

NAME

form_field - make and break connections between fields and forms

SYNOPSIS

```
#include <form.h>
```

```
int set_form_fields(FORM *form, FIELD **fields);
```

```
FIELD **form_fields(const FORM *form);
```

```
int field_count(const FORM *form);
```

```
int move_field(FIELD *field, int frow, int fcol);
```

DESCRIPTION

The function **set_form_fields** changes the field pointer array of the given *form*. The array must be terminated by a **NULL**.

The function **form_fields** returns the field array of the given form.

The function **field_count** returns the count of fields in *form*.

The function **move_field** moves the given field (which must be disconnected) to a specified location on the screen.

RETURN VALUE

The function **form_fields** returns a pointer (which may be **NULL**). It does not set **errno**.

The function **field_count** returns **ERR** if the *form* parameter is **NULL**.

The functions **set_form_fields** and **move_field** return one of the following codes on error:

E_OK

The routine succeeded.

E_BAD_ARGUMENT

Routine detected an incorrect or out-of-range argument.

E_CONNECTED

The field is already connected to a form.

E_POSTED

The form is already posted.

form_field(3X)

form_field(3X)

E_SYSTEM_ERROR

System error occurred (see **errno(3)**).

SEE ALSO

curses(3X), **form(3X)**.

NOTES

The header file **<form.h>** automatically includes the header file **<curses.h>**.

PORTABILITY

These routines emulate the System V forms library. They were not supported on Version 7 or BSD versions.

The SVr4 forms library documentation specifies the **field_count** error value as -1 (which is the value of **ERR**).

AUTHORS

Juergen Pfeifer. Manual pages and adaptation for new curses by Eric S. Raymond.

form_field(3X)