Subject: Creating classes for Esc and using in Esc.
Posted by jeremy_c on Fri, 06 Aug 2010 16:41:43 GMT
View Forum Message <> Reply to Message

I have the following code for Esc. It's obviously for learning purposes. Once I get this figured out I'll move on to more complex classes.

```
struct SlescPerson : public EscHandle
{
 String mName;

 void GetName(EscEscape& e)
 {
  Exclamation("Getting Name");
  e = mName;
 }

 void SetName(EscEscape& e)
 {
  Exclamation("Setting Name");
  mName = e[0];
 }

 typedef SlescPerson CLASSNAME;

 SlescPerson(EscValue& v)
 {
  Exclamation("Constructing Person");
  v.Escape("GetName()", this, THISBACK(GetName));
  v.Escape("SetName(value)", this, THISBACK(SetName));
 }
};
```

Then, I have this in my application to run it:

```
ArrayMap<String, EscValue> global;

StdLib(global);
Escape(global, "Exclamation(value)", SLESC_Exclamation);
Escape(global, "Person()", SLESC_Person);

Evaluate("p = Person(); p.SetName(\"John\"); Exclamation(p.GetName());", global);
```

I get the initial exclamation "Constructing Person" but after that I get nothing.

Can someone point me to what I'm doing wrong?

Thanks,

Jeremy

Subject: Re: Creating classes for Esc and using in Esc.
Posted by jeremy_c on Fri, 06 Aug 2010 18:06:57 GMT
View Forum Message <> Reply to Message

Never mind, I figured it out. I can only use Evaluate to eval the first statement it seems?

```
Scan(global,
  "main() {\n"
  "  p = Person();\n"
  "  p.SetName(\"John\");\n"
  "  Exclamation(p.GetName());\n"
  "}");
Upp::Execute(global, "main");
```

caused my "Person" example to work.

Jeremy
