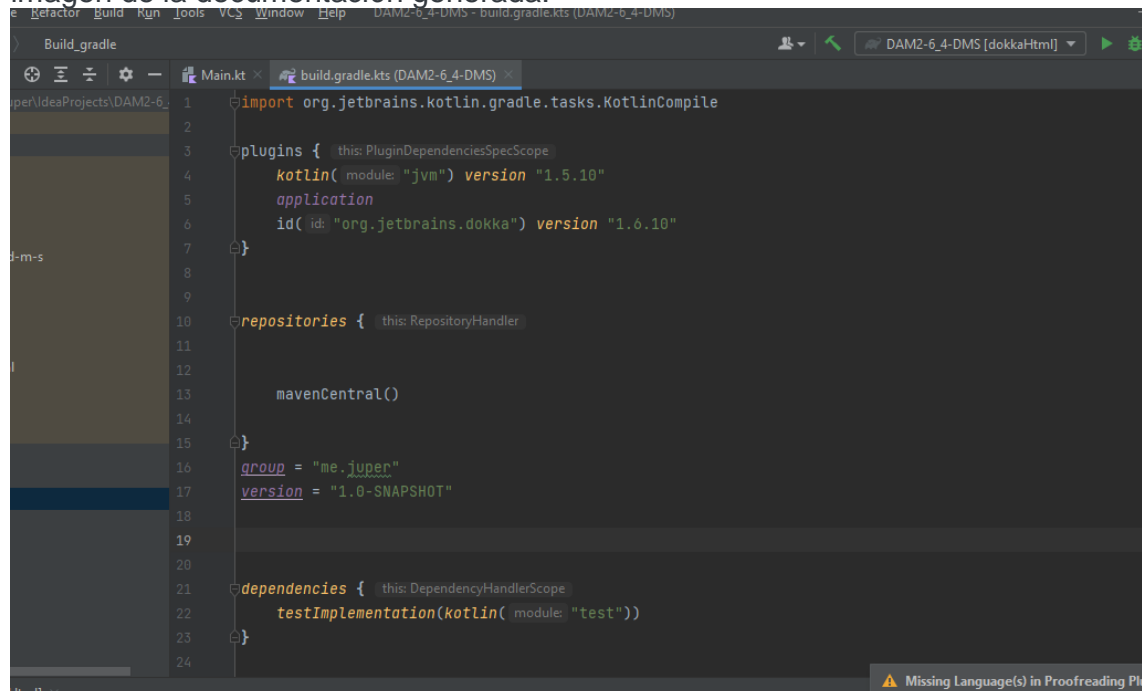
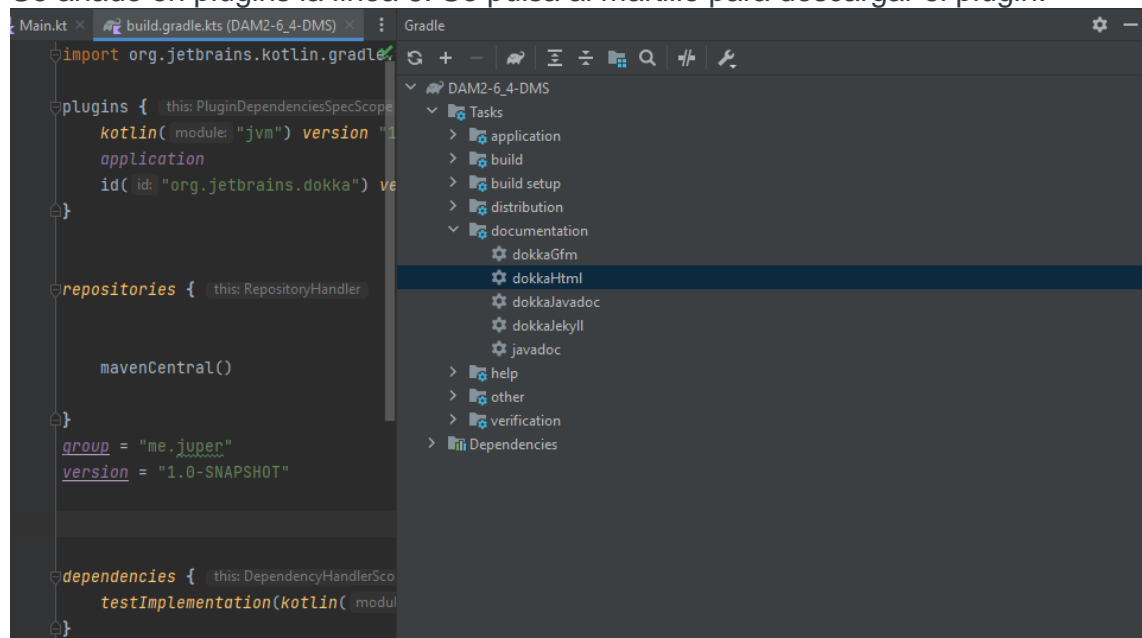


Relación 6.4

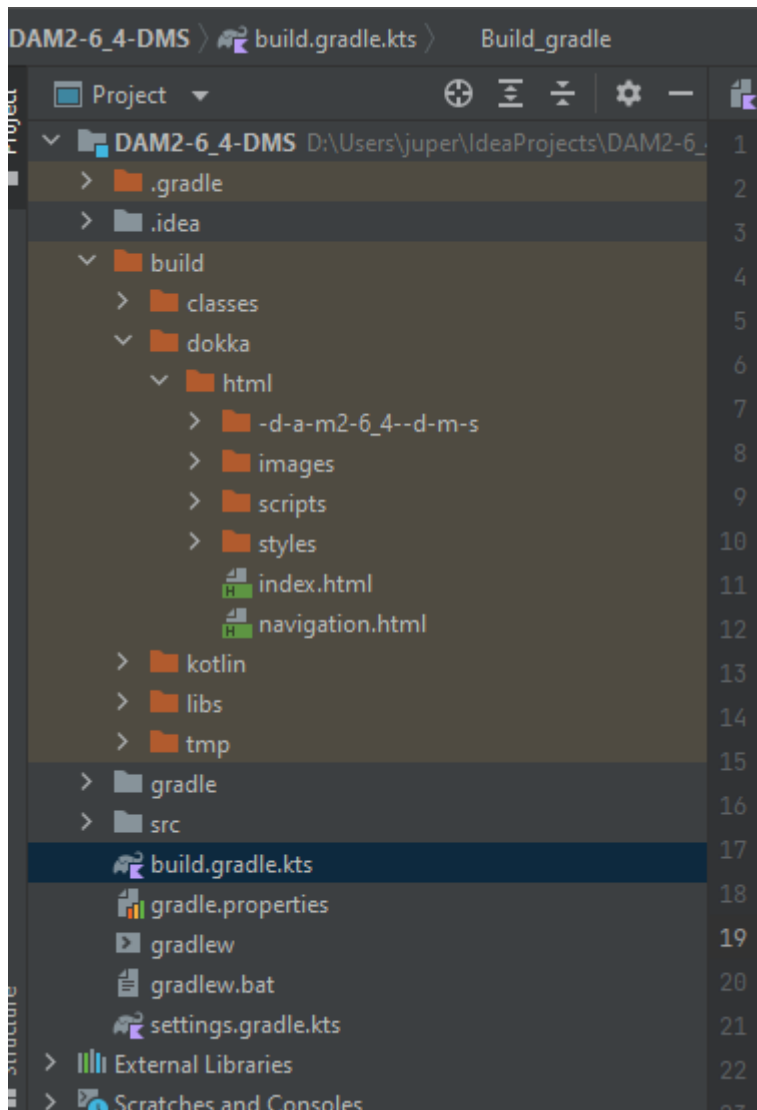
a) Memoria de instalación y generación de documentación, con una captura de imagen de la documentación generada.



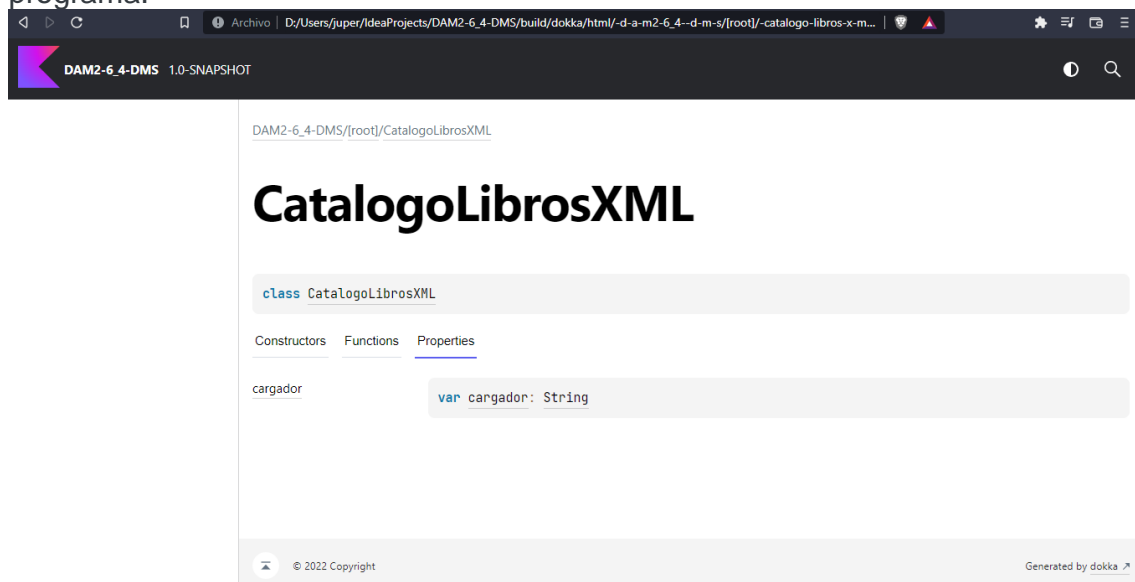
Se añade en plugins la línea 6. Se pulsa al martillo para descargar el plugin.



Se abre en panel “Gradle”, “Task”, “Documentation” y por último si se presiona en “dokkaHtml” se genera la documentación.



En build, dokka, html, index.html ya accedemos a la documentación de nuestro programa.



b) Responde a las siguientes preguntas:

1. ¿Por qué es importante documentar?

La documentación es importante para informar del uso que se le puede dar a tu programa, además de explicar el procedimiento que se ha utilizado para llevar a cabo el programa, definiendo y explicando los procesos menos intuitivos, o aclarando porqué has decidido hacerlo de cierta forma, con ejemplos y explicaciones claras. Ya que no se sabe el nivel de conocimiento que tendrá la persona que vea o use nuestro programa, para darle la mayor usabilidad posible es importante documentar. Además de clarificar lo que se quiso expresar en el código, para entender tu propio código en el futuro, encontrar partes claves del código, también se puede comentar las decisiones sobre el diseño. Además de explicar las interfaces y su ejemplo de uso, efectos de usar ciertos códigos, anécdotas sobre reutilización de código o código que al final no se ha usado y recordar las reasignaciones de variables.

2. Describe dos ejemplo de comentarios importantes y necesarios?

Los comentarios más importantes que han de usarse son los que proporcionan mayor cantidad de información útil posible, concretos y cercanos a la realidad explicando:

- Cómo usar cierta pieza de código
- Por qué existe en el sistema
- Qué efectos tiene usarlas.

3. Describe dos ejemplos de comentarios/documentación no importante/necesaria

- Los comentarios sobre la implementación que describen qué estas haciendo, como hacer una división entre dos números, esto cualquier usuario puede entenderlo sin explicación por su obviedad y es un comentario que estorba ya que en vez de proporcionar información extra son una carga que hay que mantener y pueden confundir si no son actualizados.
- Los comentarios repetidos, cuando repites bloques de código que sean necesario usar en varias funciones y repitas el comentario explicando su uso, al igual que comentar el nombre completo de una variable abreviada, a veces es mejor usar un nombre suficientemente declarativo como nombre de variable que comentar el nombre de una variable abreviada.