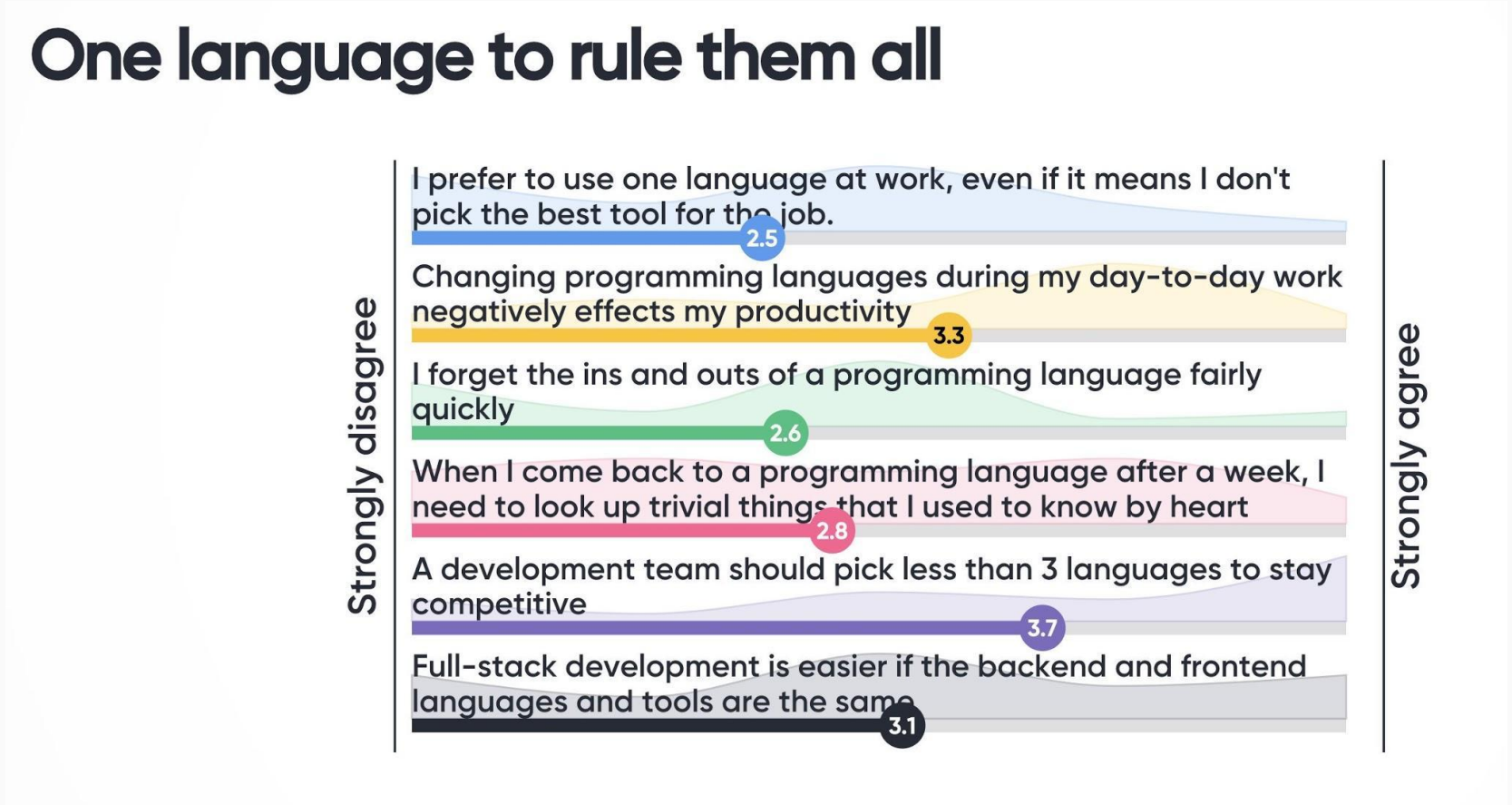
Let me share some details about the companies I worked for

- Your environment might be different
- Your experience level might be different
- Your priorities might be different
- Your background might be different

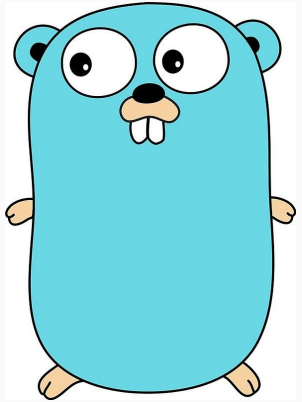
Result: you might not come to the same conclusions that I did

By sharing some stories, you'll understand how we are similar and how we are different.

My poll from the last Flutter Munich event...



Small Dev Team A



Ruby development à la Vince

My search history

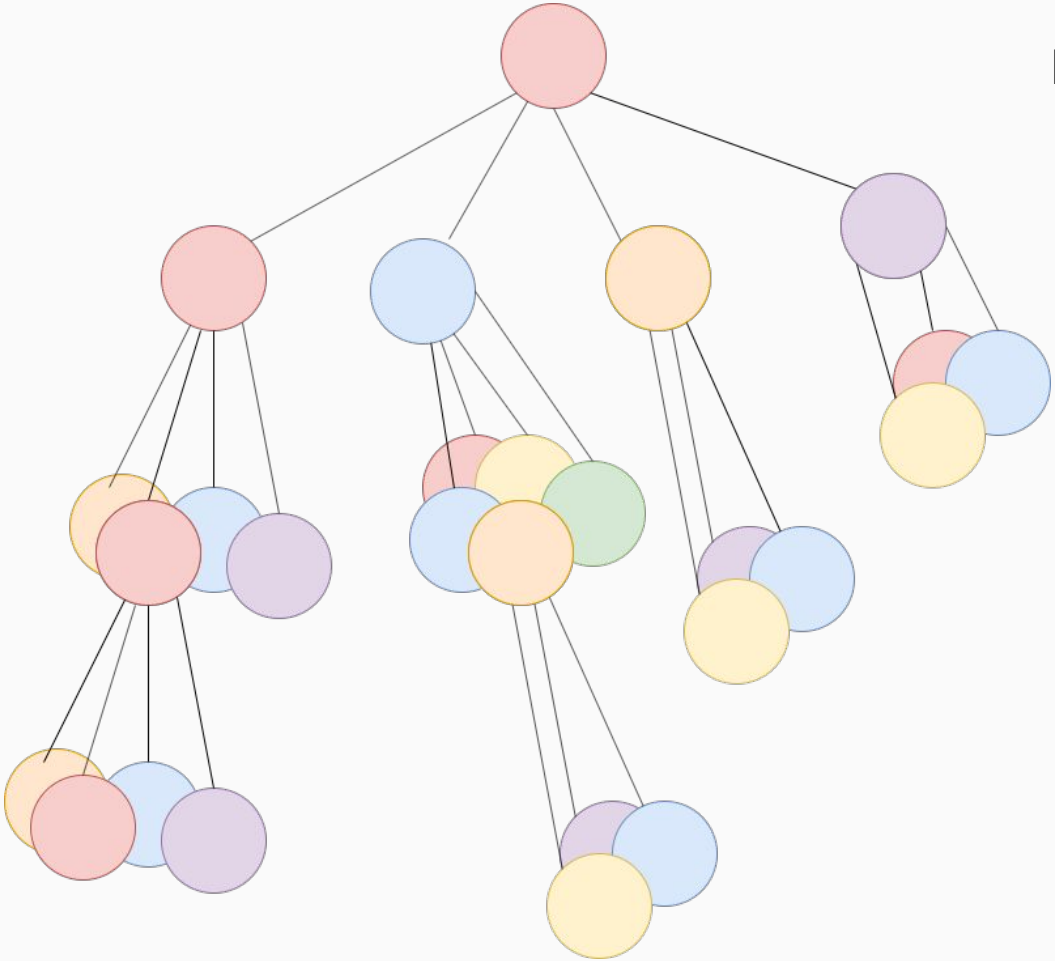
- Ruby strings
- How to declare a variable in Ruby
- switch case Ruby
- If else in Ruby
- Objects in Ruby
- Dictionaries in Ruby
- Hash in Ruby
- Get keys in hash Ruby
- Lists in Ruby
- ...

Word-for-word

Literal translation doesn’t always work.

- **Hungarians** don't “jump for joy”, they are “as happy as a monkey about its tail” (Örül, mint majom a farkának).
- **Hungarians** don't say “Bullshit!”, they say “Horse dick!” (Lófasz!)

Psych. Gotcha!



You picked a language.
Now test. lint. fmt. dep. build.

Small Dev Team B

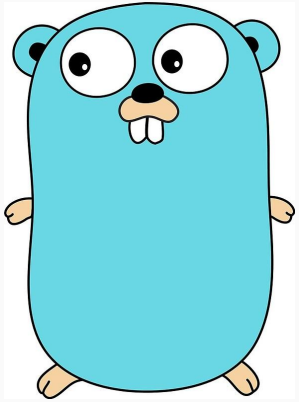
One language

1

In this case, TypeScript

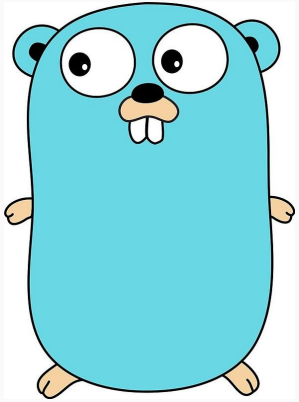
- People had time to learn the language well.
- Proper code review.
- Better quality code hits production.
- People can shift from backend to frontend, from one project to another.
- People can help others out.
- Share knowledge, give advice.
- Consistent tooling.

Small Dev Team C



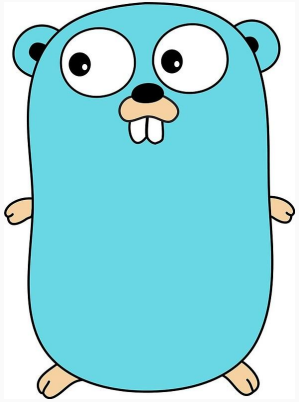
Dart

Small Dev Team C



Dart

Small Dev Team C



Dart

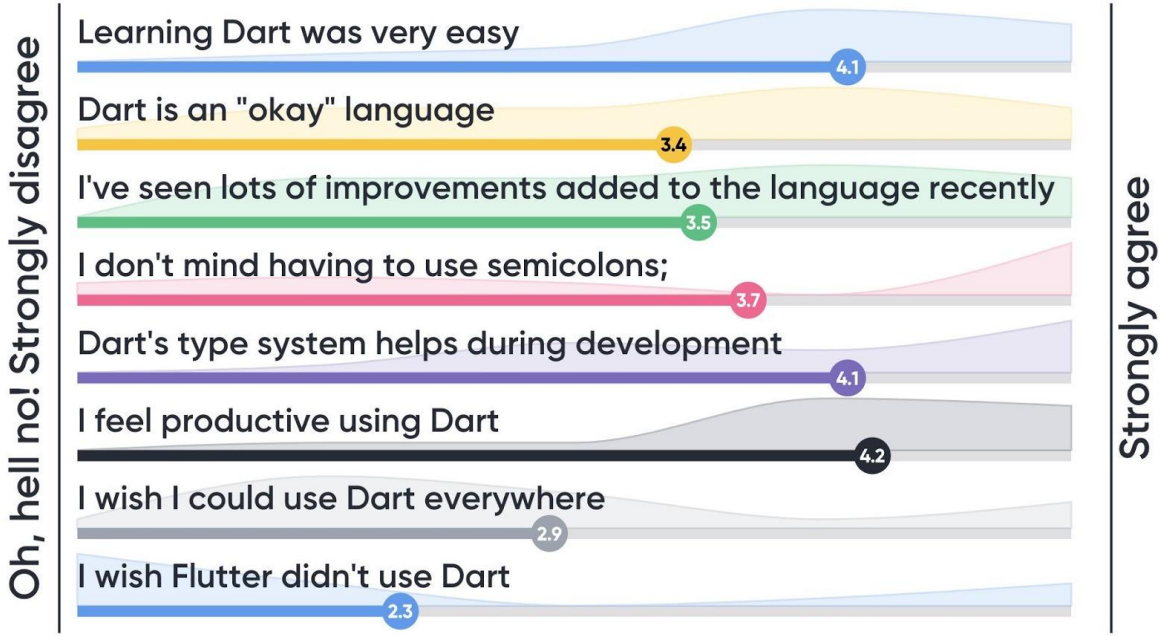
Hmm....

Reducing the number of languages can be worth it.

Dart

The Dart Programming Language

Mentimeter



18

CHANGE IN PROGRAMMING LANGUAGE USE, 2018–2019

01	Dart	532%
02	Rust	235%
03	HCL	213%
04	Kotlin	182%

Fastest growing languages

With Flutter in our trending repositories, it's not surprising that Dart gained contributors this year. We also saw trends toward statically typed languages focused on type safety and interoperability: the Rust, Kotlin, and TypeScript communities are still growing fast.*

Source: https://octoverse.github.com/

Change in Programming Language Use
Dart 532%

Dart is growing and improving (2.3)

Spread operator

```
Row(
  children: [
              ...northAmericanCountries,
              ...asianCountries,
              ...europeanCountries
            ],
)
```

If (and for) in a collection

```
var countriesToShow = [
    ...northAmericanCountries,
    ...europeanCountries,
    if (isAsian)
        ...asianCountries
];
```

dart:ffi

- Dart 2.5+
- Still in Beta
- Call C
- Sass, SQLite...

“Dart mobile, command-line, and server apps running on the Dart Native platform can use the dart:ffi library to call native C APIs.

FFI stands for foreign function interface.

Other terms for similar functionality include native interface and language bindings.”

dart2native

Use the dart2native command to AOT (ahead-of-time) compile a Dart program to native x64 machine code. The dart2native command is supported on Windows, macOS, and Linux.

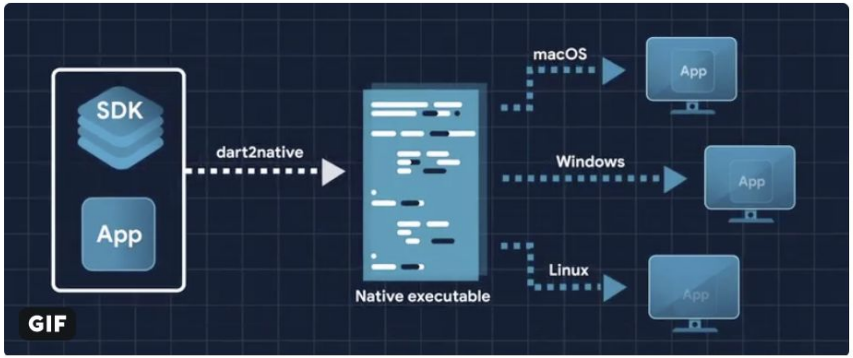
Dart Language
@dart_lang

💥Stable release of Dart 2.6 SDK!

Dart2native AOT-compiles Dart programs to self-contained executables for macOS, Windows, or Linux.

Create tools for the command line that your users can run even if they don't have the Dart SDK.

More → goo.gle/2JN1Yl0



♡ 844 6:11 PM - Nov 5, 2019

333 people are talking about this

dart2native

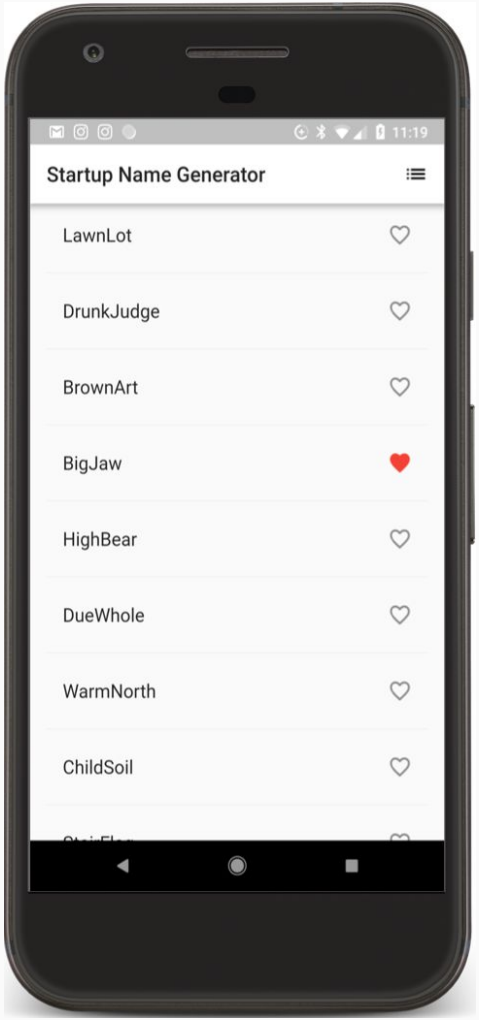
Paul Mundt reduced the size of a Docker image by more than 90% by using native code.

220 MB > 20 MB

Experiments with Dart Microservices
https://itnext.io/e-fa117aa408c7

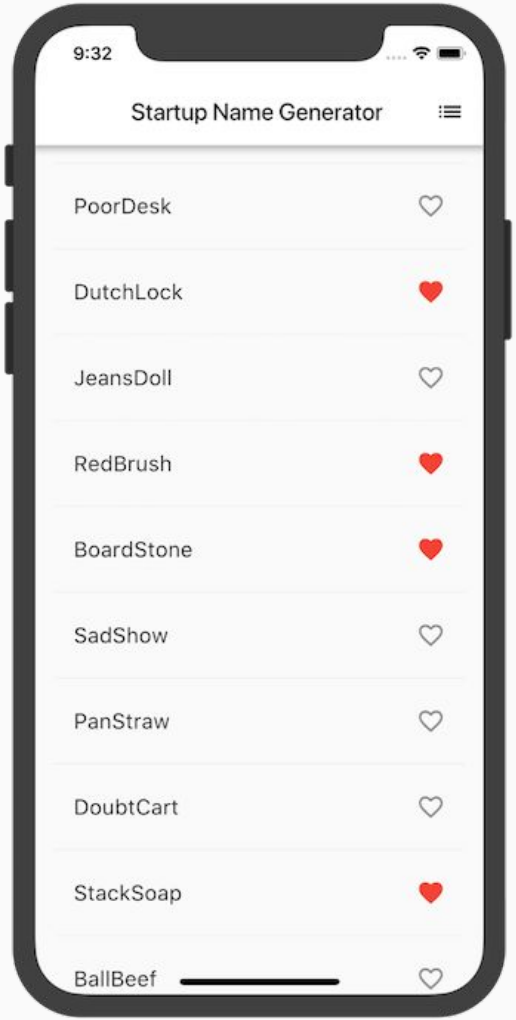
Reducing the number of languages can be worth it.

The language should be full stack.

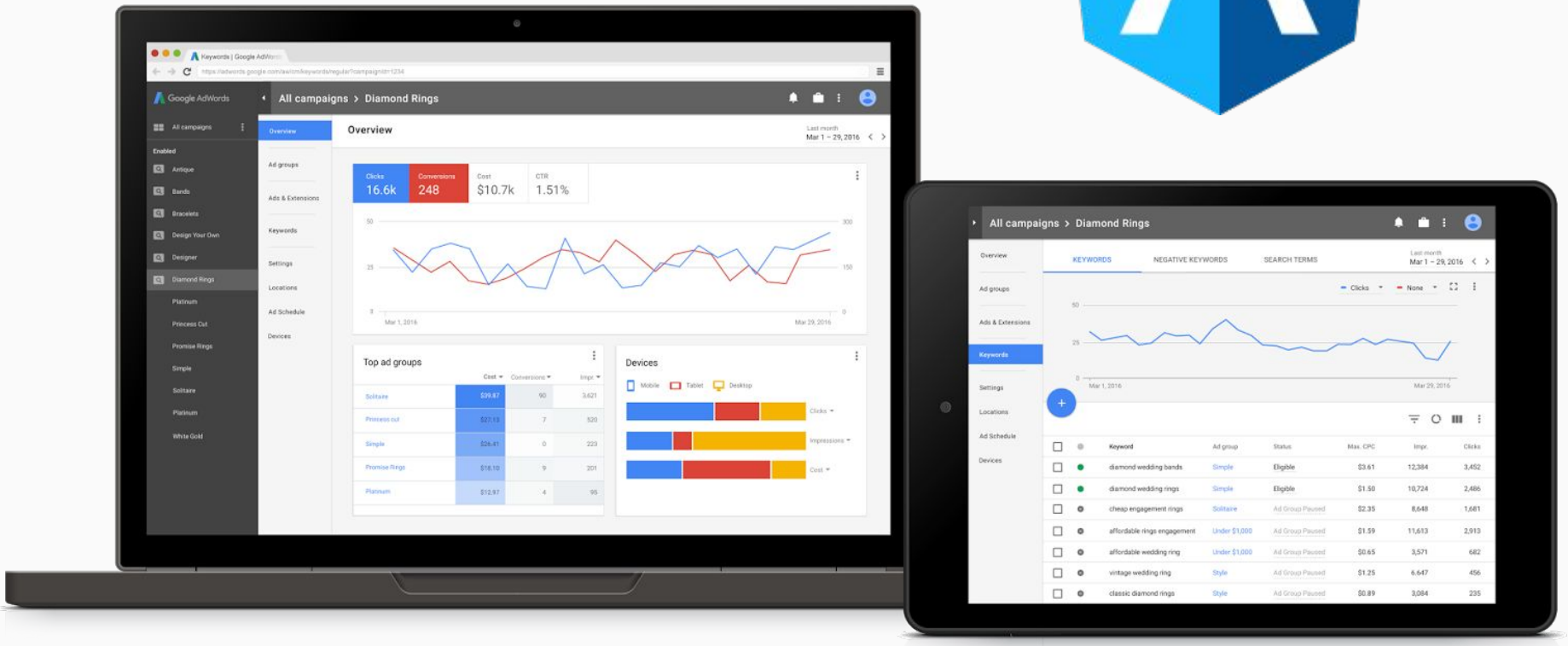


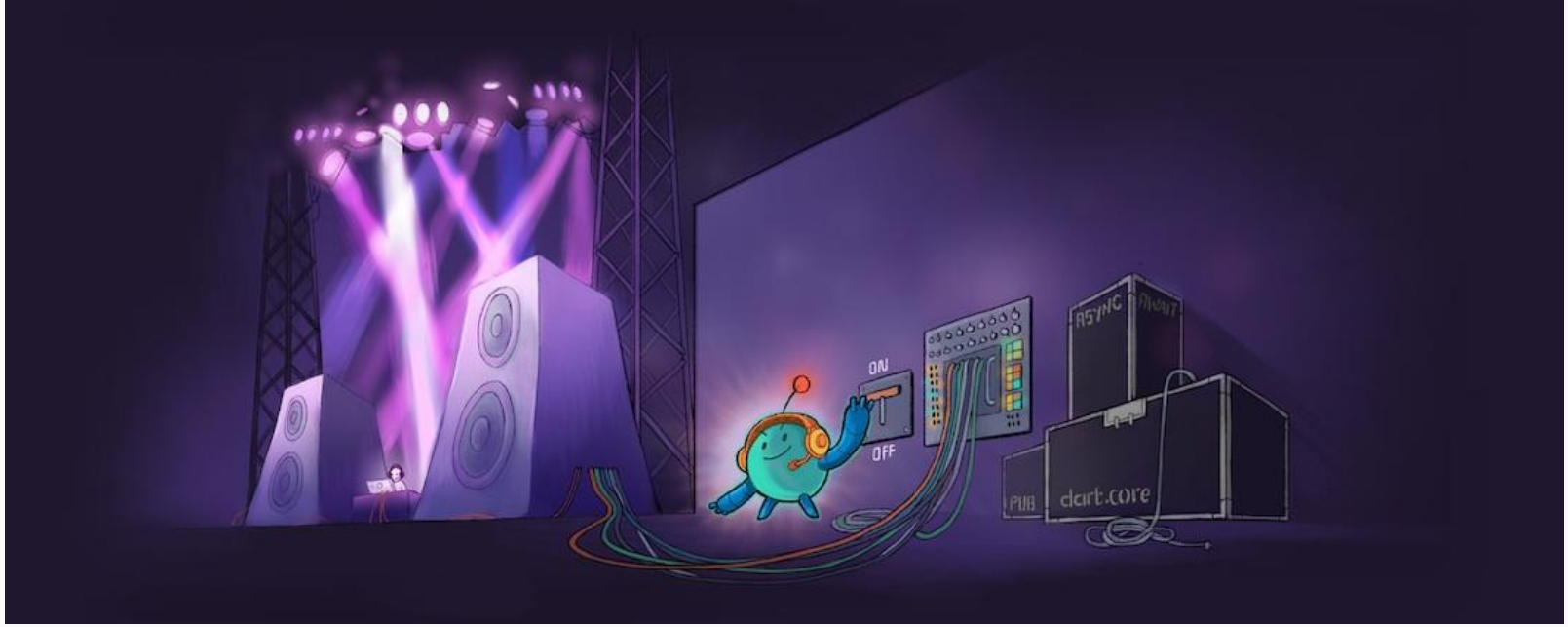
Flutter

android



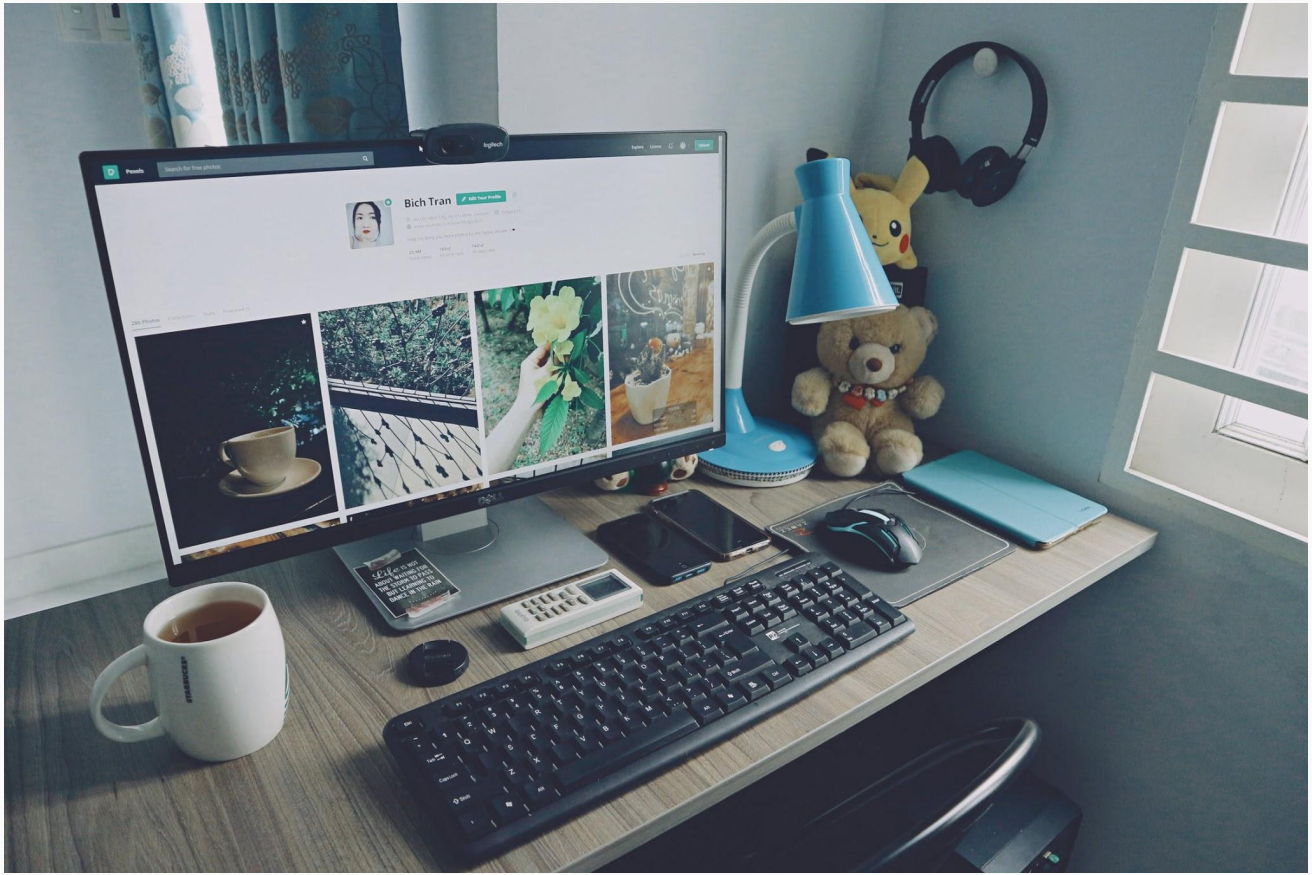
Flutter Codelab - Write your first Flutter app


Google AdWords
All campaigns > Diamond Rings
Overview
Clicks 16.6k
Conversions 248
Cost $10.7k
CTR 1.51%
Top ad groups
Devices
KEYWORDS
NEGATIVE KEYWORDS
SEARCH TERMS



- AngularDart
- Flutter Web (tech preview)
- Keep it simple. Dart:html library
- Roll your own framework?

Stagehand is the Dart project scaffolding generator.



Flutter

Maybe even desktop?

AQUEDUCT

An object-oriented, multi-threaded HTTP server framework written in Dart.

ANGEL

A polished, production-ready backend framework in Dart.

dart-lang / **shelf**

<> Code | Issues 16 | Pull requests 0 | Actions | Proj

Web server middleware for Dart https://pub.dev/packages/shelf

Examples

```
test('POST /health', () async {
  final response = await harness.agent.post('/health');
  expect(response.statusCode, 400);
});


test('GET /health', () async {
  final response = await harness.agent.get('/health');
  expect(response.statusCode, 200);
});
```

```dart
class DartMeetupCloneServerChannel extends ApplicationChannel {
  // ...

  @override
  Controller get entryPoint {
    return Router()
      // ...
      ..route('/health').link(() => HealthController());
  }
}
```

```dart
import 'package:aqueduct/aqueduct.dart';

class HealthController extends Controller {
  @override
  FutureOr<RequestOrResponse> handle(Request request) {
    if (request.method == 'GET') {
      return Response.ok({});
    }
    return Response.badRequest();
  }
}
```

```dart
import 'package:aqueduct/managed_auth.dart';

class User extends ManagedObject<_User>
    implements _User, ManagedAuthResourceOwner<_User> {
  @Serialize(input: true, output: false)
  String password;
}

class _User extends ResourceOwnerTableDefinition {}
```

```dart
class ExampleAppChannel extends ApplicationChannel {
  ManagedContext context;
  AuthServer authServer;

  @override
  Future prepare() async {
    final configFilePath = options.configurationFilePath;
    final config = DartMeetupCloneServerConfiguration(configFilePath);
    context = contextWithConnectionInfo(config.database);
    authServer = AuthServer(ManagedAuthDelegate<User>(context));
  }

  @override
  Controller get entryPoint {
    return Router()
      ..route('/register').link(() => RegisterController(context, authServer))
      ..route('/auth/token').link(() => AuthController(authServer))
      ..route('/protected')
          .link(() => Authorizer.bearer(authServer))
          .linkFunction((request) => Response.ok({}));
  }
```

```
class RegisterController extends ResourceController {
  RegisterController(this.context, this.authServer);

  final ManagedContext context;
  final AuthServer authServer;

  @Operation.post()
  Future<Response> createUser(@Bind.body() User user) async {
    // Omitted: Validate request parameters
    user
      ..salt = AuthUtility.generateRandomSalt()
      ..hashedPassword = authServer.hashPassword(user.password, user.salt);

    final createdUser = await Query(context, values: user).insert();

    return Response.ok({
      'username': createdUser.username,
      'id': createdUser.id,
    });
  }
}
```

Productive Request Bindings

Boilerplate-avoiding meta-programming increases productivity, avoids simple errors and makes sure your API handles errors correctly and consistently.

```
@Operation.put('id')
Future<Response> updateTeam(
    @Bind.path('id') int teamID,
    @Bind.body() Team team,
    @Bind.header('x-required-header') String requiredHeader,
    {@Bind.query('optional_timestamp') DateTime timestamp}) {
  return Response.ok(team);
}
```

ManagedObjectController

```
/// ManagedObjectController
/// A [Controller] that implements basic CRUD operations for a [ManagedObject].
///
/// - GET /<name>/:id -> Fetch Object by ID
/// - PUT /<name>/:id -> Update Object by ID, HTTP Request Body contains update values.
/// - DELETE /<name>/:id -> Delete Object by ID
/// - POST /<name> -> Create new Object, HTTP Request Body contains update values.
/// - GET /<name> -> Fetch instances of Object
///
```

ManagedObjectController

```
/// GET requests with no path parameter
/// can take extra query parameters
///
/// - count (integer)
/// - offset (integer)
/// - pageBy (string)
/// - pageAfter (string)
/// - pagePrior (string)
/// - sortBy (string)
```

ManagedObjectController

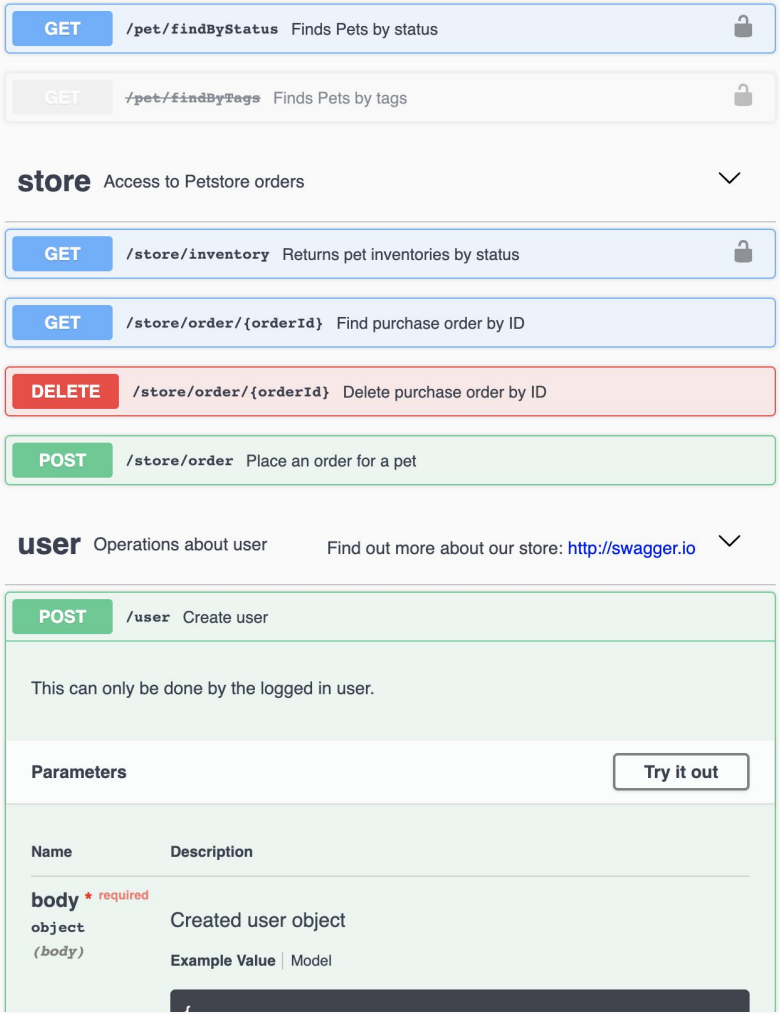
```
/// You may use this class without subclassing,
/// but you may also subclass it to modify the executed [Query]
/// prior to its execution, or modify the returned [Response]
/// after the query has been completed.
///
/// The HTTP response body is encoded according to [responseContentType].
```

```
class Label extends ManagedObject<_Label> implements _Label {}

class _Label {
  @primaryKey
  int id;

  String name;

  // ....
}
```

```
@override
Controller get entryPoint {
  return Router()
    // ...
    ..route('/label/[:id]')
        .link(() => ManagedObjectController<Label>(context));
}
```

Databases and ORM

- Docs is pretty good
  - One-to-one, one-to-many, many-to-many relationships
  - Custom table and column names
  - Transactions
  - Validations
- Migrations tool
  - Worked well
  - Only upgrade is autogenerated
  - If you want Downgrade, you need to write it
- Postgres only



OpenAPI 3 Integration

Powerful reflection and static analysis tools allow you to automatically generate OpenAPI 3 documents from your source code. Generate client integration code and deploy first class documentation without extra effort.

```
openapi: "3.0.0"
info:
  version: 1.0.0
  title: Aqueduct API
paths:
  /aqueduct:
    ...
components:
  schemas:
    ...
```

Swagger Petstore example app and Aqueduct webpage

dart2native

Paul Mundt reduced the size of a Docker image by more than 90% by using native code.

220 MB > 20 MB

Experiments with Dart Microservices
https://itnext.io/e-fa117aa408c7

dart2native

Known limitations...

The initial (Dart 2.6) version of dart2native has some known limitations:

- **No cross-compilation support** (issue 28617)

The compiler supports creating machine code only for the operating system it's running on. You need to run the compiler three times — on macOS, Windows, and Linux .

- **The format of the executables isn't compatible with standard signing tools such as codesign and SignTool.**
- **No support for dart:mirrors and dart:developer**

The code compiled by dart2native can use all of the other libraries that the Dart VM supports.

https://github.com/stablekernel/aqueduct/issues/669
https://github.com/stablekernel/aqueduct/issues/738

#669

joeconwaystk commented on Oct 17 Member

I would expect that effort can be wrapped up late Q4,
but don't want to make any promises

VERDICT

Things work

Good docs

One language

Dart packages for backend

Dart packages for backend

Performance?

Let’s be honest.
Dart iss not the obvious choice for backend development.

Let’s be honest.
Dart is not the obvious choice for backend development.
YET.

Dankeschööööön