



# Reactive Programming

---

With Spring Framework 5

RestTemplate vs WebClient



## Spring MVC & Spring WebFlux

@Controller / @RequestMapping

Router Functions

spring-webmvc

spring-webflux

Servlet API

HTTP / Reactive Streams

Servlet Container

Tomcat, Jetty, Netty, Undertow





## Spring RestTemplate

- Spring RestTemplate:
  - Is Blocking
  - Based on Java Servlet API
  - Used one thread per request
    - Will block during request





## Spring WebClient

- Spring WebClient:
  - Is asynchronous, non-blocking
  - Based on Spring Reactive Framework
  - Requests are 'queued' and processed when available
  - Uses Mono & Flux types
    - Allows for functional / declarative programming





## Which to Use?

- Performance will be similar until a heavier load
- RestTemplate has been in Spring for a long time and is very mature
- RestTemplate has been moved to 'maintenance' mode by Spring
  - Meaning no new features will be developed
- WebClient is being actively developed by Spring
- At some point in the future RestTemplate will be deprecated
  - This is likely YEARS into the future





