```
const wchar_t *Get_Wide_Player_Name_By_ID(int ID) {
 cPlayer *x = Find_Player(ID);
 if (!x) {
  return L"None";
 }
 return x->PlayerName;
}
```

For things that are stored as wide chars internally, like player names and translated strings, it's always better to use the wide version of their function. This produces cleaner, slightly more efficient code that doesn't have a chance of leaking memory if you forget to delete it.

If you need them in a char for some reason, such as comparison, you can convert them like this:

StringClass Name = Get_Wide_Player_Name_By_ID(ID);

or

StringClass(Get_Wide_Player_Name_By_ID(ID))

And StringClass will automatically handle the conversion and memory allocation.