

This code illustrate also a setting user defined font size, where some trouble occur with changing initial text's position

```
enum {fntSize = 19};

struct AppNodeSize : public Display {
    virtual void Paint(Draw& w,const Rect& r, const Value& q,Color ink,Color paper,dword style)
    const {
        w.DrawRect(r, paper);
        w.DrawText(r.left, r.top, (String)q, StdFont().Height(fntSize), ink);
    }

    Size GetStdSize(const Value& q) const {
        //1.5 is a empirical multiplier for increasing row height when fntSize used in (1) and (2);
        //how to universally set it based on fntSize?
        return 1.5 * StdDisplay().GetStdSize(q);
    }
};

class EditableTree : public TreeCtrl {
public:
    typedef EditableTree CLASSNAME;

    EditableTree() : selId_(0) {
        NoRoot();

        editNode_.SetCtrl(edit_);
        edit_.NoInternalMargin().SetFrame(NULLFrame()); //crashes when NoInternalMargin
        edit_.SetFont(StdFont().Height(fntSize)); // (1)

        WhenSel = THISBACK(OnSelect);
    }

    void AddNode() {
        CloseEdit();

        //with some ico in /**/ offset lines more visible
        selNode_ = Node(/*espudImg::secExpand(), */"NODE" + AsString(GetChildCount(0) + 1));
        selNode_.SetDisplay(Single<AppNodeSize>()); // (2)

        selId_ = Add(0, selNode_);
        BeginEdit();
    }
};
```

```

}

void OnSelect() {
    CloseEdit();

    selId_ = GetSel()[0];
    selNode_ = GetNode(selId_);

    BeginEdit();
}

private:
int selId_;
    Node selNode_;

    EditString edit_;
    Node editNode_;

void BeginEdit() {
    editNode_.image = selNode_.image;
    SetNode(selId_, editNode_);

    String val = selNode_.value;
    edit_.SetData(val);
    edit_.AutoSize();
    edit_.Show();
    edit_.SetFocus();
    edit_.SetSelection(val.GetCount());
}

void CloseEdit() {
    if (selId_ > 0) {
        selNode_.value = ~edit_;
        SetNode(selId_, selNode_);
        edit_.Hide();
    }
}
};

```
