

[Help](#)

```

#include "stda.h"

static NumFunc_1 put=
{
    Put,
    {{"Strike",PDOUBLE,{100},FORBID,UNSETABLE},{ " ",PREMIA_
    NULLTYPE,{0},FORBID,SETABLE}},
    CHK_call
};

static TYPEOPT EquityLinkedSurrenderEndowment=
{
    /* PayOff; */ {"Payoff",NUMFUNC_1,{0},FORBID,UNSETABLE
},
    /*EuOrAm*/      {"Amer",BOOL,{AMER},FORBID,UNSETABLE}
,
    /*Maturity*/    {"Maturity(in years)",DATE,{0},ALLOW,
SETABLE},
    /*DeemedContribution;*/ {"Deemed Contribution",PDOUB
LE,{0},ALLOW,SETABLE},
    /*InitialAge*/ {"Initial Age",PDOUBLE,{0},ALLOW,SETA
BLE},
    /*Premium;*/ {"Premium",PDOUBLE,{0},ALLOW,SETABLE},
    /*MinimumGuaranteedInterestRate*/ {"MinimumGuarantee
dInterestRate",PDOUBLE,{0},ALLOW,SETABLE},
};

static int OPT(Init)(Option *opt,Model *mod)
{
    TYPEOPT* pt=( TYPEOPT*)(opt->TypeOpt);

    if (opt->init == 0 )
    {
        opt->init = 1;
        opt->HelpFilenameHint = "equity_linked";
        opt->nvar = 7;
        opt->nvar_setable = 5;

        pt->PayOff.Val.V_NUMFUNC_1=&put;
        (pt->PayOff.Val.V_NUMFUNC_1)->Par[0].Val.V_PDOUBLE=10

```

```
0.0;
    (pt->EuOrAm).Val.V_BOOL=AMER;

    (pt->Maturity).Val.V_DATE=5.;
    (pt->Premium).Val.V_PDOUBLE=106;
    (pt->DeemedContribution).Val.V_PDOUBLE=100;
    (pt->MinimumGuaranteedInterestRate).Val.V_PDOUBLE=0.0
2;
    (pt->InitialAge).Val.V_PDOUBLE=50;
}
return OK;
}
```

MAKEOPT(EquityLinkedSurrenderEndowment);

## References