

[Help](#)

```
#include "
href../../mod/kould/kould_stda/kould_stda_h_src.pdfkould_stda.h"

int MOD_OPT(ChkMix)(Option *Opt, Model *Mod)
{
    TYPEOPT *ptOpt = (TYPEOPT *) (Opt->TypeOpt);
    TYPEMOD *ptMod = (TYPEMOD *) (Mod->TypeModel);
    int status = OK;

    if (ptOpt->Maturity.Val.V_DATE <= ptMod->T.Val.V_DATE)
    {
        Fprintf(TOSCREENANDFILE, "Current date greater than maturity!\ n");
        status += 1;
    };

    return status;
}

extern PricingMethod MET(AP_GAP_KOU);
extern PricingMethod MET(MC_Icppi_KOU);
PricingMethod *MOD_OPT(methods) [] =
{
    &MET(AP_GAP_KOU),
    &MET(MC_Icppi_KOU),
    NULL
};
DynamicTest *MOD_OPT(tests) [] =
{
    NULL
};

Pricing MOD_OPT(pricing) =
{
    ID_MOD_OPT,
    MOD_OPT(methods),
    MOD_OPT(tests),
    MOD_OPT(ChkMix)
```

};