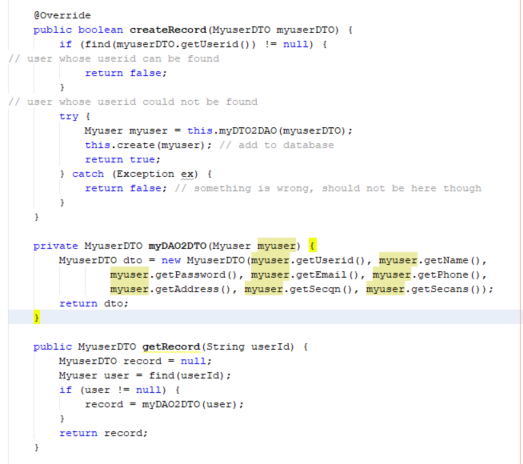
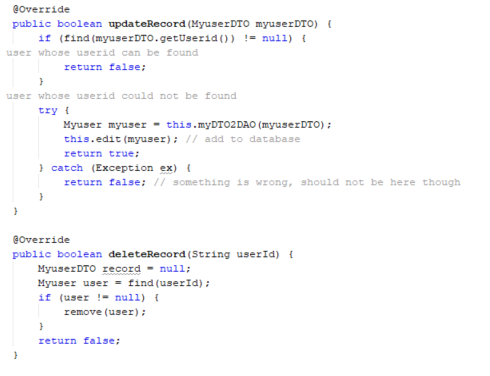
Name: Ashen Lakshitha Fonseka

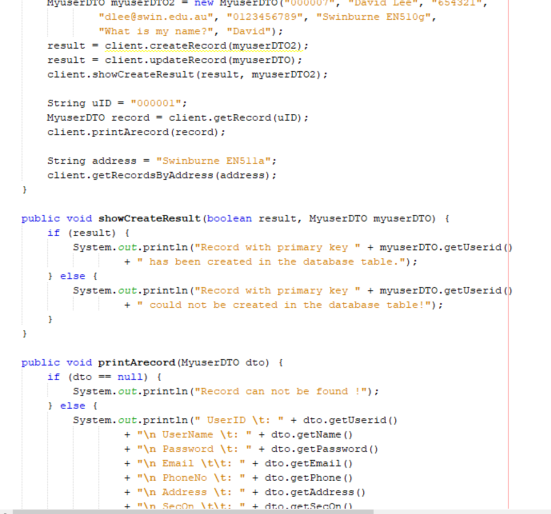
Student ID: 102238586

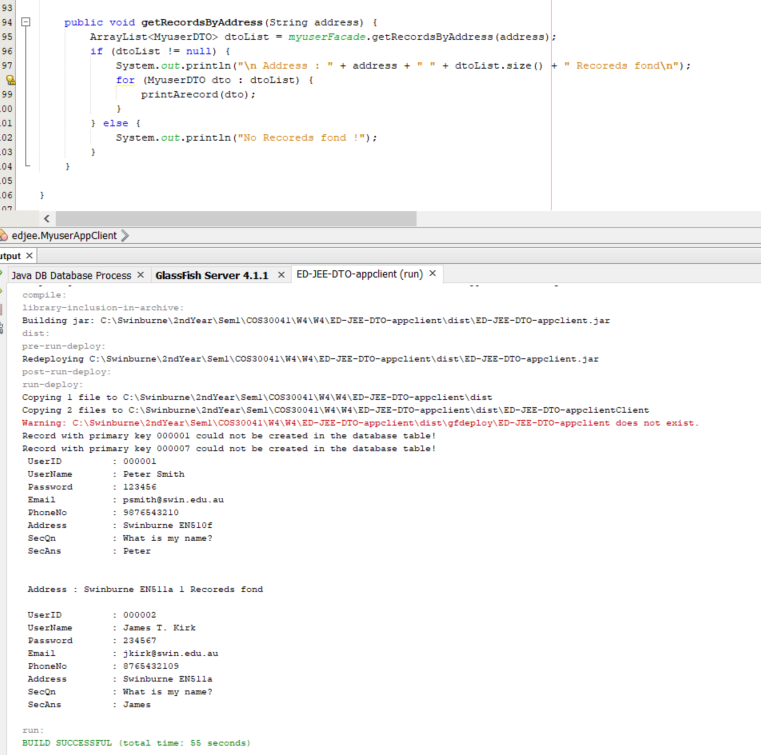
Tutor: Shibli Saleheen











Myuser class

Myuser is a DAO type object. Which means it directly access to database when the object

created. MyuserFacade is a bean type class. This makes myuser responsible for ORM work.

4.2 Stateless session beans are pooled. This means that, for each managed bean, the container

keeps a certain number of bean instances ready in a pool. For each client request, an

instance from the pool is quickly assigned to service the request. When the client finishes,

the instance is returned to the pool for later reuse. This setup using a pool means that a

small number of bean instances can service a relatively large number of clients

Myuser class

Myuser is a DAO type object. Which means it directly access to database when the object

created. MyuserFacade is a bean type class. This makes myuser responsible for ORM work.

4.2 Stateless session beans are pooled. This means that, for each managed bean, the container

keeps a certain number of bean instances ready in a pool. For each client request, an

instance from the pool is quickly assigned to service the request. When the client finishes,

the instance is returned to the pool for later reuse. This setup using a pool means that a

small number of bean instances can service a relatively large number of clients

Myuser class

Myuser is a DAO type object. Which means it directly access to database when the object

created. MyuserFacade is a bean type class. This makes myuser responsible for ORM work.

Myuser is a DAO type object. Which means it directly access to database when the object created. MyuserFacade is a bean type class. This makes responsible for ORM work.

Stateless session beans are pooled. This means that, for each managed bean, the container keeps a certain number of beans instances ready in a pool. For each client request, an instances from the pool is quickly assigned to service the request. When the client finishes, the instance is returned to the pool for later reuse. This setup using a pool means that a small number of bean instances can service a relatively large number of clients.