Flutter path

1. Learn the work of each file in the flutter project, example:
    a. What is the **pubspeck.lock** function and should I push to the repo?
    b. What is **analysis_options**?
    c. What is the generated_plugin_registrant.dart function?
    d. Learn each file and its functionality to understand what flutter does in the background
2. Always start working creating test first: read about **TDD**
    a. https://medium.com/upday-devs/flutter-test-driven-development-e57f2defff43
3. Great books to read about coding
    a. Clean Code
    b. The clean coder
    c. Clean Architecture
4. Learn about Dart, everything:
    a. Dart guides: https://dart.dev/guides
    b. Dart documantation: https://dart.dev/tools/dart-doc
5. Build all kinds of front-end structures
    a. Create, experiment and break everything: https://github.com/Solido/awesome-flutter
    b. Learn about DDD architecture: https://www.youtube.com/watch?v=RMiN59x3uH0&list=PLB6lc7nQ1n4iS5p-IezFFgqP6YvAJy84U
    c. https://www.youtube.com/c/LearnFlutterCode
    d. https://www.youtube.com/c/TechieBlossom
6. How to implement design patterns
    a. https://refactoring.guru/design-patterns
    b. **Design Patterns** to dart, there are multiple articles about this but this guy wrote awesome articles: https://medium.com/flutter-community/flutter-design-patterns-0-introduction-5e88cfff6792
    c. Remember, you don't have to apply these patterns to everything but they can be a great tool in your daily development
7. Learn how to write your own packages, no matter the complexity, just do it
8. Understood the difference between **streams and futures**
9. Work with different state managements and learn how to test them:
    a. flutter bloc
    b. riverpod
    c. provider
10. How to save data in the local storage:
    a. SQFlite: https://pub.dev/packages/sqflite
    b. HydratedBloc: https://pub.dev/packages/hydrated_bloc
11. Always It's really useful learning how to use google maps, firebase or firestore
12. Learn how to create an authentication with firebase authentication (really easy), you can find a lot of articles about this
13. Study the differences between unit testing, component testing and automated testing and apply them to your projects

14. Learn something about android and iOS, (you don't have to be an expert at the very beggining but you can always improve every day), most of the time you'll work with flutter but sometimes you'll need to create a bridge between native platforms and flutter. Learn how to make a bridge
    a. Excellent page to learn about (iOS, Android and flutter): https://www.raywenderlich.com/
    b. https://www.raywenderlich.com/30342553-platform-specific-code-with-flutter-method-channel-getting-started
15. Learn about **isolates**: this will help you out to make something in background while the user uses anothe app but the flutter app is still open
16. Learn about responsive for apps, web and desktop platforms
    a. Create web apps
    b. Create mobile apps
    c. Create everything, create, create, It's the key, repeat, learn, fix bugs, create tests and again.
17. Learn about flutter canvas or CustomPainter
18. Learn about firebase flavors (Get multiple databases for different platforms)
    a. There are multiple articles about this but you have to search which work better for you
19. Learn how to use **CI/CD (**deploy your app**)** https://codemagic.io/start/
    a. There are more CI/CD but I use this one

The best way to learn is creating your own apps, you can read, watch tutorials or courses but if you don't put that knowledge in practice nothing will happen, you're going to suffer fixing a lot of bugs, sometimes you'll feel overhelmed but never give up, there will developers that will help you and another will not at all, always push yourself to the limits and some day It will worth it.

I hope this can help you, these are some of the things I use or used to learn sometimes I wrote straignt to github issues in some packages, because you won't find always the answer in stak overflow, you must go to github and read the flutter code or the packages code to learn or create your own solution and also I read a lot medium articles so always stay update.

Do interviews, It's a great way to measure your progress and learn all the things you must to learn, I know that is a lot and there are more things to learn but you'll realize in the path. have a great time and never give up.