

AFL ASSIGNMENT

NAME:CHANDAN KH

SRN:PES1UG21CS147

NAME:CHIRAG H

SRN:PES1UG21CS157

SECTION:C

TOPICS:

1) SWITCH CASE

2)WHILE LOOP

3)DO WHILE LOOP

IN TYPESCRIPT LANGUAGE

1.SWITCH CASE

CFG OF THE SYNTAX

G={V,T,P,S}

V={s,condition,statement}

T={while,ID,Number,=,{,},(,),,,<=,>=}

S=s

P=productions

s->switch(expression){case expression1:{statement;break;} case expression2:{statement;break;} default:{statement;break;}}

condition->ID==ID|Number==ID|ID==Number|Number==Number|ID<=Number|ID>=Number|ID<=ID|ID>=ID|

expression->ID|Number|ID+Number|Number+ID|Number+Number

expression1->ID|Number|ID+Number|Number+ID|Number+Number

expression2->ID|Number|ID+Number|Number+ID|Number+Number

statement->ID=ID;|ID=Number;|ID++|s|statement

```
import ply.lex as lex
import ply.yacc as yacc

reserved={'switch':'switch','case':'case','default':'default','break':'break'}

tokens=['EQUAL','LPAREN','RPAREN','NUMBER','ID','LCURLY','RCURLY','SEMICOLON',
'PLUS','LESS','MORE','MINUS','COLON']+list(reserved.values())

t_EQUAL=r'\='
t_LPAREN=r'\('
t_RPAREN=r'\)'
t_LCURLY=r'\{'
t_RCURLY=r'\}'
t_SEMICOLON=r'\;'
```

```

t_COLON=r'\:'
t_PLUS=r'\+'
t_MINUS=r'\-'
t_LESS=r'\<'
t_MORE=r'\>'
t_ignore=' '

def t_NUMBER(t):
    r'\d+'
    t.value=int(t.value)
    return t

def t_ID(t):
    r'[a-zA-Z][a-zA-Z0-9]*'
    t.type = reserved.get(t.value, 'ID')
    return t

def t_error(t):
    print("Illegal character '%s'" % t.value[0])
    t.lexer.skip(1)

lexer = lex.lex()

def p_s(p):
    's : sw bl exp br cl ca exp co cl st brk cr ca exp co cl st brk cr def co cl
st brk cr cr'

def p_sw(p):
    'sw : switch'

def p_case(p):
    'ca : case'

def p_brk(p):
    'brk : break se'

def p_defa(p):
    'def : default'

def p_exp(p):
    'exp : id'

def p_exp1(p):
    'exp : no'

def p_exp2(p):
    'exp : id pl no'

```

```
def p_exp3(p):
    'exp : no pl id'

def p_exp4(p):
    'exp : no pl no'

def p_exp5(p):
    'exp : id mi no'

def p_exp6(p):
    'exp : no mi id'

def p_exp7(p):
    'exp : no mi no'

def p_se(p):
    'se : SEMICOLON'

def p_co(p):
    'co : COLON'

def p_id(p):
    'id : ID'

def p_eq(p):
    'eq : EQUAL'

def p_no(p):
    'no : NUMBER'

def p_cl(p):
    'cl : LCURLY'

def p_cr(p):
    'cr : RCURLY'

def p_bl(p):
    'bl : LPAREN'

def p_br(p):
    'br : RPAREN'

def p_st1(p):
    'st : id eq no se st'

def p_st2(p):
```

```

'st : id eq id se st'

def p_st3(p):
    'st : s'

def p_st4(p):
    'st : id pl pl se st'
def p_st5(p):
    'st : pl pl id se st'
def p_st6(p):
    'st : id mi mi se st'
def p_st7(p):
    'st : mi mi id se st'


def p_pl(p):
    'pl : PLUS'
def p_plm(p):
    'mi : MINUS'

def p_st8(p):
    'st : '


def p_gr(p):
    'gr : MORE'

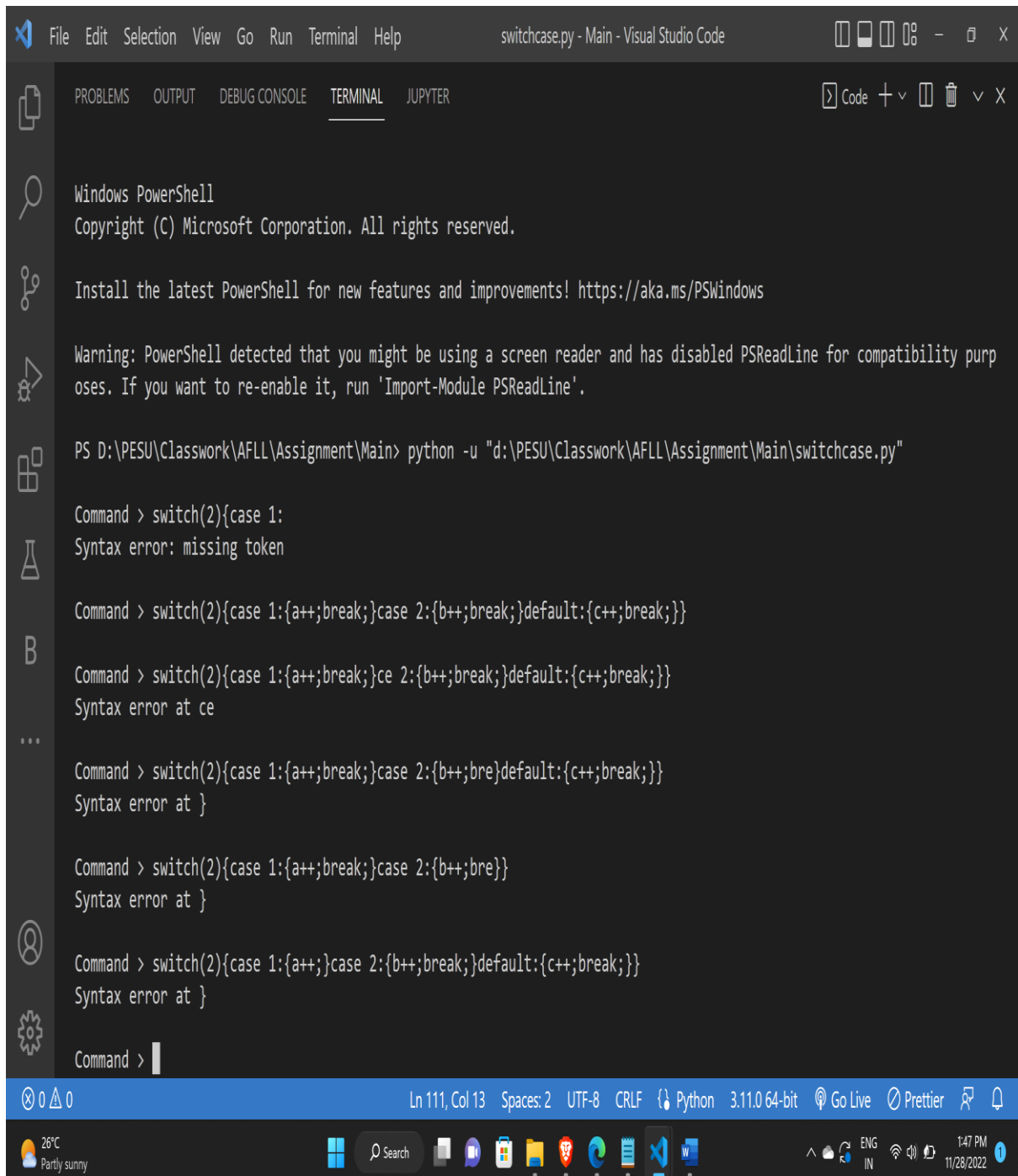
def p_le(p):
    'le : LESS'

def p_error(t):
    if(t):
        print("Syntax error at %s" %t.value)
    else:
        print("Syntax error: missing token")
parser = yacc.yacc()

while True:
    try:
        s = input('\nCommand > ')
        if s=='exit':
            print("\n")
            break
        else:
            parser.parse(s)
    except EOFError:
        break

```

OUTPUT:



The screenshot shows the Visual Studio Code interface with the 'TERMINAL' tab active. The terminal is running Windows PowerShell. The output shows the execution of a Python command to run a script named 'switchcase.py'. The script contains a switch statement with several syntax errors. The errors are as follows:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.

PS D:\PESU\Classwork\AFLL\Assignment\Main> python -u "d:\PESU\Classwork\AFLL\Assignment\Main\switchcase.py"

Command > switch(2){case 1:
Syntax error: missing token

Command > switch(2){case 1:{a++;break;}case 2:{b++;break;}default:{c++;break;}}

Command > switch(2){case 1:{a++;break;}ce 2:{b++;break;}default:{c++;break;}}
Syntax error at ce

...

Command > switch(2){case 1:{a++;break;}case 2:{b++;bre}default:{c++;break;}}
Syntax error at }

Command > switch(2){case 1:{a++;break;}case 2:{b++;bre}}
Syntax error at }

Command > switch(2){case 1:{a++;}case 2:{b++;break;}default:{c++;break;}}
Syntax error at }

Command > 
```

The status bar at the bottom indicates the current line and column (Ln 111, Col 13), the file encoding (UTF-8), the line ending (CRLF), the Python version (3.11.0 64-bit), and the Go Live extension. The system tray shows the temperature (26°C), the time (1:47 PM), and the date (11/28/2022).

2) WHILE LOOP

CFG FOR WHILE LOOP

$G = \{V, T, P, S\}$

$V = \{s, \text{condition}, \text{statement}\}$

$T = \{\text{while}, \text{ID}, \text{Number}, =, \{, \}, (,), ,, <=, >=\}$

$S = s$

$P = \text{productions}$

$s \rightarrow \text{while}(\text{condition})\{\text{statement}\}$

condition-

$> \text{ID} == \text{ID} \mid \text{Number} == \text{ID} \mid \text{ID} == \text{Number} \mid \text{Number} == \text{Number} \mid \text{ID} <= \text{Number} \mid \text{ID} >= \text{Number} \mid \text{ID} <= \text{ID} \mid \text{ID} >= \text{ID} \mid \text{ID} < \text{Number} \mid \text{ID} > \text{Number} \mid \text{ID} < \text{ID} \mid \text{ID} > \text{ID}$

statement- $\text{ID} = \text{ID}; \mid \text{ID} = \text{Number}; \mid \text{ID}++ \mid s \mid \text{statement}$

```
import ply.lex as lex
import ply.yacc as yacc

reserved={'while':'while'}

tokens=['EQUAL','LPAREN','RPAREN','NUMBER','ID','LCURLY','RCURLY','SEMICOLON',
'PLUS','LESS','MORE','MINUS']+list(reserved.values())

t_EQUAL=r'\='
t_LPAREN=r'\('
t_RPAREN=r'\)'
t_LCURLY=r'\{'
t_RCURLY=r'\}'
t_SEMICOLON=r'\;'
t_PLUS=r'\+'
t_MINUS=r'\-'
t_LESS=r'\<'
t_MORE=r'\>'
t_ignore=' '
```

```

def t_NUMBER(t):
    r'\d+'
    t.value=int(t.value)
    return t

def t_ID(t):
    r'[a-zA-Z][a-zA-Z0-9]*'
    t.type = reserved.get(t.value,'ID')
    return t

def t_error(t):
    print("Illegal character '%s'" % t.value[0])
    t.lexer.skip(1)

lexer = lex.lex()

def p_s(p):
    's : w bl con br cl st cr '

def p_w(p):
    'w : while'

def p_se(p):
    'se : SEMICOLON'

def p_con1(p):
    'con : id eq eq id'
def p_con2(p):
    'con : id eq eq no'
def p_con3(p):
    'con : no eq eq id'
def p_con4(p):
    'con : no eq eq no'
def p_con5(p):
    'con : id le eq no'
def p_con6(p):
    'con : id gr eq no'
def p_con7(p):
    'con : id le eq id'
def p_con8(p):
    'con : id gr eq id'
def p_id(p):
    'id : ID'

def p_eq(p):
    'eq : EQUAL'

def p_no(p):

```



```
'no : NUMBER'

def p_cl(p):
    'cl : LCURLY'

def p_cr(p):
    'cr : RCURLY'

def p_bl(p):
    'bl : LPAREN'

def p_br(p):
    'br : RPAREN'

def p_st1(p):
    'st : id eq no se st'

def p_st2(p):
    'st : id eq id se st'

def p_st3(p):
    'st : s'

def p_st4(p):
    'st : id pl pl se st'
def p_st5(p):
    'st : pl pl id se st'
def p_st6(p):
    'st : id mi mi se st'
def p_st7(p):
    'st : mi mi id se st'
def p_pl(p):
    'pl : PLUS'
def p_plm(p):
    'mi : MINUS'

def p_st8(p):
    'st : '

def p_gr(p):
    'gr : MORE'

def p_le(p):
    'le : LESS'

def p_error(t):
    if(t):
```

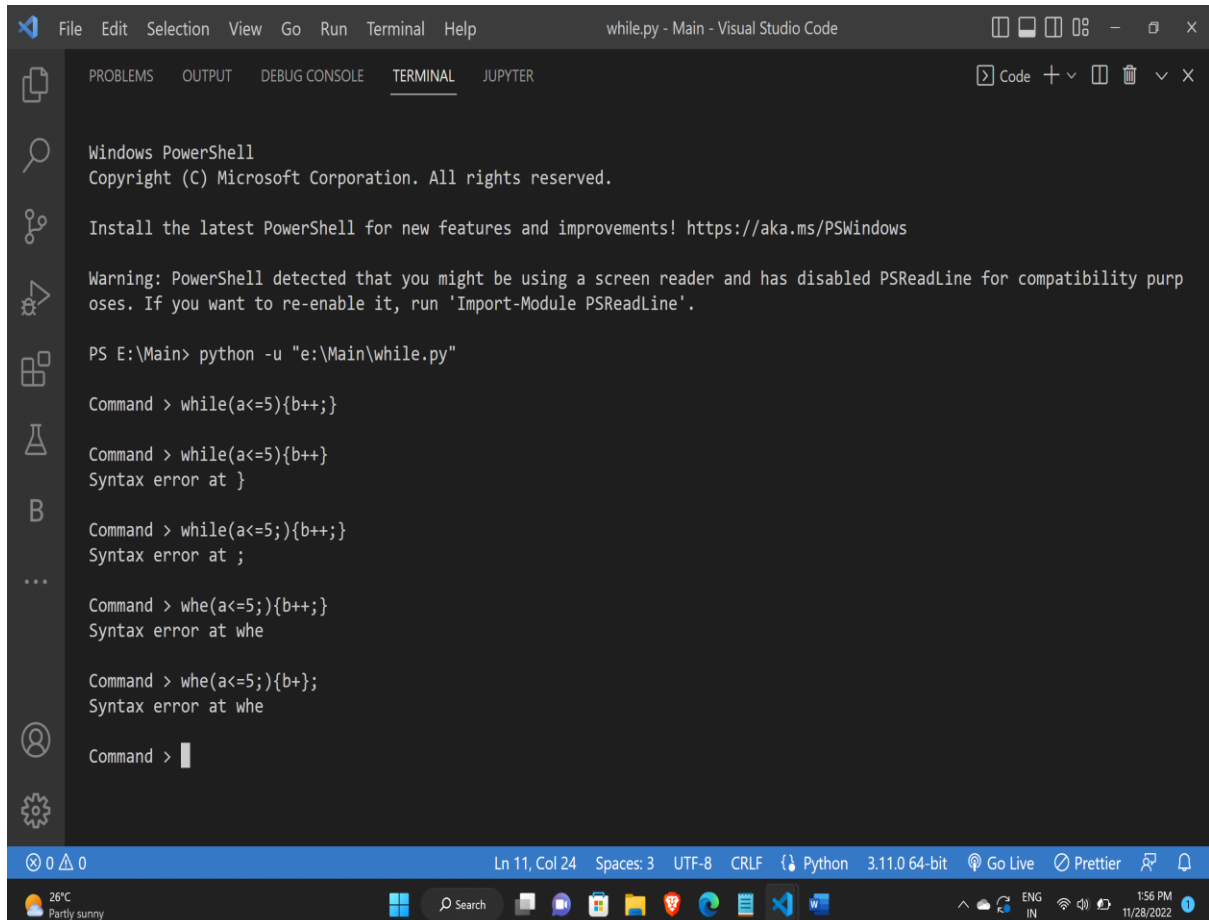
```

    print("Syntax error at %s" %t.value)
else:
    print("Syntax error: missing token")
parser = yacc.yacc()

while True:
    try:
        s = input('\nCommand > ')
        if s=='exit':
            print("\n")
            break
        else:
            parser.parse(s)
    except EOFError:
        break

```

OUTPUT:



```

File Edit Selection View Go Run Terminal Help while.py - Main - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL JUPYTER
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.

PS E:\Main> python -u "e:\Main\while.py"

Command > while(a<=5){b++;}

Command > while(a<=5){b++;}
Syntax error at }

Command > while(a<=5;){b++;}
Syntax error at ;

...

Command > whe(a<=5;){b++;}
Syntax error at whe

Command > whe(a<=5;){b++;}
Syntax error at whe

Command > 

```

Ln 11, Col 24 Spaces: 3 UTF-8 CRLF Python 3.11.0 64-bit Go Live Prettier 11/28/2022 1:56 PM

2) DO WHILE LOOP

CFG FOR DO WHILE LOOP

$G=\{V,T,P,S\}$

$V=\{s, \text{condition}, \text{statement}\}$

$T=\{\text{do}, \text{while}, \text{ID}, \text{Number}, =, \{, \}, (,), ,, <=, >=\}$

$S=s$

$P=\text{productions}$

$s \rightarrow \text{do}\{\text{statement}\}\text{while}(\text{condition})$

condition-

$> \text{ID} == \text{ID} \mid \text{Number} == \text{ID} \mid \text{ID} == \text{Number} \mid \text{Number} == \text{Number} \mid \text{ID} <= \text{Number} \mid \text{ID} >= \text{Number} \mid \text{ID} <= \text{ID} \mid \text{ID} >= \text{ID}$

statement- $\text{ID} = \text{ID}; \mid \text{ID} = \text{Number}; \mid \text{ID} ++ \mid s \mid \text{statement}$

```
import ply.lex as lex
import ply.yacc as yacc

reserved={'while':'while','do':'do'}

tokens=['EQUAL','LPAREN','RPAREN','NUMBER','ID','LCURLY','RCURLY','SEMICOLON',
'PLUS','LESS','MORE','MINUS']+list(reserved.values())

t_EQUAL=r'\='
t_LPAREN=r'\('
t_RPAREN=r'\)'
t_LCURLY=r'\{'
t_RCURLY=r'\}'
t_SEMICOLON=r'\;'
t_PLUS=r'\+'
t_MINUS=r'\-'
t_LESS=r'\<'
```

```
t_MORE=r'\>'
t_ignore=' '

def t_NUMBER(t):
    r'\d+'
    t.value=int(t.value)
    return t

def t_ID(t):
    r'[a-zA-Z][a-zA-Z0-9]*'
    t.type = reserved.get(t.value,'ID')
    return t

def t_error(t):
    print("Illegal character '%s'" % t.value[0])
    t.lexer.skip(1)

lexer = lex.lex()

def p_s(p):
    's : d cl st cr w bl con br se '

def p_do(p):
    'd : do'

def p_w(p):
    'w : while'

def p_se(p):
    'se : SEMICOLON'

def p_con1(p):
    'con : id eq eq id'
def p_con2(p):
    'con : id eq eq no'
def p_con3(p):
    'con : no eq eq id'
def p_con4(p):
    'con : no eq eq no'
def p_con5(p):
    'con : id le eq no'
def p_con6(p):
    'con : id gr eq no'
def p_con7(p):
    'con : id le eq id'
def p_con8(p):
    'con : id gr eq id'
```

```
def p_con9(p):
    'con : id le no'
def p_con10(p):
    'con : id gr no'
def p_con11(p):
    'con : id le id'
def p_con12(p):
    'con : id gr id'

def p_id(p):
    'id : ID'

def p_eq(p):
    'eq : EQUAL'

def p_no(p):
    'no : NUMBER'

def p_cl(p):
    'cl : LCURLY'

def p_cr(p):
    'cr : RCURLY'

def p_bl(p):
    'bl : LPAREN'

def p_br(p):
    'br : RPAREN'

def p_st1(p):
    'st : id eq no se st '

def p_st2(p):
    'st : id eq id se st'

def p_st3(p):
    'st : s'

def p_st4(p):
    'st : id pl pl se st'

def p_st5(p):
    'st : pl pl id se st '
```

```
def p_st6(p):
    'st : id mi mi se st '

def p_st7(p):
    'st : mi mi id se st '

def p_st10(p):
    'st : mi mi id st '

def p_st9(p):
    'st : id eq no st '

def p_pl(p):
    'pl : PLUS'
def p_plm(p):
    'mi : MINUS'

def p_st8(p):
    'st : '

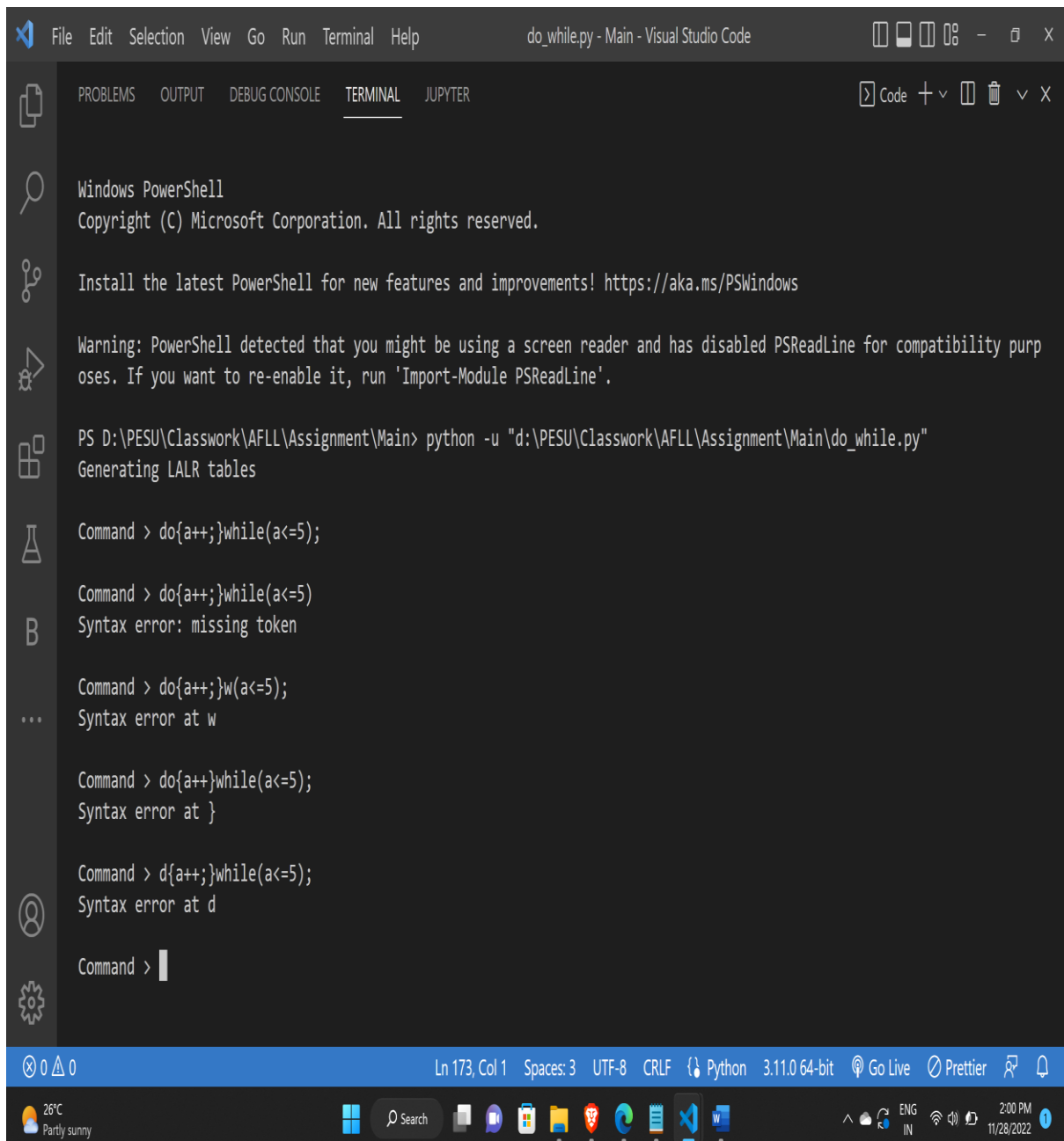
def p_gr(p):
    'gr : MORE'

def p_le(p):
    'le : LESS'

def p_error(t):
    if(t):
        print("Syntax error at %s" %t.value)
    else:
        print("Syntax error: missing token")
parser = yacc.yacc()

while True:
    try:
        s = input('\nCommand > ')
        if s=='exit':
            print("\n")
            break
        else:
            parser.parse(s)
    except EOFError:
        break
```

OUTPUT



The screenshot shows the Visual Studio Code interface with the 'Terminal' tab active. The terminal window displays the following output:

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

Warning: PowerShell detected that you might be using a screen reader and has disabled PSReadLine for compatibility purposes. If you want to re-enable it, run 'Import-Module PSReadLine'.

PS D:\PESU\Classwork\AFLL\Assignment\Main> python -u "d:\PESU\Classwork\AFLL\Assignment\Main\do_while.py"
Generating LALR tables

Command > do{a++;}while(a<=5);

Command > do{a++;}while(a<=5)
Syntax error: missing token

Command > do{a++;}w(a<=5);
Syntax error at w

Command > do{a++;}while(a<=5);
Syntax error at }

Command > d{a++;}while(a<=5);
Syntax error at d

Command > 
```

The status bar at the bottom indicates the current file is 'do_while.py' at line 173, column 1, with 3 spaces, UTF-8 encoding, and CRLF line endings. The Python version is 3.11.0 64-bit. The system tray shows a temperature of 26°C, partly sunny weather, and the date/time as 2:00 PM on 11/28/2022.

