

[Help](#)

```
#include "
href../common/optype_h_src.pdfoptype.h"
#include "
href../common/var_h_src.pdfvar.h"
#include "
href../common/test_h_src.pdftest.h"
#include "
href../common/error_msg_h_src.pdferror_msg.h"

extern char premia_man_dir[MAX_PATH_LEN];
extern char *path_sep;

/*-----DYNAMICTESTS-----
int  GetTest(int user, Planning *pt_plan, Pricing *Pr, Option *Opt, DynamicTest
{
    (Test->Init)(Test, Opt);
    if (user == TOSCREEN)
    {

        Fprintf(TOSCREEN, "\ n%s\ n", Test->Name);

        if (ShowTest(user, pt_plan, Pr, Test))
        {
            do
            {
                GetParVar(pt_plan, user, Test->Par);
            }
            while (ShowTest(user, pt_plan, Pr, Test));
        }
    }
    return ShowTest(TOSCREENANDFILE, pt_plan, Pr, Test);
}

int ShowTest(int user, Planning *pt_plan, Pricing *Pr, DynamicTest *Test)
{
    char helpfile[MAX_PATH_LEN] = "";
    int pos;
    char *pdest;
```

```

if ((2 * strlen(Pr->ID) + 4 * strlen(path_sep) + strlen("mod") + strlen(Test->
    + strlen("_src.pdf"))
    >= MAX_PATH_LEN)
{
    Fprintf(TOSCREEN, "%s\ n", error_msg[PATH_TOO_LONG]);
    exit(WRONG);
}

strcpy(helpfile, premia_man_dir);
strcat(helpfile, path_sep);
strcat(helpfile, "mod");
strcat(helpfile, path_sep);

pdest = strchr(Pr->ID, '_');
pos = pdest - Pr->ID;

strncat(helpfile, Pr->ID, pos);

strcat(helpfile, path_sep);

strcat(helpfile, Pr->ID);
strcat(helpfile, path_sep);
strcat(helpfile, Test->Name);
strcat(helpfile, "_src.pdf");

Fprintf(user, "\ n##DynamicTest:%s\ n", Test->Name);

if (user == TOSCREENANDFILE)
{
    ShowParVar(pt_plan, user, Test->Par);
}
else
{
    if (ShowParVar(pt_plan, user, Test->Par) == OK)
    {
        return Valid(user, ChkParVar(pt_plan, Test->Par), helpfile);
    }
    else

```

```

        return Valid(NO_PAR, ChkParVar(pt_plan, Test->Par), helpfile);
    }

    return OK;
}

int ShowResultTest(int user, Planning *pt_plan, int error, DynamicTest *Test)
{

    /*NEW!*/ /*ShowParVar(pt_plan,TOSCREEN,Test->Res);*/

    if ((user == NAMEONLYTOFILE) || (error == 0))
    {
        /* ShowParVar(pt_plan,user,Test->Res);*/
        ShowParVarTestRes(pt_plan, user, Test->Res);
    }
    else
    {
        Fprintf(user, "%s\ n", error_msg[error]);
    }

    return OK;
}

```