

---

Subject: get\_part\_name, chathooks and strings  
Posted by [reborn](#) on Sun, 11 Jan 2009 14:25:30 GMT  
[View Forum Message](#) <> [Reply to Message](#)

---

I wanted to make it easier to get the `GameObject *` of a player by typing his name out. Then I found it was slightly annoying when players had stupid names like "11||1!||\1\!||\11\11||", so I experimented with "Get\_Part\_Name". However, the function never seems to return a gameobject no matter how unique I start typing there name...

Here is a little chat hook I made to test the function out:

```
class testChatCommand : public ChatCommandClass {
void Triggered(int ID,const TokenClass &Text,int ChatType) {
    GameObject *obj = Get_GameObj(ID);

    if (!Text[1].empty()) {

std::string player = Text[1].c_str();

GameObject *pobj = Get_GameObj_By_Player_Name(player.c_str());

if(!pobj){
Console_Input(StrFormat("ppage %d You typed %s, there is no exact match to this
name.",ID,player.c_str()).c_str());
}
else{
Console_Input(StrFormat("ppage %d You typed %s, i've found this player and I confirm his name
is %s.",ID,player.c_str(),Get_Player_Name(pobj)).c_str());
}
GameObject *pobj2;
pobj2 = Get_Part_Name(player.c_str());
if(!pobj2){
Console_Input(StrFormat("ppage %d You typed %s, the name is not unique enough to
find.",ID,player.c_str()).c_str());
}
else{
Console_Input(StrFormat("ppage %d You typed %s, you probably mean
%s.",ID,player.c_str(),Get_Player_Name(pobj2)).c_str());
}
}
};
ChatCommandRegistrant<testChatCommand>
testChatCommandReg("!find",CHATTYPE_ALL,1,GAMEMODE_AOW);
```

I used this chathook in a server with twenty players, I typed out a few players names that was

long, but just missed off the last letter. I was always paged that the name was not unique enough.

Is there something wrong with the way I am using the function `Get_Part_Name`, or is the function itself not working how it should?

`Get_Part_Name` function is here if you can't be bothered to look it up

```
GameObject *Get_Part_Name(const char *name1)
{
    GenericSLNode *x = BaseGameObjList->HeadNode;
    int count = 0;
    GameObject *current = 0;
    while (x)
    {
        GameObject *o = As_SoldierGameObj((GameObject *)x->NodeData);
        if (o)
        {
            const char *name = Get_Player_Name(o);
            if (!stristr(name,name1))
            {
                current = o;
                count++;
            }
            delete[] name;
        }
        x = x->NodeNext;
    }
    if ((count == 1) && (current) && (Commands->Get_ID(current)))
    {
        return current;
    }
    else
    {
        return 0;
    }
}
```

---