Creating Controllers

Applications will have many services, if you have certain requests a server will handle, the controller will handle those things.

According to the Spring Boot Docs:  
Controllers provide access to the application behavior that you typically define through a service interface. Controllers interpret user input and transform it into a model that is represented to the user by the view. Spring implements a controller in a very abstract way, which enables you to create a wide variety of controllers.

Restful Applications

Controllers mainly control how data is displayed, it seems like

In conclusion it seems like with Dependency injections we can split things into layers.

@Controller

Spring Boot will take care of the configuration, but if you want more specific configurations you have to go and code that information.

Rest Controllers

Spring boot uses two main controllers, either @Controller or @RestController. The main difference between the two is that @RestController is a combination of @Controller and @ResponseBody. According to BaelDung, @ResponseBody primarily returns the information that the object is already in a JSON format and passed back into HttpResponse.

@Restcontroller

* Returns the JSON response
* So that we can have a method and expose some endpoints that clients can consume, clients being React, android etc…

Sources:

<https://www.geeksforgeeks.org/spring-rest-controller/#>

<https://www.baeldung.com/spring-request-response-body>

<https://www.baeldung.com/spring-controllers>

https://docs.spring.io/spring-framework/docs/3.0.0.M4/reference/html/ch15s03.html