Subject: Custom Brenbot Plugin Error
Posted by Gen_Blacky on Sat, 06 Dec 2008 00:47:59 GMT
View Forum Message <> Reply to Message

I have this plugin that someone made for me that uses these commands. This is not used for serverside. Its pulls sound from a database that are all in the always.dat and i get this error (at bottom)

```
  <command name="snda">
    <permission level="3"/>
    <syntax value="!snda &lt;soundfile.wav&gt;"/>
    <help value="Plays a sound heard by all players. "/>
    <enabled value="1"/>
    <hideInHelp value="0"/>

    <group>irc_admin</group>

  </command>

 <command name="sndp">
  <permission level="3"/>
  <syntax value="!sndp &lt;user&gt; &lt;soundfile.wav&gt;"/>
  <help value="Plays a sound heard (only) by user."/>
  <enabled value="1"/>
  <hideInHelp value="1"/>
  <group>irc_admin</group>
 </command>


 <command name="sndt">
  <permission level="3"/>
  <syntax value="!sndp &lt;team&gt; &lt;soundfile.wav&gt;"/>
  <help value="Plays a sound heard (only) by team."/>
  <enabled value="1"/>
  <hideInHelp value="1"/>
  <group>irc_admin</group>
 </command>
```

DBD::SQLite2::st execute failed: no such table: mp3s(1) at dbdimp.c line 412 at brdatabase.pm line 63.

I would just like to know what the error means

Subject: Re: Custom Brenbot Plugin Error
Posted by danpaul88 on Sat, 06 Dec 2008 01:28:06 GMT

It means there's no table called mp3's in the brenbot.dat database, obviously it requires some custom setup. All official plugins will setup tables on their own when required.

Also, you do know BRenBot 1.52 *already* has commands for snda, sndp and sndt don't you?

```
<!--  Sound Commands

  Commands for sending sounds to players in the server

  -->
<command name="snda">
  <permission level="3" />
  <syntax value="!snda <soundfile.wav>" />
  <help value="Plays a sound heard by all players. Filenames of soundfiles in always.dat should
be used." />
  <enabled value="1" />
  <hideInHelp value="1" />
  <group>irc_admin</group>
  <module>bhs</module>
  <alias>broadcast</alias>
  </command>
<command name="sndp">
  <permission level="3" />
  <syntax value="!sndp <user> <soundfile.wav>" />
  <help value="Plays a sound heard (only) by user. Filenames of soundfiles in always.dat should
be used." />
  <enabled value="1" />
  <hideInHelp value="1" />
  <group>irc_admin</group>
  <module>bhs</module>
  </command>
<command name="sndt">
  <permission level="3" />
  <syntax value="!snda <team> <soundfile.wav>" />
  <help value="Plays a sound heard by entire team. Filenames of soundfiles in always.dat should
be used. Team Nod or 0, Team GDI or 1." />
  <enabled value="1" />
  <hideInHelp value="1" />
  <group>irc_admin</group>
  <module>bhs</module>
  </command>
```

EDIT: Changed quote to code so indenting does not get fucked up

I created a table in brenbot.dat called mp3s and the bot gave me a error say the table already exists here is the plugin code.

```
package zbot;

use POE;

use plugin;

#use Win32::Process::List;




 our %additional_events = (
                           "snda" => "snda",
                           "sndp" => "sndp",
                           "sndt" => "sndt",
                           );




sub start
{
 my ( $session, $heap, $args ) = @_[ SESSION, HEAP, ARG0 ];
 my $kernel = $_[KERNEL];

 my $name = shift;

 my @result = brdatabase::execute_query ( "SELECT * FROM mp3s" );
      my $size = @result;

 if ($size<1)
 {

  my @result = brdatabase::execute_query ( "CREATE TABLE mp3s ( length TEXT,name TEXT,time TEXT)");
  $data_file="plugins/mp3s.pl";
          open(DAT, $data_file) || die("Could not open $data_file to read!");
          @raw_data=<DAT>;
          close(DAT);
          foreach(@raw_data)
```

```perl
        {
          my $st = $_;
          my $st =~ s/\"/""/g;
          my @ar = split(",",$st);
          my $length= $ar[0];
          my $name= $ar[1];
          my $time = $ar[2];
          my $q = "INSERT INTO mp3s(length,name,time) VALUES
(\"".$length."\",\"".$name."\",\"".$time."\")";
          my @res = brdatabase::execute_query($q);
        }
 }



}

sub sndp
{
 my %args = %{@_[ ARG0 ]};


 eval
 {
  if (!$args{arg1} || !$args{arg2})
  {
   my $syntaxvalue = $args{settings}->{syntax}->{value};

   if ($args{nicktype} == 1)
   {
    brIRC::ircmsg ( "[RT] Usage: $syntaxvalue", $args{'ircChannelCode'} );
   }
   else
   {
    modules::RenRemCMD("msg [RT] Usage: $syntaxvalue");
   }

   return;
  }


  my $pager = $args{nick};


  my ( $result, %player ) = playerData::getPlayerData( $args{'arg1'} );
  if ( $result == 1 )
  {
```

```perl
                    $snd = $args{arg2};
                    $query = "SELECT * FROM mp3s WHERE name LIKE \"%".$snd."%\"";

                    my @array = brdatabase::execute_query($query);
                    my $size = @array;
                    if($size == 1){
                          $f = $array[0];
                          my $sndname = $f->{'name'};
                          my $sndtime = $f->{'time'};
                          modules::RenRemCMD("sndp $player{id} $sndname");
                          brIRC::ircmsg ( " 9PrivateSound sent to $player{name} --> \($pager):
$sndname", $args{'ircChannelCode'} );

                          return;
                    }

                    if($size < 10)
                    {
                          brIRC::ircmsg ("There are $size sounds with $snd in name, listing them
now",$args{'ircChannelCode'} );
                          foreach(@array)
                          {
                              my $sndname = $_->{'name'};
                              my $sndtime = $_->{'time'};
                              brIRC::ircmsg ("$sndname : time:
".$sndtime,$args{'ircChannelCode'} );
                          }
                          return;
                    }
                    if($size > 9)
                     {
                          brIRC::ircmsg ("There are $size sounds with $snd in name, listing only
10 of them now",$args{'ircChannelCode'} );
                          my $cn = 0;
                          while($cn < 10)
                          {
                              my $f = $array[$cn];
                              my $sndname = $f->{'name'};
                              my $sndtime = $f->{'time'};
                              brIRC::ircmsg ("$sndname : time:
".$sndtime,$args{'ircChannelCode'} );
                              $cn++;
                          }
                          return;
                     }

          }
    else
```

```perl
    {
    brIRC::ircmsg ( "Error: $args{arg1} was not found ingame, or is not unique.",
$args{'ircChannelCode'} );
    }

 }
 or modules::display_error($@);
}



sub snda
{
 my ( $session, $heap, $args ) = @_[ SESSION, HEAP, ARG0 ];
 my $kernel = $_[KERNEL];
 my %args = %{$args};
 print("=zbot= In snda with arg $args{arg1}\r\n");
 eval
 {
 if (!$args{arg1})
 {
  if ($args{nicktype} == 1)
  {
   my $syntaxvalue = $args{settings}->{syntax}->{value};
   brIRC::ircmsg ( "Usage: $syntaxvalue", $args{'ircChannelCode'} );
  }
  else
  {
   modules::RenRemCMD("msg [RT] Usage: $args{settings}->{syntax}->{value}");
  }
  return;
 }


  my $pager = $args{nick};
  my $snd   = $args{arg1};


                $query = "SELECT * FROM mp3s WHERE name LIKE \"%".$snd."%\"";

                my @array = brdatabase::execute_query($query);
                my $size = @array;

             if ($args{nicktype} == 1)
  {

                    if($size == 1){
                        $f = $array[0];
```

```perl
                        my $sndname = $f->{'name'};
                        my $sndtime = $f->{'time'};
                        modules::RenRemCMD("snda $sndname");
                        brIRC::ircmsg ( " 9Global Sound --> \($pager): $sndname",
$args{'ircChannelCode'} );
                        return;
                }

                if($size < 10)
                {
                        brIRC::ircmsg ("There are $size sounds with $snd in name, listing
them now",$args{'ircChannelCode'} );
                        foreach(@array)
                        {
                                my $sndname = $_->{'name'};
                                my $sndtime = $_->{'time'};
                                brIRC::ircmsg ("$sndname : time:
".$sndtime,$args{'ircChannelCode'} );
                        }
                        return;
                }
                if($size > 9)
                {
                        brIRC::ircmsg ("There are $size sounds with $snd in name, listing only
10 of them now",$args{'ircChannelCode'} );
                        my $cn = 0;
                        while($cn < 10)
                        {
                         my $f = $array[$cn];
                                my $sndname = $f->{'name'};
                                my $sndtime = $f->{'time'};
                                brIRC::ircmsg ("$sndname : time:
".$sndtime,$args{'ircChannelCode'} );
                                $cn++;
                        }

                        return;
                }
        }else{
                #ingame page
                my ( $result, %player ) = plugin::getPlayerData( $pager );
                my $id = $player{'id'};

                if($size == 1){
                        $f = $array[0];
                        my $sndname = $f->{'name'};
                        my $sndtime = $f->{'time'};
                        modules::RenRemCMD("snda $sndname");
```

```perl
                    plugin::RenRemCMD ( "ppage $player{'id'}  9Global Sound --> \($pager):
$sndname" );
                        return;
                }

                if($size < 10)
                {
                        plugin::RenRemCMD ( "ppage $player{'id'} There are $size sounds
with $snd in name, listing them now" );
                        foreach(@array)
                        {
                                my $sndname = $_->{'name'};
                                my $sndtime = $_->{'time'};
                              plugin::RenRemCMD ( "ppage $player{'id'} $sndname : time:
".$sndtime );
                        }
                        return;
                }
                if($size > 9)
                {
                        plugin::RenRemCMD ("ppage ".$player{'id'}." There are $size sounds
with $snd in name, listing only 10 of them now");
                        my $cn = 0;
                        while($cn < 10)
                        {
                         my $f = $array[$cn];
                                my $sndname = $f->{'name'};
                                my $sndtime = $f->{'time'};
                                plugin::RenRemCMD ( "ppage $player{'id'} $sndname : time:
".$sndtime );

                                $cn++;
                        }

                        return;
                }
        }


  modules::RenRemCMD("snda $args{arg1}");


 }
 or modules::display_error($@);
}

sub sndt
{
 my ( $session, $heap, $args ) = @_[ SESSION, HEAP, ARG0 ];
```

```perl
my $kernel = $_[KERNEL];
my %args = %{$args};

eval
{


 if (!$args{arg1} || !$args{arg2})
 {
  if ($args{nicktype} == 1)
  {
   my $syntaxvalue = $args{settings}->{syntax}->{value};
   brIRC::ircmsg ( "Usage: $syntaxvalue", $args{'ircChannelCode'} );
  }
  else
  {
   modules::RenRemCMD("msg [RT] Usage: $args{settings}->{syntax}->{value}");
  }
  return;
 }


 my $pager = $args{nick};

 my $team = $args{arg1};
 my $snd  = $args{arg2};
               $query = "SELECT * FROM mp3s WHERE name LIKE \"%".$snd."%\"";
               my @array = brdatabase::execute_query($query);




               if ($team =~ /nod/i) {$team = "0"} #support user specifying NOD instead of 0
  if ($team =~ /gdi/i) {$team = "1"} #support user specifying GDI instead of 1

               my $size = @array;
               if($size == 1){
                     $f = $array[0];
                     my $sndname = $f->{'name'};
                     my $sndtime = $f->{'time'};
                      modules::RenRemCMD("sndt $team $sndtime");
       brIRC::ircmsg ( " 9TeamSound Sent to Team $team --> \($pager): $sndname",
$args{'ircChannelCode'} );
                return;
                }
```

```
                if($size < 10)
                {
                        brIRC::ircmsg ("There are $size sounds with $snd in name, listing them
now",$args{'ircChannelCode'} );
                        foreach(@array)
                        {
                                my $sndname = $_->{'name'};
                                my $sndtime = $_->{'time'};
                                 brIRC::ircmsg ("$sndname : time:
".$sndtime,$args{'ircChannelCode'} );
                        }
                        return;
                }
                if($size > 9)
                {
                        brIRC::ircmsg ("There are $size sounds with $snd in name, listing only 10
of them now",$args{'ircChannelCode'} );
                        my $cn = 0;
                        while($cn < 10)
                        {
                         my $f = $array[$cn];
                            my $sndname = $f->{'name'};
                                my $sndtime = $f->{'time'};
                                 brIRC::ircmsg ("$sndname : time:
".$sndtime,$args{'ircChannelCode'} );
                                $cn++;
                        }

                        return;
                }

 }
 or modules::display_error($@);
}
```

Edit: it was originally part of a module

---

use "create if not exists" instead? Of course its going to give you an error if you create the same table every time you load the plugin, since the table already exists after the first time its been loaded.

And you still didn't explain why your trying to replicate functionality that already exists...

Subject: Re: Custom Brenbot Plugin Error
Posted by Gen_Blacky on Sun, 07 Dec 2008 05:06:18 GMT
View Forum Message <> Reply to Message

it replicated because i want use a custom database of sounds

Subject: Re: Custom Brenbot Plugin Error
Posted by danpaul88 on Sun, 07 Dec 2008 11:02:30 GMT
View Forum Message <> Reply to Message

The default versions support every sound that exists inside always.dat

---