**model.people.login info** (2/23/14)

/\*

Server should be the place to maintain playerList[] and waitingList[]

in return 'updateuser' event, it should also pass the type\_list info

so the completeLogin() can update playerList[] or waitingList[], and set the type\_list

before publish 'ibet\_login' event.

\*/

var

**playerList = [] , waitingList = [];**

**isUser = false, type\_list = NO\_CHANGE;**

**current\_name**;

**login = function ( name ) {**

var sio = isFakeData ? ibet.fake.mockSio : ibet.data.getSio();

/\* check name in playerList = [] , waitingList = [] \*/

isUser = **ibet.util.userCheck**(user, playerList);

if( isUser) {

type\_list = IS\_PLAYERS;

$.gevent.publish( 'ibet-login', user.name, type\_list );

return;

}

isUser = **ibet.util.userCheck**(user, waitingList);

if( isUser) {

type\_list = IS\_WAITING;

$.gevent.publish( 'ibet-login', user.name, type\_list );

return;

}

// to use in completeLogin()

**current\_name = name**;

/\* if user is not on these two lists then create the new user \*/

stateMap.user = makePerson({

cid : makeCid(),

css\_map : {top : 25, left : 25, 'background-color':'#8f8'},

name : name

});

sio.on( 'userupdate', completeLogin );

sio.emit( 'adduser', {

cid : stateMap.user.cid,

css\_map : stateMap.user.css\_map,

name : stateMap.user.name

});

}; //end of people.login()

logout = function () {

var user = stateMap.user;

// when we add chat, we should leave the chatroom here

stateMap.user = stateMap.anon\_user;

clearPeopleDb();

$.gevent.publish( 'ibet-logout', [ user ] );

};

return {

get\_by\_cid : get\_by\_cid,

get\_db : get\_db,

get\_user : get\_user,

login : login,

logout : logout

};

---------------------------------------------------------

**In util.js** ( L17 util)

Purpose: to check the existence of user object against a list of user object

check on 'name' property

userCheck = function ( user, userList) {

// to use it

**ibet.util.userCheck**(user, userlist) => T/F