

algo : login()

rôle : permettre à un utilisateur de s'identifier et d'être reconnu par le logiciel (est ce un client ou un technicien)

entrée :

- saisieMail, chaine de caractere
- saisieMDP, chaine de caractere

sortie :

- message d'erreur en cas d'échec d'identification et réinitialisation des zone de saisie
- fermeture fenetre login et affichage fenêtre correspondant au type d'utilisateur (fenêtre client pour un client, fenêtre technicien pour un technicien)

variables :

- mailClient, liste de chaine de caractere
- mailTechnicien, liste de chaine de caractere
- nbMailClient, entier
- nbMailTechnicien, entier
- mdpMail, chaine de caractere
- saisieMail, chaine de caractere
- saisieMDP, chaine de caractere
- resultat, boleen
- indice, entier

debut :

event clique sur valider :

```
mailClient <- requeteSQL(SELECT email FROM client)
nbMailClient <- longueur(mailClient)
mailTechnicien <- requeteSQL(SELECT email FROM technicien)
nbMailTechnicien <- longueur(mailTechnicien)
saisieMail <- contenu zone de saisie email
saisieMDP <- contenu zone de saisie MDP
resultat <- 1
indice <- 0
```

```
tant que indice < nbMailClient ET resultat, faire :
```

```
  si saisieMail = mailClient[indice], alors faire :
```

```
    mdpMail <- requeteSQL(SELECT MDP, WHERE email = mailClient,
FROM Client)
```

```
    si saisieMDP = mdpMail, alors faire :
```

```
      fermer fenetre login
```

```
      lancer fenetre client
```

```
      resultat <- 1
```