

[Help](#)

```
#include "lrshjm1d.h"
#include "chk.h"
#include "error_msg.h"
#include "premia_obj.h"
#include "model.h"

double MOD(GetYield)(TYPEMOD *pt)
{
    VAR *Par;
    int val;
    val = pt->flat_flag.Val.V_ENUM.value;
    Par = lookup_premia_enum_par(&(pt->flat_flag), val);
    return Par[0].Val.V_PDOUBLE;
}

char *MOD(GetCurve)(TYPEMOD *pt)
{
    VAR *Par;
    Par = lookup_premia_enum_par(&(pt->flat_flag), 1);
    return Par[1].Val.V_FILENAME;
}

static int MOD(Init)(Model *model)
{
    VAR *Par;
    TYPEMOD *pt = (TYPEMOD *) (model->TypeModel);

    if (model->init == 0)
    {
        model->init = 1;
        model->nvar = 0;

        pt->T.Vname = "Current Date";
        pt->T.Vtype = DATE;
        pt->T.Val.V_DATE = 0.0;
        pt->T.Viter = ALLOW;
        model->nvar++;
    }
}
```

```
pt->flat_flag.Vname = "Initial Yields Curve";
pt->flat_flag.Vtype = ENUM;
pt->flat_flag.Val.V_ENUM.value = 0;
pt->flat_flag.Val.V_ENUM.members = &PremiaEnumFlat2;
pt->flat_flag.Viter = ALLOW;
model->nvar++;
Par = lookup_premia_enum_par(&(pt->flat_flag), 0);
Par[0].Vname = "Yield Value";
Par[0].Vtype = PDOUBLE;
Par[0].Val.V_PDOUBLE = 0.05;
Par[0].Viter = ALLOW;
Par = lookup_premia_enum_par(&(pt->flat_flag), 1);
Par[0].Vname = "Initial r0";
Par[0].Vtype = PDOUBLE;
Par[0].Val.V_PDOUBLE = 0.05;
Par[0].Viter = ALLOW;
Par[1].Vname = "Yield Curve";
Par[1].Vtype = FILENAME;
Par[1].Val.V_FILENAME = NULL;
Par[1].Viter = FORBID;

pt->Sigma.Vname = "Sigma Parameter";
pt->Sigma.Vtype = PDOUBLE;
pt->Sigma.Val.V_PDOUBLE = 0.1;
pt->Sigma.Viter = ALLOW;
model->nvar++;

pt->Kappa.Vname = "Kappa Parameter";
pt->Kappa.Vtype = PDOUBLE;
pt->Kappa.Val.V_PDOUBLE = 0.02;
pt->Kappa.Viter = ALLOW;
model->nvar++;

pt->Rho.Vname = "Rho Parameter";
pt->Rho.Vtype = PDOUBLE;
pt->Rho.Val.V_PDOUBLE = 1;
pt->Rho.Viter = ALLOW;
model->nvar++;

pt->Lambda.Vname = "Lambda Parameter";
```

```
    pt->Lambda.Vtype = PDOUBLE;
    pt->Lambda.Val.V_PDOUBLE = 1.224744871;
    pt->Lambda.Viter = ALLOW;
    model->nvar++;

    model->HelpFilenameHint = "LRSHJM1D";

}
Par = lookup_premia_enum_par(&(pt->flat_flag), 1);
if (Par[1].Val.V_FILENAME == NULL)
{
    if ((Par[1].Val.V_FILENAME = malloc(sizeof(char) * MAX_PATH_LEN)) == NULL)
        return MEMORY_ALLOCATION_FAILURE;
    sprintf(Par[1].Val.V_FILENAME, "%s%sinitialyield.dat", premia_data_dir, pa
}

return OK;
}

TYPEMOD LiRitchkenSankarasubramanian1d;
MAKEMOD(LiRitchkenSankarasubramanian1d);
```