**Final Ruby project 3 - Text follow-up**

Update the crud.rb file to turn it into a module, completed file below:

* require 'bundler/inline'
* gemfile true do
* source 'http://rubygems.org'
* gem 'bcrypt'
* end
* module Crud
* require 'bcrypt'
* puts "Module CRUD activated"
* def Crud.create\_hash\_digest(password)
* BCrypt::Password.create(password)
* end
* def Crud.verify\_hash\_digest(password)
* BCrypt::Password.new(password)
* end
* def Crud.create\_secure\_users(list\_of\_users)
* list\_of\_users.each do |user\_record|
* user\_record[:password] = create\_hash\_digest(user\_record[:password])
* end
* list\_of\_users
* end
* def Crud.authenticate\_user(username, password, list\_of\_users)
* list\_of\_users.each do |user\_record|
* if user\_record[:username] == username && verify\_hash\_digest(user\_record[:password]) == password
* return user\_record
* end
* end
* "Credentials were not correct"
* end
* end

In front of all the method names we used a class reference, you can replace this or use self. in their place, that is how you reference class methods. The code in the main.rb file would still work if you use self.create\_secure\_users for example. This is a good opportunity for you to read up on the difference between "class methods" and "instance methods". We will see how to convert these into instance methods in the next video by using the module as a mixin for our student object.

We then updated the main.rb file (or wherever you want to use the module from):

* require\_relative 'crud'
* users = [
* { username: "mashrur", password: "password1" },
* { username: "jack", password: "password2" },
* { username: "arya", password: "password3" },
* { username: "jonshow", password: "password4" },
* { username: "heisenberg", password: "password5" }
* ]
* hashed\_users = Crud.create\_secure\_users(users)
* puts hashed\_users