Programming Object on Forms

- Understand OS Events
- Various fns for each control

Type of Events:

- When the user first tap within the border of an object, an “enter” event of some type is generated.
- If then lift stylus within same object, a “select” event is generated.
- If lift outside of objects border, an event is generated.
- Repeat buttons scrollbars can generate repeat events
- Name of Events:
  ctlSelectEvent (Where (what kind of object) What (kind of event))

frmTitleEnterEvent
lstExit Event (list)
fldSelectEvent(field)
popSelectEvent(pop-up)
tblSelectEvent (table)
sclEnterEvent (scrollbar)
sclRepeatEvent
ctlRepeatEvent

Checking for Events

In event handler:

```
if (event->eType=ctlSelectEvent)
{
        //code here
}
```

To get the index of object selected…

```
event->data.ctlSelect.controlID
```

Example:

```
event->data.lstExit.listID
event->data.sclRepeat.scrollbarID
```

To do something with an object on a form, you need to set a ptr to it

Example:

```
FormType *form;
ControlType *ctl;
Form=FrmGetActiveForm();
Ctl= FrmGetObjectPtr(form, FrmGetObjectIndexfor, MainOkButton):
```

(very common idiom,, book makes a function GetObjectPtr to do this)

Hiding and showing an object:

Example: Hide a button depending on state of a checkbox

```
Ctl = GetObjectPtr(HideCheckBox); // (ID of check box)
If( Ctl.GetValue(ctl))
{
        FrmHideObject(form, buttonIndex);
}
else
{
        FrmShowObject(form, buttonIndex);
}
```

Do not directly access struct members for objects. Struct are shadowed on ARM processors (>OS5) so it can cause problems. Instead use PalmOS glue.

```
Button=FrmGetObjectPtr(form, FrmGetOjectIndex(form, myButton));
font = button->font; // do not do

// instead
font=CtlGetFont(button);
```

To compile need to add:

```
-l PalmOSGlue to compile line
```