

# **World of Warcraft Database**

**Created By**

**Paul McCusker**

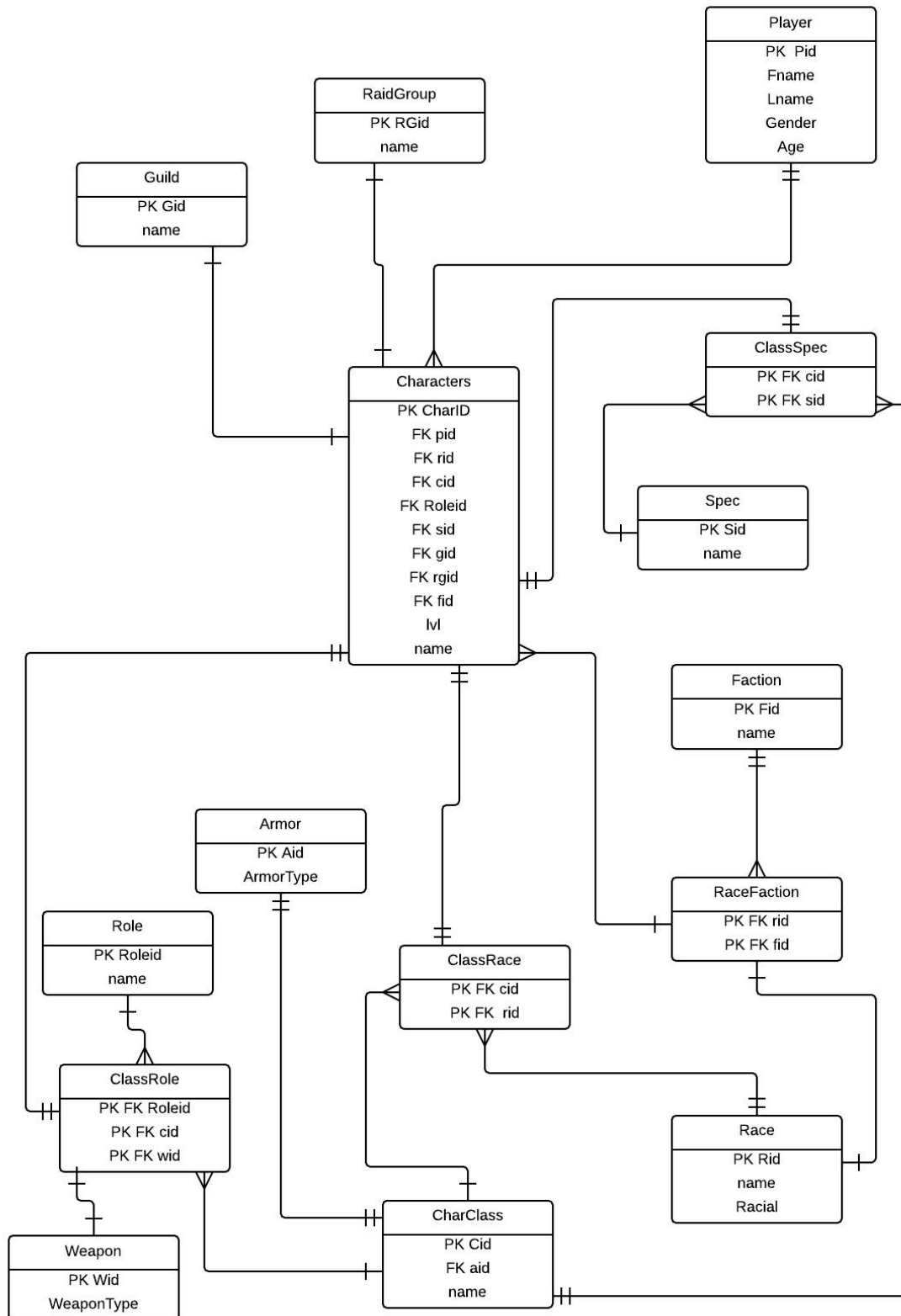
## Table of Contents

<b>Executive Summary</b> .....	3
<b>ER Diagram</b> .....	4
<b>Create Statement – Player</b> .....	5
<b>Create Statement – Armor</b> .....	6
<b>Create Statement – Spec</b> .....	7
<b>Create Statement-RaidGroup</b> .....	8
<b>Create Statement - Guild</b> .....	9
<b>Create Statement - CharClass</b> .....	10
<b>Create Statement - Race</b> .....	11
<b>Create Statement - Weapons</b> .....	12
<b>Create Statement - Factions</b> .....	13
<b>Create Statement - Roles</b> .....	14
<b>Create Statement - ClassRole</b> .....	15
<b>Create Statement - ClassSpec</b> .....	16
<b>Create Statement - ClassRace</b> .....	17
<b>Create Statement - ClassFaction</b> .....	18
<b>Create Statement - Character</b> .....	19
<b>Views – Character Profiles</b> .....	20
<b>Views – Guild Members</b> .....	21
<b>Stored Procedure - Faction</b> .....	22
<b>Trigger - Faction</b> .....	23
<b>Security – WOW_DB_ADMIN</b> .....	24
<b>Security – WOW_DB_USER</b> .....	26
<b>Implementation</b> .....	28
<b>Known Problems</b> .....	28
<b>Future Additions</b> .....	28

## **Executive Summary**

This Database is created to keep track of all character created in the world largest MMORPG World of Warcraft (WoW). The database stores all key information regarding a players characters these stats include : name, race, faction, class, specialization , role, gender, guild, raid group. The goal of this database is to make it easy to be able to quickly catagories characters based on common skills, abilities, roles, or other defining characteristics. With over seven million subscribers WoW needs a functioning database that houses vital character information with ease and simplicity, the idea that any character that is created can be easily entered into this database and populated with all information to complete a full character profile

# ER Diagram



## Create Statement – Player

The Players Table' role is to collect information about the player such as name, gender, and age.

Drop Table if exists Players

```
create table Players (  
    pid varchar(10) NOT NULL ,  
    fname text NOT NULL,  
    lname text NOT NULL,  
    gender text NOT NULL,  
    age int NOT NULL,  
    PRIMARY KEY (pid)  
);
```

Dependencies:

Pid>>fname,lname,gender,age

Output pane				
Data Output Explain Messages History				
	fname text	lname text	gender text	age integer
1	Paul	McCusker	Male	20
2	Bryan	Rockwood	Male	19
3	Mason	Crane	Male	20
4	Mark	McCusker	Male	18
5	Emma	Stone	Male	24
6	Mila	Kunis	Female	30

## Create Statement – Armor

The Armors Table' role is to house all types of armor so that they can be easily called from other tables as well as cross referenced for compatibility.

```
create table Armors (  
    aid varchar(10) NOT NULL ,  
    armorType text NOT NULL,  
    PRIMARY KEY (aid)  
);
```

Dependencies :  
aid>>armorType

Output pane		
Data Output Explain Mess		
	aid characte	armortype text
1	a001	Cloth
2	a002	Chain Mail
3	a003	Leather
4	a004	Plate

## Create Statement – Spec

The Specs Table' role is to house all types of specs so that they can be easily called from other tables as well as cross referenced for compatibility.

```
create table Spec (
    sid varchar(10) NOT NULL ,
    name text NOT NULL,
    PRIMARY KEY (sid)
);
```

Dependencies:

Sid>>name

Data Output	Explain	Messages	History
	sid character varying(10)	name text	
1	s001	Frost	
2	s002	Fire	
3	s003	Arcane	
4	s004	Holy	
5	s005	Retribution	
6	s006	Protection	
7	s007	Arms	
8	s008	Fury	
9	s009	Protection	
10	s010	Blood	
11	s011	Frost	
12	s012	Unholy	

## Create Statement-RaidGroup

The RaidGroups Table' role is to house all the different raid groups so that characters can be added in as members.

```
create table RaidGroup (  
    rgid varchar(10) NOT NULL ,  
    name text NOT NULL,  
    PRIMARY KEY(rgid)  
);
```

Dependencies:

Rgid>>>name

Output pane		
Data Output Explain Messages History		
	rgid character	name text
1	rg001	Tuesday Late Night Core
2	rg002	Core group 3
3	rg003	Monday Mid Day
4	rg004	AddictionX Speed Runs



## Create Statement - Guild

The Guilds Table' role is to house all the different Guilds so that characters can be added in as members.

```
create table Guild (  
    gid varchar(10) NOT NULL ,  
    name text NOT NULL,  
    PRIMARY KEY(gid)  
);
```

Dependencies:

Gid>>name

Output pane		
Data Output		Explain
		Message
	gid characte	name text
1	g001	C H O S E N
2	g002	MidWinter
3	g003	Method
4	g004	Blood Legion
5	g005	Milk and Honey
6	g006	Honorable Kills

## Create Statement - CharClass

The CharClass Table' role is to house all the different classes in the game so that a character can easily be assigned a class.

```
create table CharClass (  
    cid varchar(10) NOT NULL ,  
    name text NOT NULL,  
    PRIMARY KEY (cid)  
);
```

Dependencies:

Cid>>name

Output pane		
Data Output Explain Message:		
	cid character \	name text
1	c001	Death Knight
2	c002	Palidan
3	c003	Warrior
4	c004	Mage
5	c005	Warlock
6	c006	Priest
7	c007	Druid
8	c008	Rogue
9	c009	Monk
10	c010	Shaman
11	c011	Hunter

## Create Statement - Race

The Race Table' role is to house all the different Race in the game so that a character can easily be assigned a race.

```
create table Race (
    rid varchar(10) NOT NULL ,
    name text NOT NULL,
    racial text NOT NULL,
    PRIMARY KEY (rid)
);
```

Dependencies:

Rid>>name,racial

Output pane			
Data Output Explain Messages History			
	rid character varying(10)	name text	racial text
1	r001	Troll	Beserking
2	r002	Blood Elf	Arcane Torrent
3	r003	Orc	Enrage
4	r004	Goblin	Rocket Jump
5	r005	Undead	Will of the Forsaken
6	r006	Worgen	Dark Flight
7	r007	Pandarian	Quaking Palm
8	r008	Human	Every man for Themselves
9	r009	Night Elf	Shadow Meld
10	r010	Dranei	Healing Thingy
11	r011	Nome	Escape Artist
12	r012	Dwarf	Stoneskin
13	r013	Tauren	War Stomp

## Create Statement - Weapons

The Weapon Table' role is to house all the different weapons that any given character can use. The wid is also used to reference what weapons classes can and cant use.

```
create table Weapon (  
    wid varchar(10) NOT NULL ,  
    weapontype text NOT NULL,  
    PRIMARY KEY (wid)  
);
```

Dependencies:

Wid>>weapontype

Output pane		
Data Output		
	wid	weapontype
	characte	text
4	w004	2h Mace
5	w005	2h Sword
6	w006	Polearm
7	w007	Staff
8	w008	2 1h Axe2
9	w009	2h Axe
10	w010	2 Fist Weapons

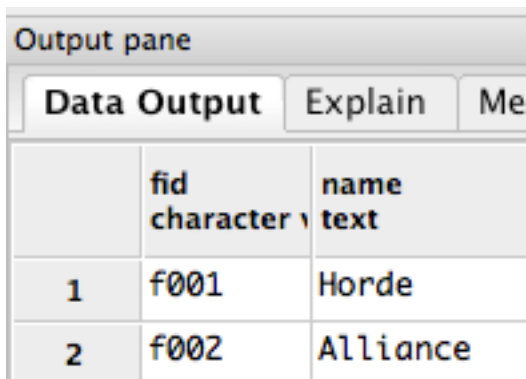
## Create Statement - Factions

The Factions Table' role is to assign characters to the two factions Horde or Alliance.

```
create table Faction (  
    fid varchar(10) NOT NULL ,  
    name text NOT NULL,  
    PRIMARY KEY (fid)  
);
```

Dependencies:

Fid>>name



The screenshot shows a database interface with an 'Output pane' at the top. Below the pane title are three tabs: 'Data Output' (which is selected), 'Explain', and 'Me'. The 'Data Output' tab displays a table with three columns: an index column, 'fid character', and 'name text'. The table contains two rows of data: row 1 with fid 'f001' and name 'Horde', and row 2 with fid 'f002' and name 'Alliance'.

	fid character	name text
1	f001	Horde
2	f002	Alliance

## Create Statement - Roles

The Roles Table' role is to assign characters to the one of three roles that they will be using in battle.

```
create table Role (  
    roleid varchar(10) NOT NULL,  
    name text NOT NULL,  
    PRIMARY KEY (roleid)  
);
```

Dependencies:

Roleid>>name

Output pane		
Data Output Explain M		
	roleid character var	name text
1	role001	DPS
2	role002	Tank
3	role003	Healer

## Create Statement - ClassRole

The ClassRoles Table' role is to check to make sure that class can only be assigned the correct roles for example priests cant tank. It also shows what weapon that role uses based on its class.

```
create table ClassRole (
    roleid varchar(10) NOT NULL references
Role(roleid),
    cid varchar(10) NOT NULL references
CharClass(cid),
    wid varchar(10) NOT NULL references
Weapon(wid),
PRIMARY KEY (roleid,cid,wid)
);
```

Dependencies:

None

Output pane			
Data Output		Explain	Messages
	name text	weapontype text	name text
6	DPS	2h Sword	Warrior
7	Tank	2 1h Axe2	Warrior
8	DPS	Staff	Mage
9	DPS	Staff	Warlock
10	DPS	Staff	Priest
11	Healer	Staff	Priest
12	DPS	Polearm	Druid
13	Tank	Polearm	Druid
14	Healer	Staff	Druid
15	DPS	2 Daggers	Rogue
16	DPS	2 1h Maces	Monk

## Create Statement - ClassSpec

The ClassSpec Table' role is to check to make sure that assigns specs to the correct class id.

```
create table ClassSpec (  
    cid varchar(10) NOT NULL references CharClass(cid),  
    sid varchar(10) NOT NULL references Spec(sid),  
    PRIMARY KEY (cid,sid)  
);
```

Dependencies:

None

Output pane		
Data Output Explain Messages		
	name text	name text
3	Arcane	Mage
4	Holy	Palidan
5	Retribution	Palidan
6	Protection	Palidan
7	Arms	Warrior
8	Fury	Warrior
9	Protection	Warrior
10	Blood	Death Knight



## Create Statement - ClassRace

The ClassRace Table' role is to check to make sure that the correct cid is eligible with the race selected.

```
create table ClassRace (  
    cid varchar(10) NOT NULL references CharClass(cid),  
    rid varchar(10) NOT NULL references Race(rid),  
    PRIMARY KEY (cid,rid)  
);
```

Dependencies:

None

Output pane		
Data Output Explain Messages History		
	name text	name text
46	Troll	Warlock
47	Blood Elf	Warlock
48	Orc	Warlock
49	Goblin	Warlock
50	Undead	Warlock
51	Worgen	Warlock
52	Human	Warlock
53	Nome	Warlock
54	Dwarf	Warlock

## Create Statement - ClassFaction

The ClassFaction Table' role is to check to make sure that the correct race is eligible with the faction selected.

```
create table RaceFaction (  
    rid varchar(10) NOT NULL references Race(rid),  
    fid varchar(10) NOT NULL references Faction(fid),  
    PRIMARY KEY (rid,fid)  
);
```

Dependencies:

None

Output pane		
Data Output		
	name text	name text
1	Troll	Horde
2	Blood Elf	Horde
3	Orc	Horde
4	Goblin	Horde
5	Undead	Horde
6	Pandarian	Horde
7	Tauren	Horde
8	Worgen	Alliance
9	Pandarian	Alliance

## Create Statement - Character

The Character Table' role is to house all information about all players characters. Easily the most important table in the database the character table holds all the vital information for creating the character profiles.

```
create table Characters (
    charID varchar(10) NOT NULL ,
    pid varchar(10) NOT NULL references Player(pid),
    rid varchar(10) NOT NULL references Race(rid),
    cid varchar(10) NOT NULL references CharClass(cid),
    roleID varchar(10) NOT NULL references Role(roleID),
    sid varchar(10) NOT NULL references Spec(sid),
    gid varchar(10) references Guild(gid),
    rgid varchar(10) references RaidGroup(rgid),
    name text NOT NULL,
    fid varchar(10) NOT NULL references Faction(fid),
    lvl int NOT NULL,
    PRIMARY KEY (charID)
);
```

Dependencies:

charID>>name,lvl

Output pane								
Data Output		Explain	Messages	History				
	name text	fname text	name text	name text	name text	name text	name text	lvl integer
1	Arabel	Paul	Troll	Tank	C H O S E N	Tuesday Late Night Core	Horde	90
2	Dawn	Paul	Tauren	Tank	MidWinter	Core group 3	Horde	90
3	AddictionX	Paul	Blood Elf	DPS	MidWinter	Tuesday Late Night Core	Horde	90
4	Orgata	Paul	Undead	DPS	C H O S E N	AddictionX Speed Runs	Horde	90
5	Genesis	Paul	Troll	Healer	C H O S E N	Core group 3	Horde	90
6	DwayneTickels	Bryan	Nome	Healer	Blood Legion	Monday Mid Day	Alliance	80
7	PingPang	Bryan	Pandarian	Tank	MidWinter	Tuesday Late Night Core	Horde	90
8	Taurgus	Bryan	Troll	DPS	C H O S E N	Tuesday Late Night Core	Horde	90
9	MyBrand	Bryan	Blood Elf	DPS	C H O S E N	Core group 3	Horde	90
10	Aqualad	Bryan	Human	Tank	Blood Legion	Monday Mid Day	Alliance	90

## **Views – Character Profiles**

This view is a good way to show all the information about a character. It is very helpful to quickly find out information such as what Guild the current character is in as well as raid group and role that they are playing.

Create View Character\_Profiles AS

Select Distinct

Characters.name,Role.name,Guild.name,RaidGroup.name,

Characters.lvl

From Characters,Role,Guild,RaidGroup

Where Characters.roleid = Role.roleid

and Characters.gid = Guild.gid

and Characters.rgid = RaidGroup.rgid

## Views – Guild Members

This view is a good way to show the members of a specific guild this would help guild leaders easily see how many members they have as well as what classes there is an abundance of.

Create View Guild\_Profile AS

Select Distinct Characters.name, CharClass.name, Characters.lvl

From Characters,CharClass,Guild

Where Characters.cid =CharClass.cid

and Characters.gid = Guild.gid

and Guild.name = '.....';

Example:

Create View Guild\_Profile AS

Select Distinct Characters.name, CharClass.name, Characters.lvl

From Characters,CharClass,Guild

Where Characters.cid =CharClass.cid

and Characters.gid = Guild.gid

and Guild.name = 'C H O S E N';

Output pane			
Data Output		Explain	Messages
	name text	name text	lvl integ
1	Taurgus	Shaman	90
2	Arabel	Druid	90
3	Genesis	Priest	90
4	Orgata	Mage	90
5	MyBrand	Mage	90

## **Stored Procedure - Faction**

Stored procedure checks whenever you update a character it checks to make sure that the race is on the correct faction. So that you don't have Human characters on the horde as well as not having undead on the Alliance.

```
CREATE FUNCTION ValidCharacterFaction() returns trigger as  
$$
```

```
BEGIN IF ( rid = 'r001' || rid = 'r002' || rid = 'r003' || rid = 'r004' ||  
rid = 'r005' || rid = 'r013') THEN
```

```
update Faction set fid = 'f001';
```

```
ELSE IF( rid = 'r006' || rid = 'r007' || rid = 'r008' || rid = 'r009' || rid =  
'r010' || rid = 'r011' || rid = 'r012') THEN
```

```
update Faction set fid = 'f001';END if;
```

```
END
```

```
$$LANGUAGE plpgsql;
```

## **Trigger - Faction**

This trigger is used to check the Factions column of the Characters table to make sure that the correct races are being played on each faction.

Create Trigger Valid\_Faction After Insert or Update

On Faction For Each row of Characters

Execute Procedure ValidCharacterFaction();

## **Security – WoW\_DB\_ADMIN**

Security to grant clearance to different parts of the database.

```
CREATE USER WoW_DB_ADMIN WITH PASSWORD  
'ADMIN';
```

```
Revoke all on Player From WoW_DB_ADMIN;
```

```
Revoke all on Armor From WoW_DB_ADMIN;
```

```
Revoke all on Spec From WoW_DB_ADMIN;
```

```
Revoke all on RaidGroup From WoW_DB_ADMIN;
```

```
Revoke all on Guild From WoW_DB_ADMIN;
```

```
Revoke all on CharClass From WoW_DB_ADMIN;
```

```
Revoke all on Race From WoW_DB_ADMIN;
```

```
Revoke all on Weapon From WoW_DB_ADMIN;
```

```
Revoke all on Faction From WoW_DB_ADMIN;
```

```
Revoke all on Role From WoW_DB_ADMIN;
```

```
Revoke all on ClassRole From WoW_DB_ADMIN;
```

```
Revoke all on ClassSpec From WoW_DB_ADMIN;
```

```
Revoke all on ClassRace From WoW_DB_ADMIN;
```

```
Revoke all on RaceFaction From WoW_DB_ADMIN;
```

```
Revoke all on Characters From WoW_DB_ADMIN;
```

```
Grant Insert, Update, Select, Delete on Player To WoW_DB_ADMIN;
```

```
Grant Insert, Update, Select, Delete on Armor To WoW_DB_ADMIN;
```

```
Grant Insert, Update, Select, Delete on Spec To WoW_DB_ADMIN;
```



Grant Insert, Update, Select, Delete on RaidGroup To WoW\_DB\_ADMIN;

Grant Insert, Update, Select, Delete on Guild To WoW\_DB\_ADMIN;

Grant Insert, Update, Select, Delete on CharClass To WoW\_DB\_ADMIN;

Grant Insert, Update, Select, Delete on Race To WoW\_DB\_ADMIN;

Grant Insert, Update, Select, Delete on Weapon To WoW\_DB\_ADMIN;

Grant Insert, Update, Select, Delete on Faction To WoW\_DB\_ADMIN;

Grant Insert, Update, Select, Delete on Role To WoW\_DB\_ADMIN;

Grant Insert, Update, Select, Delete on ClassRole To WoW\_DB\_ADMIN;

Grant Insert, Update, Select, Delete on ClassSpec To WoW\_DB\_ADMIN;

Grant Insert, Update, Select, Delete on ClassRace To WoW\_DB\_ADMIN;

Grant Insert, Update, Select, Delete on RaceFaction To WoW\_DB\_ADMIN;

Grant Insert, Update, Select, Delete on Characters To WoW\_DB\_ADMIN;

## **Security – WoW\_DB\_USER**

Security to grant clearance to different parts of the database.

```
CREATE USER WoW_DB_USER WITH PASSWORD 'USER';
```

```
Revoke all on Player From WoW_DB_USER;
```

```
Revoke all on Armor From WoW_DB_USER;
```

```
Revoke all on Spec From WoW_DB_USER;
```

```
Revoke all on RaidGroup From WoW_DB_USER;
```

```
Revoke all on Guild From WoW_DB_USER;
```

```
Revoke all on CharClass From WoW_DB_USER;
```

```
Revoke all on Race From WoW_DB_USER;
```

```
Revoke all on Weapon From WoW_DB_USER;
```

```
Revoke all on Faction From WoW_DB_USER;
```

```
Revoke all on Role From WoW_DB_USER;
```

```
Revoke all on ClassRole From WoW_DB_USER;
```

```
Revoke all on ClassSpec From WoW_DB_USER;
```

```
Revoke all on ClassRace From WoW_DB_USER;
```

```
Revoke all on RaceFaction From WoW_DB_USER;
```

```
Revoke all on Characters From WoW_DB_ADMIN;
```

```
Grant Select on Player To WoW_DB_USER;
```

```
Grant Select on Armor To WoW_DB_USER;
```

```
Grant Select on Spec To WoW_DB_USER;
```

```
Grant Select on RaidGroup To WoW_DB_USER;  
Grant Select on Guild To WoW_DB_USER;  
Grant Select on CharClass To WoW_DB_USER;  
Grant Select on Race To WoW_DB_USER;  
Grant Select on Weapon To WoW_DB_USER;  
Grant Select on Faction To WoW_DB_USER;  
Grant Select on Role To WoW_DB_USER;  
Grant Select on ClassRole To WoW_DB_USER;  
Grant Select on ClassSpec To WoW_DB_USER;  
Grant Select on ClassRace To WoW_DB_USER;  
Grant Select on RaceFaction To WoW_DB_USER;  
Grant Select on Characters To WoW_DB_USER;
```

## **Implementation**

Currently the World of Warcraft Database is in its alpha form. There will be more than enough room for future expansion. There are constantly new characters being added to the game, so it is not very practical to implement during the active server time. This means in order to implement this database it would be advantageous to wait for a point when a new Expansion Pack (i.e Mists of Pandaria) comes out. This way there would be plenty of server down time for the database to be populated and be ready to have new characters added to it.

## **Known Problems**

World of Warcraft is an ever changing game. This means that setting a database in stone causes problems. As of right now all functionalities of the database work smoothly, this does not mean that with future expansion there will not be errors. For example a error that is foreseeable in the future is specializations. In the last expansion pack Blizzard re worked many of the classes "specs" adding and taking some away this would throw off all of the sid in the Spec table leading to problems in multiple other tables.

## **Future Additions**

With the implementation of more and more content in World of Warcraft there is always new possibilities for storing data. Things such as GearScore a system to evaluate a players skill level would be a great addition to the characters table. Other ideas include players PVP status and Tittles.