To be put simply, static testing is testing the product without executing any code. This can be reviewing of requirements, or analysis of the program for weaknesses that may cause problems or defects later. Here we can find defects related to deviation from standards, missing requirements, poor design, maintainability issues, or bad specifications.

Dynamic testing, on the other hand, is the opposite of this. Dynamic testing occurs after the program is built. The program is executed, and the behavior is then examined for defects afterwards. These are usually functional defects that would affect the experience of the user when using your application.

Hopefully, the difference between these two types of testing is apparent. I will mention that it is better to find the defects that may arise in static testing rather than dynamic. This is because it significantly reduces cost. Not only with development time but can also reduce cost incurred when maintaining the application.

Of course, both methods of testing are important. While much is gained by finding defects during static testing, it would be folly to expect everything to be found during the collaboration phase of the development process and therefore practical testing and real-world testing is required so few defects make it to go-live. The point is to obviously create the best application that can be made for your use-case and that means finding as many defects as possible.