**ANALIZADOR DE LÉXICO Y SINTAXIS**

# Instrucciones

**Diseñe un analizador de léxico y sintaxis que reconozca la frase**

hola, hola, hola …, hola que tal puede haber cualquier número de hola`s separados por comas, terminar sin coma o bien, hola que tal

**En este lenguaje sólo hay tokens, los puede o no manejar como palabras reservadas, no hay identificadores.**

**Puede usar la gramática:**

S->X que tal

X->X, hola

X->hola

**Utilice el generador de compiladores que corresponda al lenguaje de programación en el que vaya a programar su compilador (su proyecto).**

**Fecha de entrega: viernes 18 de septiembre, 2020**

Por favor, subir su código y tres screenshots, dos de corridas con strings correctos y una con un string incorrecto… a canvas sección de tareas,

Contents

[Instrucciones 1](#_Toc51232969)

[Desarrollo de la Tarea 2](#_Toc51232970)

[Investigación 2](#_Toc51232971)

[Codigo Funete 3](#_Toc51232972)

[Pruebas 4](#_Toc51232973)

# Desarrollo de la Tarea

## Investigación

* **Para conceptos Generales**
  + <https://www.youtube.com/watch?v=_PlRmlYEHRo&ab_channel=CodeEmporium>
* **Conceptos Generales de Generador de Parser de Python**
  + <https://www.youtube.com/watch?v=_PlRmlYEHRo&ab_channel=CodeEmporium>
* **Videos para construir el Léxico con Python**
  + <https://www.youtube.com/watch?v=Zbk0lic04SI&ab_channel=howCode>
* **Instalar Python en Computadora**
  + <https://www.python.org/>
* **Instalar herramienta de Lex y Yacc para Python**
  + <https://www.dabeaz.com/ply/>

## Codigo Funete

import ply.lex as lex

import ply.yacc as yacc

import sys

tokens = [

    'HOLA',

    'COMA',

    'QUE',

    'TAL'

]

t\_ignore = r' '

def t\_HOLA(t):

    r'hola'

    t.type = 'HOLA'

    print("HOLA")

    return t

def t\_COMA(t):

    r','

    t.type = 'COMA'

    print(",")

    return t

def t\_QUE(t):

    r'que'

    t.type = 'QUE'

    print("QUE")

    return t

def t\_TAL(t):

    r'tal'

    t.type = 'TAL'

    print("TAL")

    return t

def t\_error(t):

    print("Illegal characters!")

    t.lexer.skip(1)

lexer = lex.lex()

def p\_S(p):

    '''

    S : HOLA X

        | HOLA Y

        | HOLA

    '''

    print("\tCORRECTO")

def p\_X(p):

    '''

    X : COMA S

    '''

def p\_Y(p):

    '''

    Y : QUE TAL

    '''

def p\_error(p):

    print("\tINCORRECTO")

parser = yacc.yacc()

while True:

    try:

        s = input('')

    except EOFError:

        break

    parser.parse(s)

## Pruebas

hola

HOLA

CORRECTO

hola, hola, hola

HOLA

,

HOLA

,

HOLA

CORRECTO

CORRECTO

CORRECTO

hola que tal

HOLA

QUE

TAL

CORRECTO

hola, hola, hola que tal

HOLA

,

HOLA

,

HOLA

QUE

TAL

CORRECTO

CORRECTO

CORRECTO

que tal

QUE

INCORRECTO

TAL