

Q1:

Output:

bye from message1

Non-dependency and only one method that does not need to be predicted will appear as it is

Q2:

Output:

bye from message1

bye from message2

Its appearance depends on the appearance of the getMessage1() method due to the presence of @Qualifier

Q3:

Output:

bye from message1

bye from message3

bye from message2

@Qualifier was used 3 times, but the only time it was used was in getMessage2(@Qualifier("3") String data), so the reason for the appearance of this method depends on the appearance of getMessage3()

Q4:

Output:

bye from message1

bye from Main controller

bye from message3

bye from message2

The appearance of message1 depends on the appearance of the Main controller, so message1 will appear, then the Main controller, as well as message2, depend on the appearance of message3

Q5:

Output:

bye from message3

bye from message2

bye from Main controller

bye from message1

Message3 appeared because it is non-dependent, then Message2 because it depends on Message3, and MainController because it depends on Message2, and the last message will be Message1.

***Non-dependencies are always printed before dependencies, unless @Qualifier is placed based on the connection of the method to each other. The codes and values placed inside @Qualifier are printed just as symbols for the link. Number 1 does not mean that it is in the order of appearance. The first will be just predictions of the appearance. The first non-dependency method I expect will appear first. Then, if there is a link. It has another method that I put, and if I do not find it, I move to the other non-reliable method, and so on until I reach the reliable one.**