Modelo: classe donde vamos a mapear (que tiene equivalente en base de datos) las apis nos lo ponen fàcil a nivel de entry points.

MODELOS: /TODO, /USER /POST /COMMENTS

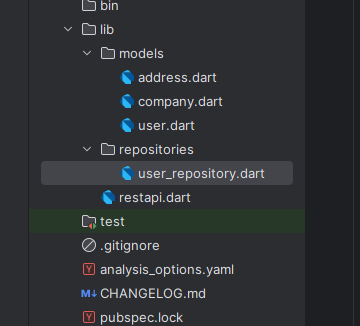
Para cada modelo creamos repositorio: serà una classe. En caso de todos serà todoRepository().

Repository: ayuda a ir a buscar. Todo repository, post repository y user repository. (tendra estos)

En modelos van las clases.

Lo que es maquina es sincrono, cuandop se van a buscar datos de afuera como webs / apis se hacen fx\* (fx\*: funciones) asincronas.

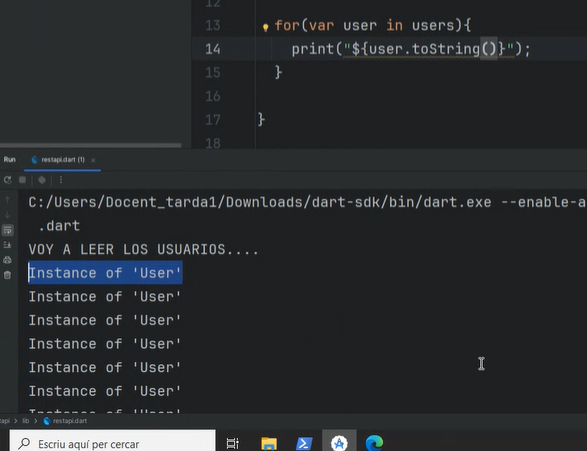
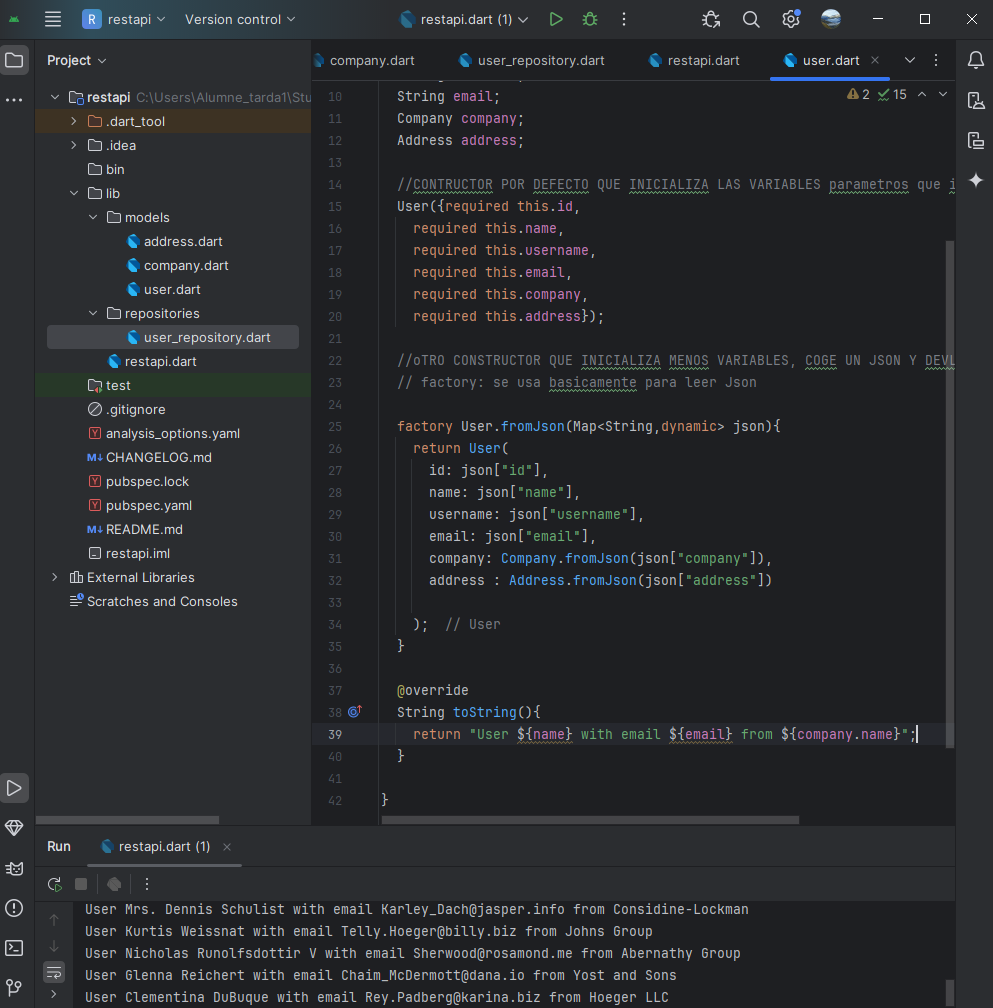
Pondremos los fetch en **repositories**, otra carpeta como models, dentro de **lib**.

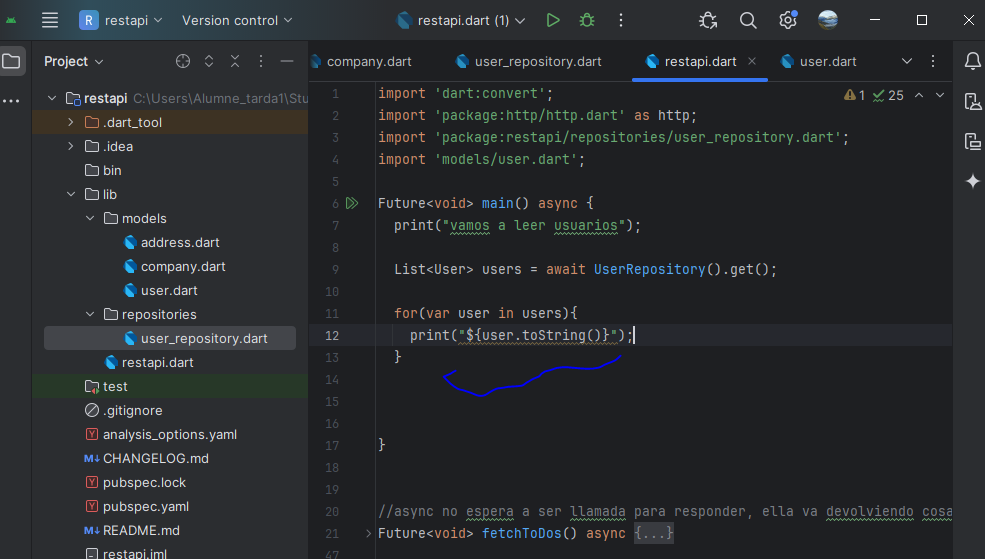


Repositorios tendran que tenir opciones poara crear , modificar,m eliminar ...

Constant es variable q ha de ser inicialitzada i no es pot mofificar en tot el codi.  
Finals: dins de funcions nomes inicialitzada una vegada.

Const esta inicialitzada fora de funcions.

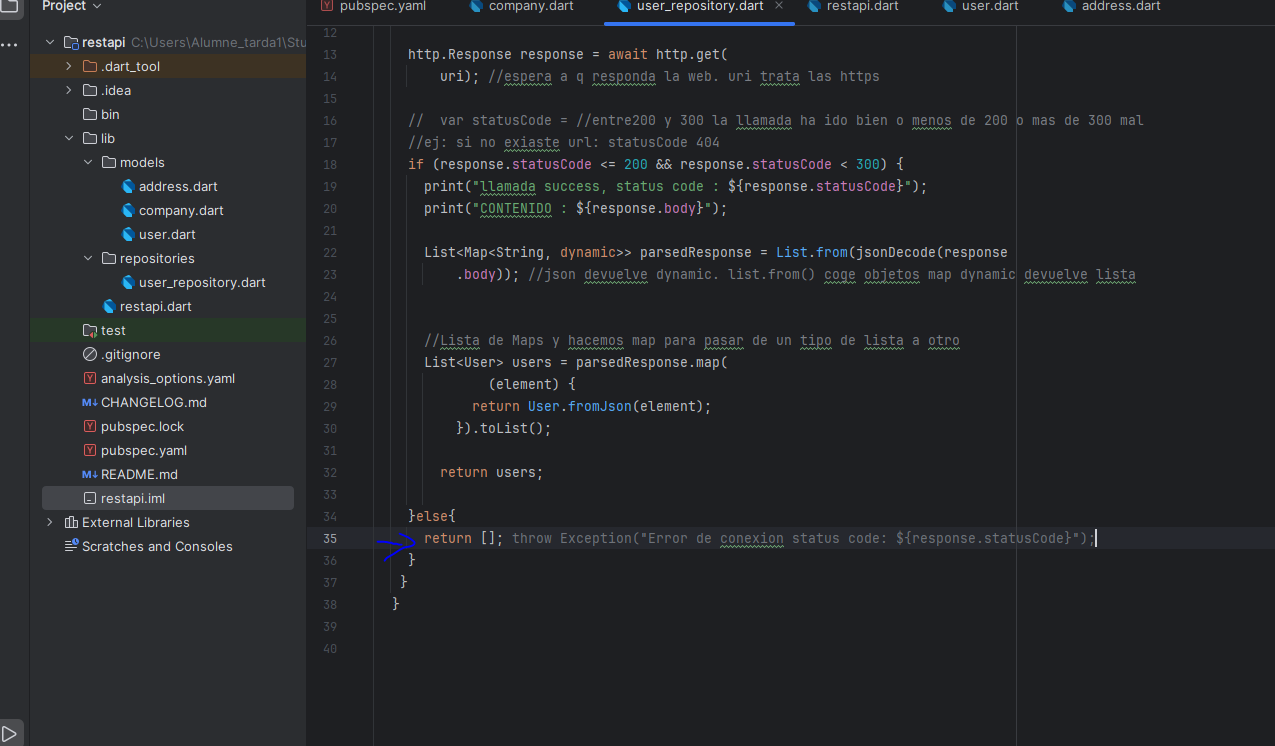
.toStrin() modificado con @override reescribiendo la classe toString(). Si nmo ponemos override nos devolveria esto: instanco of User.  
  
aquí hacemos override de tostring:  


Se llama desde main:  


Para devolver future list user user o list user es algo que se gestiona interna,mente de flutter.

Una variable Dentro de un scope no se puede usar fuera del scope, scope son las llaves {}.

for(var user in users){  
 print("${user.toString()}"); //aqui no hace falta interpolar con ${} a no ser que vaya dentro de texto.  
}

en vez de retorn lista vacia hacemos lanzar el error throw exception.

las excepciones van de abajo a arriba.

Si hay varios catch se queda en el catch mas cercano hacia arriba.

Rethrow : excepcion catpturada se relanza para arriba. Cuando se captura yn catch cuando hay error se ejecuta. Si queremos que sea otro catch hay que hacer rethrow.

Firebase o json para bases de datos: no se necessita backend ya que tendremos la base de datos como si estuviera en el movil, con una libreria flutter firebase.(no se necessita api, es libreria)

Les clases repositori no tenen constructor. No s ha de declarar constructor.

jsonDecode

parsea Json que es Map string dynamic>

int.parse(“43”) devuelve 43 un int.

Haremos  
int.tryParse(“43”) devuelve entero parseado o null si no es numero.