

**NAME**

**expr** – c-like expression library

**SYNOPSIS**

```
#include <graphviz/expr.h>

Expr_t*          exopen(Exdisc_t*);
Excc_t*          excopen(Expr_t*, Exccdisc_t*);
int              excc(Excc_t*, const char*, Exid_t*, int);
int              excclose(Excc_t*);
void             exclose(Expr_t*, int);
char*            excontext(Expr_t*, char*, int);
void             exerror(const char*, ...);
Exnode_t*        exeval(Expr_t*, Exnode_t*, void*);
Exnode_t*        exexpr(Expr_t*, const char*, Exid_t*, int);

Exnode_t*        excast(Expr_t*, Exnode_t*, int, Exnode_t*, int);
Exnode_t*        exnewnode(Expr_t*, int, int, int, Exnode_t*, Exnode_t*);
void             exfreenode(Expr_t*, Exnode_t*);
int              expush(Expr_t*, const char*, int, const char*, Sfio_t*);
int              expop(Expr_t*);
int              excomp(Expr_t*, const char*, int, const char*, Sfio_t*);
int              exrewind(Expr_t*);
void             exstatement(Expr_t*);
int              extoken(Expr_t*);
char*            extype(int);
Extype_t         exzero(int);
```

**DESCRIPTION**

`exopen()` is the first function called. `exclose()` is the last function called. `excopen()` is the called if code generation will be used. `exccclose()` releases the state information allocated in `excopen()`. `exstatement()` saves statement start information. `exrewind()` restores statement start information saved by `exstatement()`.

**SEE ALSO**