

## 3 - Les effets de bord

---

Tous les programmes ne font pas que des opérations mathématiques et toutes les fonctions ne sont pas pures.

```
function createUser(firstname, lastname) {  
  let user = {  
    firstname: firstname,  
    lastname: lastname,  
  };  
  
  result = db.insert("users", user);  
  return result;  
}
```

ici, la fonction dépend d'une opération extérieure d'insertion d'un utilisateur dans ce qu'on imagine être une base de données.  
comment vérifier que l'utilisateur a bien été inséré et qu'il n'y a pas eu d'erreur en cours ?

```
let result = db.insert("users", user);  
return result;
```

qui est sûr que `db` est bien une connexion à une vraie base de données ?  
on s'offre la possibilité de `mock` la fonction insert d'un potentiel objet "db".

```
class DBMock {  
    insert(table, object) {  
        return true;  
    };  
}
```

On peut mettre à jour notre fonction

```
function createUser(db, firstname, lastname) {  
    let user = {  
        firstname: firstname,  
        lastname: lastname,  
    };  
  
    result = db.insert("users", user);  
    return result;  
}
```

```
testEqual(true, createUser(new DBMock, "Victor", "Hugo"));
```

on ne cherche pas à valider l'écriture en base de données qui est un effet de bord

on cherche à valider en dehors de l'effet de bord:

est-ce que le prénom est supérieur à 3 caractères ?

est-ce que le nom est supérieur à 3 caractères ?

---

créer une fonction createUser prenant comme paramètres firstname,

lastname et qui insère un dictionnaire dans une base de données

créer un mock de fonction de base de données

intégrer un test de createUser en utilisant un mock de fonction db.insert

