

Exxy says, "Our game is built in a unique fashion... it isn't anything like DikuMUD or CircleMUD... ROM... or any of the other popular MUD bases out there."

Exxy says, "We have a centralized game engine that we built from scratch."

Exxy says, "The game engine handles very low-level events and manages data and keeps track of connections."

Exxy says, "Other than that, it provides a unique programming language we've developed called EaxiaSEL."

Exxy says, "This is known as "softcode". The softcode allows us to write custom code for events, objects, NPCs, etc. and to modify the way the game actually runs without having to modify the engine."

Exxy says, "This is important because we can make huge changes without having to shut our game down, we can test our live changes immediately, and we can customize almost anything we want... it's near limitless."

Exxy says, "When we want to use this softcode to enhance and customize parts of our game, we do it by writing a script in this language (EaxiaSEL)."

Exxy says, "Scripts are noted by a numeric ID and so objects, rooms, NPCs, and even players will be able to have a script ID assigned to them, if we wanted to customize something."

Exxy says, "Now we've already seen some of this live. For instance, the objects in Character Management System that let you change your ability scores and features are regular objects with scripts attached to them."

Exxy says, "We have four basic data types: objects, rooms, NPCs, and players."

Exxy says, "Each of them have a seperate script type. The language is still EaxiaSEL for each, but the way you write them may be different."

Exxy says, "Player scripts are still under development in the engine, and are thus not supported yet. We will not be touching on them this evening."

Exxy says, "There are also more script types. One example is the 'command' script."

Exxy says, "Emotes and normal actions are handled by command scripts."

Exxy says, "And they're the easiest, so we'll start there."

Exxy says, "When a player (you, I, a player, etc.) types a command, the engine scans through a list of all of the commands a player can type."

Exxy says, "It makes a match against the list to see if the command you type is a command listed and if so, there's a number next to this command that indicates a script ID."

Exxy says, "There is some tricky manipulation that the engine does to check uppercase/lowercase and abbreviated matches, etc, so luckily we don't have to enter in every iteration that the player can try and type."

Exxy says, "Once a match has been made, the script automatically opens up that script file and begins running the script. This is called parsing."

Exxy says, "Say I wanted to create a WAVE command from scratch."

Exxy says, "The first thing I would do is open up my text editor. I prefer notepad, but any editor that will save in straight text format is fine."

Exxy asks, "First we have to think about what kind of command this is?"

Exxy says, "We know that the player will be able to wave at people... they'll be able to wave

objects around... and they'll be able to just wave."

Exxy says, "When a script is run, we are given a few variables that tell us information about who is typing the command and exactly how they typed it."

Exxy says, "This also applies to 'spell' scripts, most 'utility' scripts, and a few other types of scripts."

Exxy says, "For information about how they typed the command, refer to the following list of variables:"

Exxy says, "\$parameter0 this is the command they typed. You will probably not need to check this parameter in this script type because each script file usually only handles one command."

Exxy says, "\$parameter1 this is the first command they typed after the command itself. For instance, if they typed WAVE EXXY, the word "EXXY" would be in this variable."

Exxy says, "And so on with \$parameter2 through \$parameter9, they contain extra parameters the player may have typed."

Exxy says, "If a player has not typed in anything for a parameter, then it will be blank. For example, if I typed in "WAVE" without any parameters, \$parameter1 will be equal to "" (nothing). If I typed in "WAVE EXXY", then \$parameter1 will be "EXXY" and \$parameter2 will be "" (nothing)."

Tess asks, "so if i type WAVE AT EXXY, "AT" is parameter1 and "EXXY" becomes parameter2?"

Exxy says, "Yes, so if I typed "WAVE AT EXXY", my parameter list would be: \$parameter0="WAVE" \$parameter1="AT" \$parameter2="EXXY"

Exxy says, "There are also two more variables that will allow you to check what the player has typed in."

Exxy says, "That is \$line and \$subline."

Exxy says, "\$line is the combination of everything the player typed after the initial command."

Exxy says, "If I typed "WAVE AT BANANA PERSON", \$line would be equal to "AT BANANA PERSON"."

Exxy says, "This is important to use for REPORT and THINK commands."

Exxy says, "So you can grab all of the text they are saying and use it all at once."

Exxy says, "\$subline works the same way, but begins at the second word after the command they initially typed."

Exxy says, "In our example, \$subline="BANANA PERSON"."

Exxy says, "This is important for commands like WHISPER and SEND where the first parameter needs to be the recipient and the rest of the text needs to be used."

Exxy says, "So, for our command we wish to create (WAVE), we need to divide the script into two basic sections."

Exxy says, "The first section is for no parameters. If they type WAVE by itself."

Exxy says, "The second section is for one (or more) parameters."

Exxy says, "We code this by using the IF command."

Exxy says, "If you are familiar with either C/C++, perl, or any type of BASIC (including Visual Basic), you're in luck... the IF command supports multiple ways of expressing what you mean."

Exxy says, "For this example, we'd want to check to see if \$parameter1 has anything in it. If it does, then we need to default to our second section... otherwise, we're in our first section."

Exxy says, "We code this by forming our IF statement like this:"

```
IF ($parameter1 = "")
```

Exxy says, "Which means: 'if the first parameter is empty'."

Exxy says, "To add code to your script that happens only if this statement is true (the first parameter being empty), we need an open brace and a close brace."

Exxy says, "These are like parenthesis. They let us know that everything between the open brace and the close brace should be parsed only if the IF statement is true."

Exxy says, "IF statements are also called conditions."

Exxy says, "So say I typed "WAVE" without any parameters."

Exxy says, "Our expression would result in TRUE because \$parameter1 is empty."

Exxy says, "These braces need to be on their own line."

Exxy says, "I'm picky about format, so I've made the engine only accept scripts written so that the braces are on their own line in your script."

Exxy says, "So for the IF statement, it'd look like this:"

```
IF ($parameter1 = "")
{
}
```

Exxy says, "Now, in between these braces (as earlier mentioned), we will write code that will only be parsed if that condition is true."

Exxy says, "So if I only "WAVE", then that code should go in here."

Exxy says, "Since this is a simple emote command, the only thing that happens is messaging gets sent to the player and the rest of the people in the room."

Exxy says, "I would see "You wave." and the rest of the room would see "Exxy waves"."

Exxy says, "There are several commands to help you with messaging."

Exxy says, "To send a message back to just the player who typed the command, use the MSGP command (MSG for message - P for player)."

Exxy says, "The command would simply look like this:"

```
MSGP You wave.
```

Exxy says, "For everyone else, we'd use MSGE (MSG - E for everyone except player)."

Exxy says, "Note that this is everyone in the room, not everyone in the game."

Exxy says, "For this line, we'll also need to use information about who's typing the command."

Exxy says, "We want the message to change based on who's typing it."

Exxy says, "If I type it, we want it to say "Exxy waves..."

Exxy says, "If Keris is typing it, we want it to say "Keris waves..."

Exxy says, "If Solice is typing it, we want it to say "Smelly waves..."

Exxy says, "So we need information about who's typing the command -- specifically what their name is."

>

Exxy says, "This is contained in the \$ch variable."

Exxy says, "\$ch is a unique type of variable. It has 'subvariables'."

Exxy says, "One subvariable is the player that identifies this character."

Exxy says, "Another one is the room that the character is in."

Exxy says, "The player subvariable is :player"

Exxy says, "So we access it by using the variable construct \$ch:player"

Exxy says, "Player also has subvariables, such as name, race, stats, experience, level, etc."

Exxy says, "The subvariable we want in our WAVE example is name."

Exxy says, "It's called :name."

Exxy says, "So at last, we access it by using the variable construct \$ch:player:name."

Exxy says, "And our messaging command looks like:"

```
MSGE $ch:player:name waves.
```

Rhiann asks, "I know from experience I learn best by having an example... is there a way to could post a sample in docs for us?"

Exxy says, "Actually the WAVE script in its entirety (along with all of our other scripts) are already posted on our FTP site (which I will get into later tonight)."

Exxy says, "So you'll have time to browse all of our scripts at your leisure."

Exxy says, "So back to our script, our first five commands look like this:"

```
IF ($parameter1 = "")
{
    MSGP You wave.
    MSGE $ch:player:name waves.
}
```

Exxy says, "Now some people (and I encourage this) like to indent/tabulate code inside of the braces."

Exxy says, "This format makes code easier to read."

Exxy says, "Now there's several different ways we can handle the second section of our script (the part that handles if they type the WAVE command with a parameter)."

Exxy says, "I'll use the preferred method."

Exxy says, "After our closing brace, on a new line, use the ELSE command."

Exxy says, "The ELSE command pairs with an IF command."

Exxy says, "It essentially means "if the condition from our matching IF command was FALSE, then..."..."

Exxy says, "So we don't actually have to write another IF command that tests to see if there \*is\* information in \$parameter1."

Exxy says, "We will know it by default because we didn't parse code in the first section."

Exxy says, "So this will look like:"

```
ELSE
```

```
{  
}
```

Exxy says, "So they've typed something in after the WAVE command."

Exxy says, "We have to find out what that is."

Exxy asks, "Is it another player? Is it an object? Is it themselves?"

Exxy says, "We have to figure this out and handle it accordingly."

Exxy says, "The first thing I like to check is to see if it's a player."

Exxy says, "To do that, we will use the FIND\_PLAYER EaxiaSEL command."

Exxy says, "FIND\_PLAYER takes 1 parameter, the name of the person you are referring to."

Exxy says, "The target player must be in the same room as \$ch for this to work."

Exxy says, "(we wouldn't want JoeSchmoe waving to me from Cal'Ziz if I'm in Semanri would we?)"

Exxy says, "So in our script, it would look like:"

```
FIND_PERSON $parameter1
```

Exxy says, "We use \$parameter1 because that's the first word they typed after WAVE. We're checking to see if it's a person in the room with them."

Exxy says, "The FIND\_PERSON command will perform this search and return a player into the master variable \$target if a match is found."

Exxy says, "\$target works simularly to \$ch in its subvariable hierarchy."

Exxy says, "So now we need to check this."

Exxy says, "We check \$target:player:name. If FIND\_PERSON found the person they typed in, then we'll have someone's name in \$target:player:name. If \$parameter1 wasn't a person's name or that person's not in the same room with the player, then \$target:player:name will be blank."

Exxy says, "The condition will look like this:"

```
IF ($target:player:name = "")
```

Exxy says, "It is perfectly legal to put one IF statement with its own open/close braces inside another."

Exxy says, "This is called 'nesting'..."

Exxy says, "So for this section, we have the following now:"

```
ELSE  
{  
    FIND_PERSON $parameter1  
    IF ($target:player:name = "")  
    {  
    }  
}
```

---UNEDITED FROM THIS POINT ON---

Exxy says, "Now. here's a good example of ELSE\_IF, a new command."

>

Exxy says, "ELSE\_IF lets you test a new condition as if you were using IF."

Exxy says, "The difference is that ELSE\_IF will only be tested if the matching IF condition was FALSE."

>

Maeve nods.

Tess nods.

>

Exxy says, "So we can have IF the target is blank = FALSE, then check to see if the target is the player (they waved at their self), and if that = FALSE, then they must be waving at a player that's not themselves and who is in the same room."

>

Exxy says, "Three seperate sections to an IF statement."

Exxy says, "IF, ELSE\_IF, ELSE."

>

Maeve asks, "can have more than one ELSE\_IF?"

Exxy says, "Note that IF commands do not need an ELSE section. Use an ELSE section if you want something to happen only if the condition is false. Don't use it if nothing happens if the condition is false."

>

Exxy says, "You may have unlimited ELSE\_IFs."

Exxy says, "IF my roses are red, ELSE\_IF my toes are blue, ELSE\_IF Solice's hair is short, ELSE\_IF mountain dew is yummy, etc."

>

Maeve grins at Exxy.

Keris giggles.

>

Keris smooches Exxy on the cheek.

Exxy grins at Keris.

>

Exxy snuggles Keris.

Maeve asks, "are they blue?"

>

Maeve leans on Keris.

Exxy says, "So, now our second section looks like this"

>

Maeve ducks.

Exxy says, "(no)"

>

Exxy says, "ELSE"

Exxy says, "{ "

>

Tess laughs.

Exxy says, "FIND\_PERSON \$parameter1"

>

Exxy says, "IF \$target:player:name = """

```
Exxy says, "{"
>
Exxy says, "}"
Exxy says, "ELSE_IF $ch:player:name = $target:player:name"

>
Exxy says, "{"
Exxy says, "}"

>
Exxy says, "ELSE"
Exxy says, "{"

>
Kzin asks, "Hmm, quick question... Is there a switch/case statement available to replace
multiple elseifs?"
Exxy says, "}"

>
Exxy says, "}"
Exxy says, "Not at this time."

>
Exxy says, "Use IF's."
Maeve says, "you wish"

>
Maeve grins at Kzin.
Tess asks, "whats a switch/cast statement?"

>
Tess says, "not that it matters if we dont have one"
Tess babbles.

>
Maeve says, "don't go there Tess"
Maeve says, "not now"

>
Tess backs up.
Exxy says, "For those of you who aren't familiar, SWITCH is a method used in C/C++, perl,
BASIC/VB, etc that allows you to check multiple events of something in a more orderly
method."

>
Exxy says, "It's known as SELECT CASE in BASIC/VB and SWITCH in C/C++ and perl."
Exxy says, "I find it takes up about the same amount of code space."

>
Exxy says, "We may at one time support it, but there's really no benefit of it."
Kzin nods.

>
Kzin says, "Just used to it is all"
Kzin chuckles.

>
Exxy smiles at Kzin.
Maeve says, "some find it less confusing with alot of nesting"

>
Exxy nods to Maeve.
```

Rhiann nods to Maeve.

>  
Rhiann says, "All those brackets"  
Rhiann giggles.

>  
Exxy says, "Format is definately a plus... but it loses flexibility if you're comparing any "involved" formulas."

>  
Exxy says, "Anyhow, back to EaxiaSEL."  
Tess says, "i am so over my head"

>  
Maeve giggles at Tess.  
Maeve says, "no you'll do fine"

>  
Maeve says, "you catch on quick"  
Kzin nods to Tess.

>  
Exxy says, "Le'ts do the last of the three sections first."  
Tess grins.

>  
Exxy says, "If \$target:player:name is not empty and if \$ch:player:name is not \$target:player:name"

>  
Exxy says, "This section means "if the parameter they typed is a person in the room with the player and the person is not the player"."

>  
Exxy says, "So we want messaging showing that person X is waving to person Y."  
Exxy says, "Three messaging commands are necessary here, because we have to (1) message the player typing command, (2) message the recipient, (3) message everyone else in the room."

>  
Exxy says, "We use good ol familiar MSGP for the player."  
Exxy says, "MSGP You wave to \$target:player:name."

>  
Exxy says, "For the target, we use MSGV (MSG=message - V=victim)."  
Tess giggles.  
Exxy says, "MSGV \$ch:player:name waves to you."

>  
Exxy says, "And finally, for everyone else we use MSGNV (MSG=message - NV=not victim)."  
Tess says, "thats cute"  
Exxy says, "MSGNV \$ch:player:name waves to \$target:player:name."

>  
Exxy says, "So just this third section looks like:"  
Exxy says, "ELSE"

>  
Exxy says, "{ "  
Exxy says, "MSGP You wave to \$target:player:name."

>  
Exxy says, "MSGV \$ch:player:name waves to you."



Exxy says, "MSGNV \$ch:player:name waves to \$target:player:name."

>

Exxy says, "}"

Exxy asks, "Make sense? Questions?"

>

Maeve says, "no wonder I couldn't figure it out"

Tess nods.

>

Tess asks, "did you come up with victim and nonvictim?"

Maeve says, "course he did"

>

Exxy asks, "Beg your pardon, Tess?"

Tess asks, "msgv, thats your code right?"

>

Exxy says, "Yes, this is all our code."

Tess grins.

>

Tess says, "its cute"

Tess says, "thats all"

>

Tess reaches over and pokes Exxy.

Exxy chuckles.

>

Exxy asks, "Why's that?"

Tess grins.

>

Tess says, "you were just the victim of a poke"

Exxy chuckles.

>

Exxy nods.

Exxy says, "Yes I was."

>

Maeve grins at Tess.

Exxy says, "The ELSE\_IF section we'll do next."

>

Tess says, "i'll remember how to do that now"

Exxy says, "This section is parsed only if the player types his or herself as the recipient."

>

Exxy says, "In our example, if I typed "WAVE EXXY"."

Exxy says, "We'd need two messaging commands."

>

Exxy says, "One would be to tell the player that you did in fact wave to yourself."

Exxy says, "The other message would show everyone else that Exxy is waving to himself."

>

Maeve fans herself.

Exxy says, "We've chosen WAVE <self> to mean "So-and-so is fanning his/herself."."

>  
Exxy says, "So... as Maeve has just demonstrated, WAVE <self> shows that messaging."  
Exxy says, "We code this with our popular MSGP command first."

>  
Exxy says, "MSGP You fan yourself."  
Exxy says, "Then we come across a new problem."

>  
Exxy asks, "How do we identify the gender of our player?"  
Tess coughs.

>  
Rhiann says, "\$S"  
Kzin asks, "#Ch:player:sexarticle?"

>  
Rhiann ponders.  
Rhiann giggles.

>  
Exxy says, "We want the message "So-and-so waves him/herself." to show "him" if its a male and "her" if its a female."

>  
Exxy says, "That was a rhetorical question."  
Tess grins at Exxy.

>  
Kzin blushes.  
Exxy glances Kzin.

>  
Rhiann says, "Something besides \$n"  
Kzin says, "/quiet sorry."

>  
Rhiann says, "Something besides I thought it was \$s"  
Maeve grins at Kzin.

>  
Exxy says, "As Kzin has shown, \$ch:player has another subvariable named 'sexarticle'.  
Exxy says, "\$ch:player:sexarticle will equal 'his' for a male and 'her' for female."

>  
Exxy says, "\$s and \$S do not work in EaxiaSEL."  
Exxy says, "They are for room links only."

>  
Exxy says, "They will not work elsewhere."  
Exxy says, "We'll get into that later.."

>  
Exxy says, "We need to check the gender of our player, so we form our IF condition like this:"

>  
Exxy says, "IF \$ch:player:sexarticle = "his"  
Exxy says, "{"

>  
Exxy says, "}"  
Rhiann says, "Ahh that is where i read it..."

```
>
Rhiann digs herself a hole and climbs in.
Exxy says, "If it's a guy, we'll do message one way."

>
Exxy says, "Otherwise:"
Exxy says, "ELSE"

>
Exxy says, "{"
Exxy says, "}"

>
Exxy says, "The command we'll use is our old friend MSGE."
Exxy says, "So now this whole section look like:"

>
Exxy says, "ELSE_IF $ch:player:name = $target:player:name"
Exxy says, "{"

>
Exxy says, "MSGP You fan yourself."
Exxy says, "IF ($ch:player:sexarticle == "his")"

>
Exxy says, "{"
Exxy says, "MSGE $ch:player:name fans himself."

>
Exxy says, "}"
Exxy says, "ELSE"

>
Exxy says, "{"
Exxy says, "MSGE $ch:player:name fans herself."

>
Exxy says, "}"
Exxy says, "}"

>
Exxy says, "... "
Exxy asks, "Everyone okay with that?"

>
Maeve swoons.
Kaze says, "Fairly.."
Maeve nods.

>
Kaze grins.
Exxy says, "Oh, looks like I changed the IF condition a little."

>
Exxy chuckles.
Tess nods.

>
Exxy says, "Sorry about that."
Kaze says, "It'll make perfect sense when I see the full thing in front of me."

>
```

Exxy says, "A double-equal sign is the same as a single equal sign."  
Exxy says, "For C/C++ and perl programmers, you may feel more comfortable with =="  
>  
Exxy says, "The extra parenthesis are more of a C/C++ or perl style as well."  
Exxy says, "Again, either work."  
>  
Kzin nods.  
Kzin smiles.  
Exxy says, "The rest of this command, including how to access objects and run object scripts from a player-command are shown on our FTP."  
>  
Exxy says, "We also have a comprehensive reference manual for EaxiaSEL in the staff-only section of our website."  
>  
Exxy says, "It includes information on how expressions (conditions) work, a full list of variables, and a full list of commands and how they're used."  
>  
Exxy says, "As for our FTP, I need to get everyone's login information."  
Exxy says, "Your user ID will be the first letter of your first name (real name) followed by your whole last name."  
>  
Exxy says, "For me, that's psouza."  
Exxy says, "I would request that each of you send me an Instant Message with your desired password."  
>  
Maeve asks, "mine works fine still want it again?"  
Exxy says, "If you already have a login set up, then don't worry about it."  
>  
Exxy grins.  
Maeve nods.  
>  
Exxy says, "Once I set up your login, I will give you the instructions on how to access files."  
>  
Maeve asks, "can we upload as well or do we need too?"  
Exxy says, "Yes, upload/download"  
>  
Exxy says, "You'll also have full access to your own directory, which includes the ability to delete files and to create/remove subfolders from your personal directory."  
>  
Keris says, "Exxy will be back in a minute"  
nod  
You nod.  
>  
Exxy says, "Everyone take a look at the diamond."  
Natheme howls.  
>  
Exxy says, "And by look, I mean use the "SHOW\_OBJECT" command on it."

Maeve laughs.

>  
Exxy reaches over and pokes Tulsidas.  
show\_object diam  
Exxy reaches over and pokes Maeve.

>  
Object data for a reset diamond:  
Object ID: 773  
Ar/Ds/Nm: 'a' 'reset' 'diamond'  
Script: 151  
Data1: 1 0 0 0 14 0 0 0  
Data2: 0 0 0 0 0 0 0 0

>  
Exxy reaches over and pokes Natheme.

>  
Natheme says, "hey."

>  
Tulsidas says, "ok.. ok"

>  
Tess says, "i already did it"

>  
Natheme says, "I did it."  
Tulsidas says, "i am slloooowww"

>  
Natheme says, "I just looked at it too, for good measure."  
Exxy grins.

>  
Maeve says, "me too"  
Maeve nods.

>  
Exxy says, "The reset diamond has script 151 attached to it."  
Tulsidas says, "big brother sees all"

>  
Tulsidas leans on Exxy.  
Natheme asks, "what's that mean?"

>  
Exxy says, "Now object scripts have two uses."  
Exxy says, "They can #1 handle an event triggered by a character (player) in which it does something customary if a 'verb' is used on it."

>  
Natheme laughs.  
Maeve grins at Exxy.

>  
Exxy says, "For instance, you may want to be able to WAVE a sword in a certain way."  
Exxy says, "The other general use for an object script is to handle an event triggered every so often by the engine itself. This is based on a single repeating timer."

>  
From out of nowhere, Gregarin appears.  
Exxy says, "Used in conjunction with a simple random number and some messaging, this is how

items such as "snake charms" and other objects that do nothing but send out cutesy messages every so often."

>

Maeve nods to Exxy.

Tess taps a writhing serpent draped around the neck.

>

Gregarin waves.

Exxy says, "We need to write our script to determine how our object script is being called first."

>

Tess leans on Gregarin.

Exxy says, "Scripts that are attached to objects will be sent a special value in the \$parameter0 variable indicating what mode the script is being called under."

>

Tulsidas leans on Gregarin.

Gregarin says, "Sorry I wasn't around all weekend. My grandmother died."

>

Tess rubs Gregarin.

Exxy says, "When we test \$parameter0, we're looking for the value 'HANDLE-TIMER-SCRIPT'. If \$parameter0 equals this, we know that our script is being called under the "timer" mode."

>

Exxy says, "Anything else in this variable should be considered as the -other- mode (let's call it the "player event" mode)."

>

Exxy says, "So the beginning of our script looks like this so far:"

Exxy says, "IF (\$parameter0 = "HANDLE-TIMER-SCRIPT")"

>

Exxy says, "{"

Exxy says, "}"

>

Exxy says, "ELSE"

Exxy says, "{"

>

Exxy says, "}"

Exxy asks, "Any questions on that?"

>

Maeve says, "nope"

Tulsidas says, "nah"

>

Exxy smiles.

Natheme asks, "do we need to specify that it looks for Player-Event or just else?"

>

Exxy says, "Just 'else' is fine."

Natheme says, "alright"

>

Gregarin asks, "Can I ask how many modes there are? and, will we learn those?"

Exxy says, "Anything other than 'HANDLE-TIMER-SCRIPT' is going to be the player event mode by default."

>  
Exxy says, "As I said, there's two."  
Natheme says, "oh, ok."

>  
Exxy says, "Object scripts only have two modes."  
Exxy says, "For the purposes of this example, we will build the script that's used behind this reset diamond."

>  
Exxy points at a reset diamond.  
Natheme asks, "just for starters, what does it do?"

>  
Exxy says, "As you can tell, it doesn't --appear- to be doing anything on its own."  
Exxy says, "(getting there, Natheme)"

>  
The night has begun.  
The lightning stops.

>  
Exxy says, "Which means that it doesn't visibly show any reaction to the 'HANDLE-TIMER-SCRIPT' event."

>  
Exxy says, "It's possible that it's doing something that doesn't provide any messaging, but most likely it's doing nothing at all.... we'll I'll reveal the shocking mystery and tell you that it is in fact doing absolutely nothing."

>  
Exxy says, "Which leads me to a command that everyone here should know about: RETURN."  
Natheme asks, "So it's like me, only fancier?"

>  
Exxy says, "RETURN is an EaxiaSEL command that tells the engine to stop parsing the script immediately."

>  
Tulsidas says, "i don't know about RETURN"  
Maeve nods to Natheme.

>  
Tulsidas says, "now i do"  
Natheme nods to Tulsidas.

>  
Natheme says, "Good, Tulsie. Good."  
Exxy says, "RETURN also has an opportunity to pass a value back to the 'caller' of the script."

>  
Maeve asks, "so does return store a value as well?"  
Exxy says, "Just let me finish"

>  
Exxy sighs.  
Natheme says, "heh, poor Exxy."

>  
Exxy says, "Up until now, the engine itself has been the only entity to call scripts."  
Exxy says, "Where 'call' is simply the same as 'instructing the engine to parse a script'."

>

Exxy says, "It's possible for one script to call another script."

Exxy says, "For example, a lot of our commands call a script that checks to see if the person is in a roundtime."

>

Exxy says, "This is an example of a utility script. The script's purpose is to check something or provide messaging that's used in multiple other scripts."

>

Exxy says, "A script gets called, performs some function, and can then optionally pass a value back to the caller. This value is known as the result."

>

Exxy says, "So, for instance, the script that checks for a roundtime right now will return a 0 back to the calling script if there is no roundtime on the player..... or....."

>

Exxy says, "It will return the number of seconds left remaining in the person's roundtime."

Exxy says, "The calling script can check the 'return value' (result) of the script. If the result was 0, then the script can proceed.... they can pick their nose, move north, etc."

>

Exxy says, "If the return value was something else, then the script has the opportunity to tell the user that they need to "...wait XX seconds." and can abort out."

>

Exxy says, "This is how scripts can communicate with each other. It's a little overwhelming now, but it will make more sense as we move on."

>

Exxy says, "If a script has no caller (for example, just the script behind the WAVE command), then you can RETURN any value you want."

>

Exxy says, "Now to tie this back into where we left off..."

Exxy says, "Instead of writing our IF/ELSE statement as above, I could rewrite it like:"

>

Exxy says, "IF (\$parameter0 == "HANDLE-TIMER-SCRIPT")"

Exxy says, "{ "

>

Exxy says, "RETURN 0"

Exxy says, "}"

>

Exxy says, "And that's it."

Exxy says, "Our second section ("player event" mode) code goes right after the last brace."

>

Exxy asks, "Anyone have any questions?"

Tess asks, "umm, can one script handle both modes?"

>

Exxy says, "Yes, that's what we're doing in this script."

Exxy says, "We're handling the 'timer' mode (by doing nothing) and we're handling the 'player event' mode (which we'll add more code to the bottom of the script in a moment)."

>

Tess says, "ok, think i get it"

Exxy smiles.



>  
Natheme says, "wait, I'm confused."  
Exxy smiles at Natheme.

>  
Natheme asks, "ok... uhm... how does the script you just pasted act the same as the If/Else?"

>  
Natheme says, "I don't get it."  
Exxy says, "Okay, think of it this way."

>  
Exxy says, "We have our IF/ELSE again..."  
Exxy says, "If it's timer mode, do this code.      If it isn't, do this code."

>  
Exxy asks, "That's what it says in English, right?"  
Exxy asks, "Natheme?"

>  
Natheme says, "well, yes."  
Exxy says, "Okay."

>  
Exxy says, "Now in the second format, what it says is:"  
Exxy says, "If it's timer mode, do this code and return out of the script.      If we don't parse the code in the following condition, run the rest of the script."

>  
Natheme asks, "oh, so there would be another set of brackets after the ones that enclose return?"

>  
Exxy says, "No."  
Exxy says, "They are no longer necessary."

>  
Natheme says, "oooooh, I get it."  
Tulsidas says, "because of the 0 or because of the Return."

>  
Exxy says, "We don't need to test a condition and we don't need to include them in an ELSE { } section because the only way we can get to that code is if we --didn't-- return out of the script in the IF statement."

>  
Natheme asks, "that script is being called by a script, and if that script is okay, that script goes fine, but if that script is bad, the first stops, right?"

>  
Exxy says, "Eh, something like that"  
Exxy says, "I think you understand."

>  
Exxy smiles.  
Natheme says, "okay. I get it."

>  
Exxy says, "I'll throw one more example into the crowd to make sure we're all on the same page."

>

```
Exxy says, "Say we want to prevent people from WAVEing if they're in a roundtime."  
Exxy says, "Without showing messaging, the easiest way to do this is to check for the  
presense of a roundtime and to abort the script if there is one."  
  
>  
Exxy says, "So we can do it the long way:"  
Exxy says, "IF $ch:player:roundtime > 0"  
  
>  
Exxy says, "{"  
Exxy says, "RETURN 0"  
  
>  
Exxy says, "}"  
Exxy says, "ELSE"  
  
>  
Exxy says, "{"  
Exxy says, "... (rest of our script from last night here)"  
  
>  
Exxy says, "}"  
Exxy says, "... or, we can do it this way (prefered):"  
  
>  
Exxy says, "IF $ch:player:roundtime > 0"  
Exxy says, "{"  
  
>  
Exxy says, "RETURN 0"  
Exxy says, "}"  
  
>  
Exxy says, "... (rest of our script from last night here)"  
Exxy says, "Because the only way we'll get to the rest of our script is if we didn't stop  
the script at the roundtime check."  
  
>  
Exxy exclaims, "Incidentally!!"  
Natheme says, "utoh."  
  
>  
Exxy says, "Our script could have been made much easier if we'd written it like this:"  
Exxy says, "IF $parameter1 = ""  
  
>  
Exxy says, "{"  
Exxy says, "MSGP You wave."  
  
>  
Exxy says, "MSGE $ch:player:name waves."  
Exxy says, "RETURN 0"  
  
>  
Exxy says, "}"  
Exxy says, "And then the rest of our script here. We can completely eliminate the ELSE and  
{ } braces."  
  
>  
Exxy asks, "Does that make sense?"  
Tulsidas asks, "yes, as much as the rest does, so does it, make sense that is, is that  
clear?"
```

>  
Exxy smiles.  
Tess grins.

>  
Tess says, "what he said"  
Tess nods.

>  
Exxy says, "Well good, if you all get it, great."  
Exxy says, "This leads us back to our reset diamond"

>  
Tulsidas exclaims, "i will get it, this I vow!"  
Tess grins at Tulsidas.

>  
Tulsidas says, "ahh that reset diamond.. that is doing nothing"  
Exxy says, "The reset diamond has a RUB verb."

>  
Maeve is staying silent  
Natheme says, "ack"

>  
Exxy says, "It's handled in the 'player event' mode."  
Natheme cries.

>  
Exxy says, "The basis for how this works is this:"  
Exxy says, "(1) you type RUB DIAMOND"

>  
Exxy says, "(2) the engine matches RUB to the command listed in COMMANDS.TXT and runs the associated script."

>  
Exxy says, "(3) the RUB script checks to see if you've typed in a parameter. You have, it's DIAMOND."

>  
Exxy says, "(4) the RUB script checks to see if the parameter is a person in the room with you. Sadly, it is not."

>  
Tulsidas chuckles.  
Exxy exclaims, "(5) the RUB script checks to see if the parameter is an object in the room with you (regardless of its orientation -- in your inventory, on the ground, in a container, etc.). It is!"

>  
Exxy exclaims, "(6) the RUB script checks to see if the object you're referring to has an object script attached to it. It does!"

>  
Exxy says, "(7) the RUB script calls the special object script."  
Gregarin just left.

>  
Exxy says, "(8) the special object script gets told in \$parameter0 what command the player typed. In this example, it was 'RUB', so the script knows to check this and to respond with special messaging."

>

Exxy says, "(9) the special object script returns back the value 1, which means "Yes, I handle the RUB command as a custom event"."

>

Exxy says, "(10) the RUB script begins parsing where it left off and checks the return value of the special object script, which was 1. It knows that 1 means "Okay, the script handled it, so I should do nothing"."

>

Exxy says, "And that's that."

Exxy says, "If our special object script doesn't have a RUB verb (perhaps a WAVE verb? a TAP verb? no verbs?), then the object script would return a 0, which means "Nope, I don't do anything special when I'm RUBbed"."

>

Tulsidas says, "and makes a lot of sense, finally."

Exxy says, "So the RUB script checks that returned value, knows that 0 means "nope, the script didn't do anything special when RUBbed" and responds with a default message."

>

Exxy says, "For example..."

Exxy rubs a leather backpack.

>

Exxy rubs a jewel-inlaid ebony-pomeled sword.

Exxy rubs some silvery robes that flow nearly to the ground.

>

Tulsidas rubs a black leather backpack.

Tulsidas rubs some silvery robes that flow nearly to the ground.

>

Exxy says, "Default messages for objects that don't respond in a custom way to the RUB command."

>

Exxy says, "The diamond responds and so the default messaging is aborted."

Exxy asks, "So now we know the overview process and why it's important to be able to RETURN back values. Now how do we represent this in code?"

>

Exxy says, "If everyone will open up script 36 [MS000036.TXT], we'll take a look."

Exxy says, "FTP, then go to /GameData/Scripts/MS000036.TXT"

Exxy yawns.

>

Exxy stretches.

Maeve sighs.

>

Natheme reaches over and pokes Exxy.

Natheme says, "I have to offer you my thanks, Exxy."

>

Exxy watches his cat roll into a ball, stretch out, roll over, and then fall asleep upside-down.

>

Natheme asks, "how come some MS CDs are hologram and some are just colorful?"

Exxy grins at Natheme.

>  
Exxy says, "Dunno"  
Natheme says, "Sorry, I have a poor attention span."

>  
Natheme says, "I'm good so long as there's movement and action."  
Natheme says, "but once that stops, I'm lost."

>  
Exxy asks, "Who has the script so far?"  
Exxy turns around while observing his surroundings.

>  
Tess says, "i have it"  
Maeve dances on clouds to entertain Natheme

>  
Maeve says, "had it long time"  
Tess says, "belly dance"

>  
Tess nods to Maeve.  
Exxy asks, "natheme? Tulsidas?"

>  
Maeve shakes Natheme until his teeth rattle!

>  
Maeve says, "we lost him"  
'Its up for me just so ya know  
You say, "Its up for me just so ya know"

>  
Exxy nods to you.  
Natheme asks, "what?"

>  
Natheme says, "oh, I have it."  
Natheme preens.

>  
Exxy bops Natheme over the head.  
Maeve laughs.

>  
Exxy says, "Just Tulsidas now."  
Natheme says, "thank you for fixing that period"

>  
Natheme says, "like I asked"  
Exxy grins.

>  
Natheme says, "it made Maeve upset and me happy. best of both worlds."  
Maeve ponders.

>  
Natheme laughs in evil glee.  
Maeve asks, "what period?"

>  
Natheme pokes you  
Exxy says, "The auto-period at the end of the ACTION command."

Natheme pokes you.

>

Tess says, "Tuls is at work so maybe distracted a minute, he said something about lives hanging in the balance.."

>

Natheme says, "see"

'what?

You ask, "what?"

>

Maeve grunts.

Exxy grins.

>

Exxy says, "K, we'll he can catch up."

Exxy smiles.

> slobb nathem

You slobber on Natheme.

>

Natheme is flirting with you.

Maeve says, "ewww"

>

Maeve says, "that's so nasty"

Exxy says, "Notice about 20 lines down"

>

Exxy says, "It looks for an object, checks for an object scripts, runs the script, checks the return value, and preforms default messaging."

>

Exxy says, "I'm not going to make it confusing and try and type all that out into several lines here."

>

Exxy grins.

Natheme says, "lazy."

>

Maeve says, "yeah"

Maeve says, "I wanna see it"

>

Exxy asks, "Have a dollar?"

Lightning starts to show in the sky.

>

Maeve says, "I can make one"

Maeve grins at Exxy.

>

Exxy smiles.

Natheme says, "Exxy"

>

Exxy says, "Natheme."

Natheme asks, "will you make a variable for he/she?"

>

Exxy says, "There is one"

Natheme says, "oh wait, someone told me you did"

>

Natheme says, "I think it was Tess."  
Maeve babbles at Natheme.

>

Natheme ponders.  
Exxy says, "\$ch:player:sexarticle"

>

Maeve says, "this is one"  
Exxy says, "Returns 'his' for male, 'her' for female."

>

Natheme bops Exxy over the head.  
Natheme says, "that's his/her"

>

Natheme says, "I want he/she"  
Exxy says, "Deal with it."

>

Natheme says, "it'd really be good if you made he/she"  
Exxy says, "Or write a utility script"

>

Natheme says, "okay"  
Natheme says, "will do, hommie"

>

Exxy says, "No that'd be somewhat useful"  
Tess says, "himself and herself too"

>

Natheme sighs.  
Exxy says, "It'd be really good if I won the lottery."

>

Maeve says, "only if you share"  
Maeve stares at Exxy.

>

Natheme says, "how about if Solace pays you \$100 to make the he/she verb"  
Natheme says, "er, variable"

> bop nath  
You bop Natheme over the head.

>

Natheme says, "oy"  
Natheme says, "I shall thrash you."

>

Natheme says, "and your little dog too"  
Maeve asks, "Natheme?"

>

Natheme asks, "aye?"  
Maeve exclaims, "hush and let Exxy get on with it!"

>

Natheme ducks.

Exxy asks, "Anyhow, anyone have any questions why/how the script works? Is it at all confusing to anyone?"

>

Natheme says, "nope, pretty simple stuffins"

Exxy says, "RUB works almost the exact same way, but with different messaging."

>

Tess says, "so far so good, i can read what its doing even if i would have a hard time doing it from scratch"

>

Exxy says, "Same with TAP."

Exxy says, "Except RUB and TAP check to see if the object is anywhere visible."

>

Exxy says, "WAVE only checks to see if you're holding the object."

Exxy says, "Now open up script 151 [MS000151.TXT]."

>

Maeve babbles at Natheme.

Natheme says, "bah. it timed me out."

>

Exxy chuckles at Natheme.

Tess says, "open"

>

Tulsidas says, "open"

Maeve says, "open"

>

Natheme says, "viewing"

Maeve says, "and the answer is no Natheme"

>

Natheme says, "Sorry, I had to be different."

Natheme asks, "how about two parallel ports?"

>

Tess shakes you until your teeth rattle!

Maeve says, "will you stop"

>

Exxy says, "Okay, everyone.."

Natheme says, "heh, sorry."

>

Exxy says, "Look at script 151."

Maeve says, "we can talk about it later"

>

Tulsidas says, "me too this time"

Exxy asks, "Does it look difficult to understand?"

>

Exxy says, "There's some new commands in there but nothing that shouldn't be too difficult to decipher."

>

Tess shakes her head.

Natheme asks, "new commands?"



>  
Tulsidas says, "not really"  
Natheme asks, "which?"

>  
Tess asks, "||?"  
Tulsidas says, "piping"

>  
Maeve asks, "'or?"

Exxy says, "SET\_STAT command."

Natheme says, "oh, I see. I thought you meant new as in new as in just made... I... okay, I get it."

>  
Exxy grins at Natheme.

Maeve says, "heh"

>  
Tess says, "nobody tell him when its over"  
Exxy has fun with the commoners.

>  
Tess shakes her head.  
Maeve says, "kill him"

>  
Tulsidas chuckles.  
Maeve nods to Exxy.

>  
Exxy asks, "So you all okay with 151?"  
Tulsidas says, "yepp"

>  
Maeve says, "still with ||"  
Tulsidas says, "unbelievably so"

>  
Tess says, "Exxy teaches well"  
Tess nods to Tulsidas.

>  
Exxy grins.

>  
Exxy says, "Yes, I am."  
Exxy says, "Open up script 79 [MS000079.TXT -- see a pattern yet?]."

>  
Maeve says, "guess not"  
Exxy says, "This one uses another new command, RANDOM. Which lets you pick a number between X and Y and it gives you the result in \$returned."

>  
Exxy says, "With this, you can have an object give random messaging or random results."  
Exxy says, "Notice also the use of the 'RUN\_SCRIPT' command."

>  
Exxy says, "It runs script 56 (the utility script for checking the roundtime a player has)."

Exxy says, "If you have any seconds remaining, it aborts."

>

Exxy says, "(The messaging is handled by script 56)"

Exxy says, "This particular object (a kain doll) has three verbs."

>

Natheme asks, "can a script be accessed by more than one object?"

Exxy says, "Should be fairly simple to follow it."

>

Exxy says, "Absolutely."

Natheme nods.

>

Natheme says, "thought so, just making sure"

Natheme asks, "so you could use scripts like these and change a few things in them to set them up for say, a merchant or something with 5 dolls, different names, but same verbs?"

>

Exxy says, "The CHANGE\_OBJECT\_SHORTDESC command requires an object to be located and in the \$object master variable. So the script locates the object by the name 'doll001'. The object it refers to will have the actual name be 'doll001'"

>

Exxy nods to Natheme.

Exxy says, "Yes."

>

Exxy asks, "We comfy with this script?"

Tess says, "i was till you got to the doll001 part"

>

Tulsidas says, "well, i am tryin to snuggle up to it,,, yes"

Exxy grins at Tess.

>

Tulsidas says, "it is interesting"

Exxy nods.

>

Exxy says, "Get cozy with it"

Exxy says, "Open up a few room scripts."

>

Tess says, "what are the || things, Maeve and i both wanna know"

Tess nags Exxy.

>

Exxy says, "You can take a look at the room script 170, for instance."

Exxy asks, "What || things?"

>

Exxy says, "Oh"

Exxy chuckles.

>

Tess says, "the || things between parameters in the script"

Exxy says, "Someone hasn't read their reference manual."

>

Tess exclaims, "did too!"

Exxy says, "|| is the same as 'Or'."

>  
Tess says, "oh"  
Exxy says, "Open the refernce manual and click on 'Expressions'."

>  
Tess ducks.  
Tess says, "its alot to remember"

>  
Exxy says, "|| means 'logical OR comparison' in C/C++ and perl."  
Tess says, "oh"

>  
Exxy says, "In BASIC/VB you'd use the 'Or' keyword."  
Exxy says, "We support both."

>  
Exxy says, "I'm personally more accustomed to C/C++ so I use ||"  
Tess says, "okies, that answered my question but hers was prolly not the same question if thats the answer"

>  
Exxy asks, "Then..... what was yer question Maeve?"  
Tess says, "btw, i have awful thunder so i may poof"  
Maeve says, "doesn't matter"

>  
Maeve says, "just continue along"  
Exxy says, "That's okay, I think we're about done for tonight"

>  
Exxy says, "It's after 10pm EST and I'm diein over here."  
Exxy says, "Room scripts shouldn't be too difficult to understand"

>  
Tess says, "its alot to digest, least for me"  
Tulsidas says, "i got alot to study on.. and i am gonna"

>  
Exxy says, "Take a look at them and I'll answer questions tomorrow"

Maeve nods to Exxy.

>  
Rhiann exclaims, "Yes!!"  
Tulsidas nods.

>  
Exxy says, "Okay."  
Exxy says, "Then lets make this conference room a shop."

>  
Exxy says, "From scratch."  
Maeve grins.

>  
Exxy says, "All we need is an object or two to sell to start with."  
Exxy says, "I have a backpack"

>  
Exxy rubs a leather backpack.  
Tulsidas gets a test object from inside his cloak.

>  
Maeve gets a dill pickle from inside her bag.  
Exxy says, "Let's make that for sale"

>  
Tulsidas taps a test object.  
Maeve licks a dill pickle.

>  
Exxy glances Tulsidas.  
Exxy says, "This would work better if I get to talk and there's not a lot of scroll"

>  
Exxy grins.  
Exxy says, "First thing we need to do is make the object invisible and 'able to be manipulated while invisible'."

>  
Exxy says, "This can be done with the spreadsheet up on the website."  
Exxy waits for it to load....

>  
Exxy grins.  
Maeve puts a dill pickle in a lilac quilted bag.

>  
Exxy says, "In fact, it'll be quicker to create an object from scratch."  
Exxy drops a short sword.

>  
Exxy says, "There."  
The night has begun.

>  
Exxy says, "Now you'll still be able to see the sword because you are a staff member."  
Exxy says, "(Really, you are)"

>  
Exxy grins.  
Exxy says, "But no one else will be able to see the sword on the ground here."

>  
Exxy says, "Now we need to make the short sword buyable."  
Exxy says, "We attach script 124 to the object..."

>  
Exxy says, "(CHANGE\_OBJECT\_SCRIPT sword 124)"  
Exxy says, "And now anyone can BUY the SWORD."

>  
Exxy says, "There's two more things you should do though."  
Exxy says, "#1 (and the most important) normal people can pick up the sword"

>  
Maeve asks, "fixed that part?"  
Maeve grins at Exxy.

>  
Exxy says, "That's a bad thing. To avoid that, we would normally set the object to 'nonpickupable' or 'staff-only'."

>

>  
Exxy says, "But that's not what we want to do, because then when they purchase the item and drop it (or put it in a container), they wouldn't be able to pick it back up."

>  
Exxy says, "What we want to do instead, is write a special room script to block the player from picking up or dropping items in this room."

>  
Exxy says, "Likewise, when we have stealing and combat systems live, we'll want the room script to disable that too for this area."

>  
Exxy says, "We don't want a player to be able to DROP a sword in this room because then when people go to BUY SWORD, the engine will not see the correct sword on the ground."

>  
Exxy says, "So, as noted in my test of a room script # 170, we'll block all players from picking up or dropping objects."

>  
Exxy says, "I made a cheapo script that does not just now"  
Exxy says, "\*that does that just now"

>  
Exxy mutters.  
Exxy grins.

>  
Maevve chuckles.  
Exxy says, "No one should be able to drop/take/get objects now."

>  
Tulsidas grins.  
Maevve asks, "does that excude staff?"

>  
Exxy says, "There are of course, more sophisticated ways of handling this"  
Exxy says, "No, it's not excluding anyone right now."

>  
Tulsidas puts a pointy stick in a well-worn hooded doeskin cloak.  
Tulsidas puts a test object in a well-worn hooded doeskin cloak.

>  
Exxy says, "PUT doesn't hurt anything."  
Exxy says, "Though I guess if the object we wanna buy is a container....."

>  
Exxy grins.  
Maevve nods to Exxy.

>  
Tulsidas grins.  
Exxy says, "This is the simplist way to create a shop."

>  
Exxy says, "The only thing to do now is to add a simple sign/menu/catelog that shows it."  
Exxy hands a merchant some gold and claims his short sword.

>  
Exxy waves a short sword around.  
Exxy puts a short sword in a leather backpack.

>

Tulsidas hands a merchant some gold and claims his short sword.  
Tulsidas says, "ha"

>

Tess hands a merchant some gold and claims her short sword.  
Tulsidas puts a short sword in a well-worn hooded doeskin cloak.

>

Kaze blinks.  
Tess says, "ya know i had no gold"

>

Exxy says, "Right now, the shopkeeper will not actually take any gold."  
Tulsidas says, "i still got 915"

>

Exxy nods to Tulsidas.  
Exxy says, "Other ways of creating a shop could be done entirely from a room script."

>

Exxy says, "Where you could trap and see if they typed 'BUY SWORD' you could create an object and transfer it to the character (if they have a spare hand, or on a counter perhaps if not)"

>

Exxy says, "And that avoids the GET/TAKE/DROP issue altogether, but involves more coding."  
Exxy says, "This is just one of many ways it can be done."

>

Exxy asks, "Any questions?"  
Maeve says, "it's a nice idea so something have things displayed though"

>

Tess whispers, "accept link on your IM"  
Maeve babbles.

>

Exxy says, "Yes, of course."  
Exxy smiles.

>

Maeve asks, "you read typo right?"  
Maeve grins at Exxy.

>

Exxy says, "Sure"  
Exxy grins.

>

Maeve asks, "does this avoid the problem comoners have been having?"  
Exxy says, "Also now if you're creating a shop in this first method, you may want to have code in your script send a message anyone who's a staff member and allowed to pick up objects regardless of their flags/room restrictions "

>

Exxy says, "Saying "you picked up an object that's in this shop, you moron" or somethin"  
Exxy nods to Maeve.

>

Maeve says, "nice idea"  
Maeve nods to Exxy.

>

Exxy says, "The object that was purchased in a shop by a commoner and then realized he/she couldnt pick it up was either trying to pick up the object in the room or the object itself had the "staff only" flag on or wasn't set to "pickupable"."

>

Maeve says, "had two last night with swords in backpack that they couldn't get out"  
Maeve grins at Exxy.

Exxy says, "We have special room scripts set up to block out all but our priviledged testers though"

>

Maeve says, "good stuff"  
Exxy says, "So only a few commoners are allowed to move past the "blocks""

>

Maeve asks, "the system keeps a list?"  
Tess asks, "they understand that some of those areas are not finished right?"

>

Rhiann asks, "Will the non priviledged feel left out?"  
Exxy says, "Hopefully"

>

Exxy grins at Rhiann.  
Maeve laughs at Exxy.

>

Exxy says, "The system has been preprogrammed with a list of people who are 'priviledged'"  
Rhiann grins at Exxy.

>

Exxy says, "Anyone not on that list and not a member of staff will be denied."  
Maeve nods.

>

Exxy nods to Tess.  
Exxy says, "Yes, they do."

>

Tess asks, "do we have a way to see the priviledged list?"  
Exxy says, "Eve/Severa, Axenar, Kain, Lightwalker, Taofang, Gavyne, Edge, and Majebrad"

Exxy asks, "So did that answer all your questions about shops for now?"  
Exxy smiles.