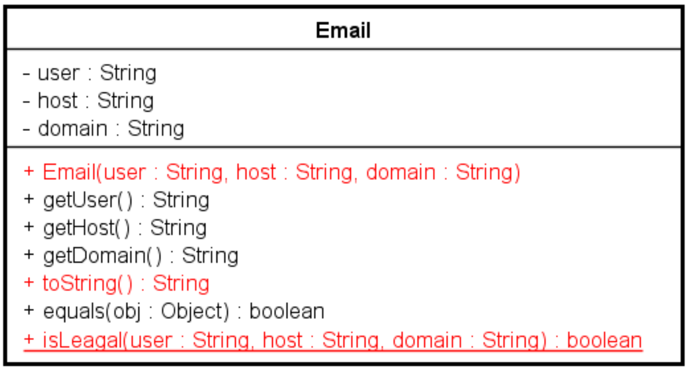
Exercise: Email, version 2



Email

- user : String
- host : String
- domain : String

+ Email(user : String, host : String, domain : String)
+ getUser() : String
+ getHost() : String
+ getDomain() : String
+ toString() : String
+ equals(obj : Object) : boolean
+ isLeagal(user : String, host : String, domain : String) : boolean

Create a new module, name it `Email_v2` and copy class `Email` (from Emal_v1) into the module. (If you did not make Email_v1, then just implement all methods shown in the class diagram)

Background:
An email address contains the three case insensitive parts: `user`, `host` and `domain` and the string representation is given as: `user@host.domain`

A legal email address (in this version) has the following restrictions:

- The `user` part has at least one character and no more than 64 characters
- The `user` part may contain letters (upper- and lowercase) and digits
- The `host` part has at least one character and no more than 63 characters
- The `host` part may contain letters (upper- and lowercase) and digits
- The first character in the `host` part has to be a letter – not a digit
- The `domain` part has at least one character and no more than 63 characters
- The `domain` part may contain letters (upper- and lowercase) and digits
- The `domain` part has at least one letter – cannot be all digits

Modify class `Email`:

a) Create a class method (a static method) `isLegal` with three `String` parameters, `user`, `host` and `domain`. The method returns `true` if the arguments represents a legal email address and returns `false` if any of the rules presented above are violated, i.e. if the email address is illegal.
Hint: To check each characters of a string, you may use a loop from the first character (index 0) to the last character (the string length -1) and use the String method `charAt(index)`.
Note that the following condition checks if a char variable (`ch`) is an uppercase letter:

```
      if (ch >= 'A' && ch <= 'Z')
```

b) Update the three-argument constructor to call the static method to validate the input. Initialise the email to `"wrong@email.address"` if the email address is illegal.

c) Modify method `toString()` such that it returns ”`Wrong format`” if the email address is illegal.

Modify the test application named `EmailTest` such that you test all the paths in your code checking for an illegal email.