

identifier.l

```
%{
#include "y.tab.h"
%}

%%
[a-zA-Z][a-zA-Z0-9]*      { return IDENTIFIER; }
[^\t\n]+                  { return INVALID; }
[ \t\n]+                  ;
%%

int yywrap() {
    return 1;
}
```

identifier.y

```
%{
#include <stdio.h>
#include <stdlib.h>

int yylex(void);
void yyerror(const char *s);
int error_handled = 0;
%}

%token IDENTIFIER INVALID

%%
input:
    IDENTIFIER      { printf("Valid variable name.\n"); }
  | INVALID         { printf("Invalid variable name.\n"); error_handled = 1; }
  ;
%%

void yyerror(const char *s) {
    if (!error_handled)
        printf("Syntax Error: %s\n", s);
}

int main() {
    printf("Enter a variable name: ");
    yyparse();
    return 0;
}
```

## OUTPUT

```
[cdlab88@localhost valid_variable]$ vi identifier.y
[cdlab88@localhost valid_variable]$ lex identifier.l
[cdlab88@localhost valid_variable]$ yacc -d identifier.y
[cdlab88@localhost valid_variable]$ gcc lex.yy.c y.tab.c -o identifier
[cdlab88@localhost valid_variable]$ ./identifier
Enter a variable name: hello123
Valid variable name.
^C
[cdlab88@localhost valid_variable]$ ./identifier
Enter a variable name: 123hello
Invalid variable name.
^C
[cdlab88@localhost valid_variable]$ ./identifier
Enter a variable name: hello123
Valid variable name.
123hello
Syntax Error: syntax error
[cdlab88@localhost valid_variable]$ ./identifier
Enter a variable name: 123hello
Invalid variable name.
^C
[cdlab88@localhost valid_variable]$ ./identifier
Enter a variable name: @var
Invalid variable name.
^C
```