Test Code:	Lexer/Parser Output:

{print ("start")
int a
int b
a = 1
b = true
c = 2 + 4
{
boolean c
}
print ("end")
}$
	Compilation Started
	Lex returned
	[{,print,(,"start",),int,a,int,b,a,=,1,b,=,true,c,=,2,+,4,{,boolean,c,},print,(,"end",),},$]
	Lex Errors: 0 | Lex Warnings: 0
	Token Accepted: Expecting input of RegEx form /\{/, found token of value {
	Token Accepted: Expecting input of RegEx form /print/, found token of value print
	Token Accepted: Expecting input of RegEx form /\(/, found token of value (
	Token Accepted: Expecting token of string type, found token of value "start"
	Token Accepted: Expecting input of RegEx form /\)/, found token of value)
	Token Accepted: Expecting input of RegEx form /(int)|(string)|(boolean)/, found token of value int
	Token Accepted: Expecting input of RegEx form /[a-z]/, found token of value a
	Token Accepted: Expecting input of RegEx form /(int)|(string)|(boolean)/, found token of value int
	Token Accepted: Expecting input of RegEx form /[a-z]/, found token of value b
	Token Accepted: Expecting input of RegEx form /[a-z]/, found token of value a
	Token Accepted: Expecting input of RegEx form /=/, found token of value =
	Token Accepted: Expecting input of RegEx form /[0-9]/, found token of value 1
	Token Accepted: Expecting input of RegEx form /[a-z]/, found token of value b
	Token Accepted: Expecting input of RegEx form /=/, found token of value =
	Token Accepted: Expecting input of RegEx form /(true)|(false)/, found token of value true
	Token Accepted: Expecting input of RegEx form /[a-z]/, found token of value c
	Token Accepted: Expecting input of RegEx form /=/, found token of value =
	Token Accepted: Expecting input of RegEx form /[0-9]/, found token of value 2
	Token Accepted: Expecting input of RegEx form /\+/, found token of value +
	Token Accepted: Expecting input of RegEx form /[0-9]/, found token of value 4
	Token Accepted: Expecting input of RegEx form /\{/, found token of value {
	Token Accepted: Expecting input of RegEx form /(int)|(string)|(boolean)/, found token of value boolean
	Token Accepted: Expecting input of RegEx form /[a-z]/, found token of value c
	Token Accepted: Expecting input of RegEx form /\}/, found token of value }
	Token Accepted: Expecting input of RegEx form /print/, found token of value print
	Token Accepted: Expecting input of RegEx form /\(/, found token of value (
	Token Accepted: Expecting token of string type, found token of value "end"
	Token Accepted: Expecting input of RegEx form /\)/, found token of value)
	Token Accepted: Expecting input of RegEx form /\}/, found token of value }
	Token Accepted: Expecting input of RegEx form /\$/, found token of value $
	Parse Errors: 0 | Parse Warnings: 0

{{{int a}}	Compilation Started Lex returned [{,{,{,int,a,},},$] Warning: EOF symbol not found at end of program, $ inserted at end Lex Errors: 0 | Lex Warnings: 1 Token Accepted: Expecting input of RegEx form /\{/, found token of value { Token Accepted: Expecting input of RegEx form /\{/, found token of value { Token Accepted: Expecting input of RegEx form /\{/, found token of value {

	Token Accepted: Expecting input of RegEx form /(int)|(string)|(boolean)/, found token of value int Token Accepted: Expecting input of RegEx form /[a-z]/, found token of value a Token Accepted: Expecting input of RegEx form /\}/, found token of value } Token Accepted: Expecting input of RegEx form /\}/, found token of value } Parse Error: Line 1, Found $, Expecting input of RegEx form /\}/ Token Accepted: Expecting input of RegEx form /\$/, found token of value $ Parse Errors: 1 | Parse Warnings: 0
{a = 2 + 22}$	Compilation Started Lex returned [{,a,=,2,+,2,2,},$] Lex Errors: 0 | Lex Warnings: 0 Token Accepted: Expecting input of RegEx form /\{/, found token of value { Token Accepted: Expecting input of RegEx form /[a-z]/, found token of value a Token Accepted: Expecting input of RegEx form /=/, found token of value = Token Accepted: Expecting input of RegEx form /[0-9]/, found token of value 2 Token Accepted: Expecting input of RegEx form /\+/, found token of value + Token Accepted: Expecting input of RegEx form /[0-9]/, found token of value 2 Parse Error: Line 1, Found 2, Expecting input of RegEx form /\}/ Parse Error: Line 1, Found }, Expecting input of RegEx form /\$/ Parse Errors: 2 | Parse Warnings: 0

{
{{print((true != false))}}
while true {
string z = "z"
string y = "y"
$
}
}$
	Compilation Started
	Lex returned [{,{,{,print,(,(,true,!=,false,),),},},while,true,{,string,z,=,"z",string,y,=,"y",$]
	Warning: EOF symbol found in middle of program, code afterward ignored
	Lex Errors: 0 | Lex Warnings: 1
	Token Accepted: Expecting input of RegEx form /\{/, found token of value {
	Token Accepted: Expecting input of RegEx form /\{/, found token of value {
	Token Accepted: Expecting input of RegEx form /\{/, found token of value {
	Token Accepted: Expecting input of RegEx form /print/, found token of value print
	Token Accepted: Expecting input of RegEx form /\(/, found token of value (
	Token Accepted: Expecting input of RegEx form /\(/, found token of value (
	Token Accepted: Expecting input of RegEx form /(true)|(false)/, found token of value true
	Token Accepted: Expecting input of RegEx form /(!=)|(==)/, found token of value !=
	Token Accepted: Expecting input of RegEx form /(true)|(false)/, found token of value false
	Token Accepted: Expecting input of RegEx form /\)/, found token of value)
	Token Accepted: Expecting input of RegEx form /\)/, found token of value)
	Token Accepted: Expecting input of RegEx form /\}/, found token of value }
	Token Accepted: Expecting input of RegEx form /\}/, found token of value }
	Token Accepted: Expecting input of RegEx form /while/, found token of value while
	Token Accepted: Expecting input of RegEx form /(true)|(false)/, found token of value true
	Token Accepted: Expecting input of RegEx form /\{/, found token of value {
	Token Accepted: Expecting input of RegEx form /(int)|(string)|(boolean)/, found token of value string

	Token Accepted: Expecting input of RegEx form /[a-z]/, found token of value z Parse Error: Line 4, Found =, Expecting input of RegEx form /\}/ Token Accepted: Expecting input of RegEx form /[a-z]/, found token of value "z" Parse Error: Line 5, Found string, Expecting input of RegEx form /=/ Parse Error: Line 5, Found y, Expecting "(", a digit, a string, or a char from a-z. Token Accepted: Expecting input of RegEx form /[a-z]/, found token of value y Token Accepted: Expecting input of RegEx form /=/, found token of value = Token Accepted: Expecting token of string type, found token of value "y" Parse Error: Line 6, Found $, Expecting input of RegEx form /\}/ Token Accepted: Expecting input of RegEx form /\$/, found token of value $ Parse Errors: 4 | Parse Warnings: 0