



Τμήμα Μηχανικών Η/Υ & Πληροφορικής  
Πανεπιστήμιο Πατρών

Οντοκεντρικός Προγραμματισμός

Σύστημα κοινωνικής δικτύωσης (Social network)

ΑΚΑΔΗΜΑΙΚΟ ΕΤΟΣ: 2018-2019

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΓΙΑΝΝΟΥΛΑ ΔΗΜΗΤΡΟΥΚΑ

A.M.: 1043770

### This:

Μέσα σε ένα στιγμιότυπο μεθόδου ή constructor, το this κάνει αναφορά στο τρέχον αντικείμενο, το αντικείμενο του constructor ή της μεθόδου που καλείται. Μπορούμε να ανατρέξουμε σε οποιοδήποτε μεταβλητή ή μέλος του αντικειμένου.

### Κώδικας(από τη κλάση ReplyMessage )

...

```
public ReplyMessage(String message_text, String message_creator,int likes)
```

```
{  this.message_text = message_text;
    this.message_creator = message_creator;
    this.likes=likes; }
```

...

### Static:

Γίνεται χρήση της static σε πολλές μεθόδους μέσα στις κλάσεις του project, όπως (*public static ArrayList<Users> UsersInitialazation()*),δηλαδή στην αρχικοποίηση του πίνακα με τους χρήστες), έτσι ώστε να μπορώ κάνω επίκληση στις μεθόδους αυτές, χωρίς να χρειαστεί να δημιουργήσω κάποιο στιγμιότυπο κλάσης.

Μια στατική μέθοδος μπορεί να έχει πρόσβαση σε στατικά δεδομένα και μπορεί να αλλάξει τη τιμή τους, αλλά δεν μπορεί να χρησιμοποιήσει μη στατικά δεδομένα ή να καλέσει μη στατική μέθοδο απευθείας. Το this και το super δε μπορούν να χρησιμοποιηθούν σε στατικό περιβάλλον.

### Κώδικας(από τη κλάση Main)

...

```
//Method to search if a member with this name and this password exists
```

```
public static String Login(String username,String password, ArrayList<Users> Userslist)
```

```
{
    String login_username="NULL";
```

```
for(Users i : Userslist){
```

```
    if((i.getFirstname().equals(username)) &&
(i.getPassword().equals(password)))
```

```

        {
            login_username=i.getFirstname();
        }
    }//for end
    return login_username;
} //Login function end
...

```

### Iterator:

Χρησιμοποιώντας Iterator, μπορούμε να εκτελέσουμε τις λειτουργίες της ανάγνωσης και αφαίρεσης (remove), δίνοντας μας τη δυνατότητα να περνάτε μέσα από μια συλλογή(λίστα). Το list iterator επιτρέπει την αμφίδρομη μετακίνηση μιας λίστας και την τροποποίηση των στοιχείων.

### Κώδικας(από τη κλάση LoginUserMenu)

```

...
Iterator<Users> it = UsersList.iterator();
    while (it.hasNext()) {
        if (it.next().getFirstname().equals(username)) {
            it.remove();
            System.out.print("User :" + username + "Deleted
Successfully!\n");
        }
    }
...

```

### Getters and setters:

Δηλώνοντας τις μεταβλητές στις κλάσεις ως `private`, μπορώ να έχω πρόσβαση σε αυτές μόνο από την ίδια την κλάση που έχουν οριστεί.

Χρησιμοποιώντας, όμως `public` μεθόδους `set` και `get` μπορώ να έχω πρόσβαση και να χρησιμοποιήσω αυτές τις μεταβλητές και από άλλες κλάσεις, οπουδήποτε τις χρειάζομαι.

### Κώδικας(από τη κλάση *User*)

...

```
//get firstname
public String getFirstname()
{ return firstname }

//set firstname
public void setFirstname(String firstname)
{ this.firstname = firstname }

//get lastname
public String getLaststname()
{ return lastname; }

//set lastname
public void setLaststname(String lastname)
{ this.lastname = lastname; }
```

...

### toString:

Μπορούμε να χρησιμοποιήσουμε τη μέθοδο toString () για να πάρουμε συμβολοσειρά αντικειμένου.

### Κώδικας(από τη κλάση User)

...

```
public String toString() {  
    return "Name:" + this.firstname      + "\n"  
    + "Lastname:" + this.lastname + "\n"  
    + "E-mail :"  + this.email      + "\n"  
    + "password :" + this.password  + "\n"  
    + "ID="       + this.userID     + "\n" ;  
}
```

...

## Διάγραμμα κλάσεων (UML)



