**📩 Request**

When a bean has a request scope, it means that a**new instance of the bean is created for each HTTP request that arrives at the server**. This instance of the bean will be available for the duration of the processing of that specific request, and will be destroyed at the end of the request.

@Component  
@RequestScope  
public class RequestBean {  
   
}

Here are some key features of the scope request in Spring:

* **One instance per request**: Each time a web request arrives at the server, a new instance of the bean with scope request is created for that specific request. This allows to have a specific state for each request and avoid sharing information between different requests.
* **In-request state sharing**: Bean instances with request scope share their state within the context of the request in which they were created. This means that all components or classes accessing the same bean in the same request will see the same state of the bean.
* **Suitable for request-specific data**: The request scope is suitable for beans that contain HTTP request-specific data, such as user information, request parameters, or request-specific session data.
* **Limited to request context**: Bean instances with request scope are only accessible within the context of the request in which they were created. They cannot be accessed from other requests or outside the context of the current request.
* **Useful in web applications**: The request scope is especially useful in web applications, where you need to have a specific bean instance for each request and you want to share the same bean between different components within the same request lifecycle.

It is important to note that the Spring request scope only makes sense in the context of web applications, and will only be available in the context of a web request that is handled by a Spring framework, such as Spring MVC. In addition, care should be taken when handling shared data in beans with scope request, as it can have implications on concurrency and state integrity in applications with multiple concurrent requests.