KENDRIYA VIDHYALAYA

CHANDKHEDA, AHMEDABAD

तत् त्वं पूषन् अपावृणु

केन्द्रीय विद्यालय संगठन

PROJECT REPORT ON

ORGANIC CHEMISTRY DATABASE

SUBMITTED TO

MR. JITENDRA VARATIYA

P.G.T.(COMP. SC)

SUBMITTED BY:

HARSHIL SOLANKI

& VANSH KANODIYA

INDEX

CERTIFICATE

THIS IS TO CERTIFY THAT HARSHIL SOLANKI OF CLASS XII, ROLL NO 12117 HAS SUCCESSFULLY COMPLETED PROJECT TITLED ORGANIC CHEMISTRY DATABASE FOR SUBJECT OF COMPUTER SCIENCE NEW (083) FOR FULFILLMENT OF AISSCE PRACTICAL EXAMINATION 2022-23.

INTERNAL EXAMINER

EXTERNAL EXAMINER

PRINCIPAL

ACKNOWLEDGMENT

It is with pleasure that I acknowledge my sincere gratitude to our teacher, MR. JITENDRA VARATIYA SIR who taught and undertook the responsibility of teaching the subject computer science. I have greatly benefited from his classes.

I am especially indebted to our Principal MR. ASHOK RATHI SIR who has always been a source of encouragement and support and without whose inspiration this project would not have been a successful I would like to place on record heartfelt thanks to him.

Finally, I would like to express my sincere appreciation for all the other students of my batch, their friendship & the fine times that we all shared together.

INTRODUCTION

This project aims to provide a user-friendly python interface to deal with the chemical reactions, their reactants, reagents, products etc. of the organic chemistry branch of chemistry. Using this code, users will be able to search the database using any term of the reaction, which may prove helpful in solving questions, users would no longer have to search books or notebooks to find out the desired chemical reaction. The user will be able to expand the database by inserting new reactions into the database, and could save the changes in database into a sql file for extended use. The user will be able to insert brief remarks over the reaction, its mechanism and any exceptions for the reagents or reactants as per case.

The project is written completely in python programming language and it uses python interface at the front-end and the mysql database at the back-end. Function definitions are used to reduce the length of the code and provide a convenient way for the program to be inspected. It includes use of some mysql commands to fetch data or execute operations in the back-end, i.e. MySql. It is easy to use.

HARDWARE AND SOFTWARE REQUIREMENTS

- **Modern Operating System:** Windows 7 to 11
- **X86 64-bit CPU:** Intel/AMD Architecture
- **4 GB** minimum RAM Support
- **5 GB** minimum Free Disk Space (For Smooth Running of Program)
- MySQL should be supported on the operating system and Database should be accessible by the user.

DATABASE DESCRIPTION

The database is named **`Organic_chem`** containing the table **`Reactions`** containing Columns **`Name`** (with constraint NOT NULL), **`Reactant`**, **`Reagent`** (NOT NULL), **`Conditions`**, **`Product`**, **`Remarks`**. Column Remarks has datatype TEXT while all other columns have datatype TINYTEXT. TEXT is a data type with a field with a maximum length of 65535 characters. Sorts and comparisons on stored data are not case sensitive in TEXT fields. We do not need to specify a length with TEXT. TINYTEXT is Text datatype. It is a TEXT column with a maximum length of 255 characters. We do not need to specify a length with TINYTEXT.

CODE

```
import mysql.connector
import os
dir_path = os.path.dirname(os.path.realpath(__file__))
cnx = mysql.connector.connect(host='localhost', port='2836',
user='root', passwd='root', database = 'Organic_chem')
cur = cnx.cursor()
_format_ = ['Name', 'Reactant', 'Reagent', 'Conditions', 'Product',
'Remarks']


class helpers():
    def _print_rx_(rx):
        print(*('-->'+key+':- '+item for item, key in zip(rx,_format_)),
sep='\n')


    def search_term(term):
            '''OLD CODE DISCARDED
            term1 = input("Enter %s of the reaction you want to search:-
"%term)
            if term1 in rx[n]:
                print("The Reaction is as follows:")
                helpers._print_rx_(rx)
                return rx
            print("The Reaction having such %s could not be found"%term)
            return None
            '''
            term1 = input("Enter %s of the reaction you want to search:-
"%term)
            cur.execute("select * from Reactions where {} LIKE
'%{}%';".format(term, term1))
            rx = cur.fetchall()
            if rx == []:
                print("The Reaction having such %s could not be
found"%term)
                helpers._continue_()
                return None
            elif len(rx)== 1:
                helpers._print_rx_(rx[0])
                helpers._continue_()
                return rx[0]
            else:
                n = len(rx)
                print('There are the following {} results: '.format(n))
                helpers._continue_()
                for i in range(0, n):
                    helpers.page_break()
                    print('<REACTION {}>'.format(i+1))
                    helpers._print_rx_(rx[i])
                    helpers._continue_()
                return rx

    def change_term(term, name):
        term2 = input("Enter new %s for the Reaction:- "%term)
        cur.execute("update Reactions set %s='%s' where
'Name'='%s';"%(term, term2, name))
```

```
        cnx.commit()
        print("Reaction successfully updated.")
        return True

    def page_break():
        print('-'*120)

    def _continue_():
        input("Press ENTER to Continue")

class Ops():
    def insert_rx():
        print('<INSERTING A REACTION>')
        v1 = str(input("Enter name:- "))
        v2 = str(input("Enter reactant(s):- "))
        v3 = str(input("Enter reagent(s):- "))
        v4 = str(input("Enter condition(s):- "))
        v5 = str(input("Enter product(s):- "))
        v6 = str(input("Remarks over the reaction (mechanism,
explanation, reactivity,...):-   "))
        st = "insert into Reactions
values('{}','{}','{}','{}','{}','{}');".format(v1,v2,v3,v4,v5,v6)
        cur.execute(st)
        cnx.commit()
        print("Reaction successfully inserted into the database.")
        return True

    def list_db():
        print('<LISTING THE DATABASE>')
        cur.execute('select * from Reactions;')
        data = cur.fetchall()
        i = 1
        for rx in data:
            print('<Reaction %d>'%i)
            helpers._print_rx_(rx)
            helpers.page_break()
            i+=1
        if data == []:
            print("Database is Empty.")
        helpers._continue_()
        return

    def search_db():
        print('<SEARCHING THE DATABASE>')
        print('''By which parameter would you like to search for
reaction?
    1. Name         2. Reactant     3. Reagent              4.
Conditions
    5. Product      (6. Remarks (searches same as 7.))      7. Search in
all
        ''')
        '''OLD CODE DISCARDED
        cur.execute('select * from Reactions;')
        data = cur.fetchall()'''
        i = input("Enter your choice (1-6):-  ")
        helpers.page_break()
        #for rx in data:
```

```
    if int(i) == 1:
        return helpers.search_term('Name')
    if int(i) == 2:
        return helpers.search_term('Reactant')
    if int(i) == 3:
        return helpers.search_term('Reagent')
    if int(i) == 4:
        return helpers.search_term('Conditions')
    if int(i) == 5:
        return helpers.search_term('Product')
    if int(i) == 6 or 7:
        a = input("Enter the term present in the reaction:-   ")
        cur.execute('select * from Reactions;')
        data = cur.fetchall()
        for rx in data:
            for el in rx:
                if a in el:
                    print("The Reaction is as follows:")
                    helpers._print_rx_(rx)
                    return rx
        print("None of the Reactions contain this term")
        return None
    else:
        print("There was an error in the input, execute again")
        return None

def update_db():
        print('<UPDATING A REACTION')
        print("First search for the reaction you want to update: ")

        def sub_update(rx):
            '''COMMON PROCESS TO BE USED'''
            # rx = list(rx) ---Used in old code, no longer needed.
            i = int(input("Enter the term you want to update(1-6):- "))
            if i == 1:
                helpers.change_term('Name', rx[0])
            if i == 2:
                helpers.change_term('Reactant', rx[0])
            if i == 3:
                helpers.change_term('Reagent', rx[0])
            if i == 4:
                helpers.change_term('Conditions', rx[0])
            if i == 5:
                helpers.change_term('Product', rx[0])
            if i == 6:
                helpers.change_term('Remarks', rx[0])
            helpers._continue_()
            return

        try:
            rx = Ops.search_db()
            if type(rx[0])== tuple:
                n = int(input('Which one of the reactions would you like
to update?(1-{}):- '.format(len(rx))))
                return sub_update(rx[n-1])
            else:
                return sub_update(rx)

        except ValueError or TypeError:
            return

    def delete_rx():
        print('<DELETING A REACTION>')
        print("First search for the reaction you want to delete: ")

        def sub_del(rx):
            cur.execute("delete from Reactions where
{}='{}';".format('Name', rx[0]))
            cnx.commit()
            helpers.page_break()
            print("Reaction successfully deleted.")
            helpers._continue_()
            return

        try:
            rx = Ops.search_db()
            if type(rx[0])==tuple:
                n = int(input('Which one of the reactions would you like
to delete?(1-{}):- '.format(len(rx))))
                return sub_del(rx[n-1])
            else:
                return sub_del(rx)
        except ValueError or TypeError:
            return

    def Naming():
        f = open('{}\\datafiles\\IUPAC_A.txt'.format(dir_path), 'r')
        data = f.read()
        print(data)
        f.close()
        helpers._continue_()
        helpers.page_break()
        f = open('{}\\datafiles\\IUPAC_B.txt'.format(dir_path), 'r')
        data = f.read()
        print(data)
        return

    def OR():
        f = open('{}\\datafiles\\OR_C.txt'.format(dir_path), 'r')
        data = f.read()
        print(data)
        f.close()
        helpers._continue_()
        helpers.page_break()
        f = open('{}\\datafiles\\OR_D.txt'.format(dir_path), 'r')
        data = f.read()
        print(data)
        return
```

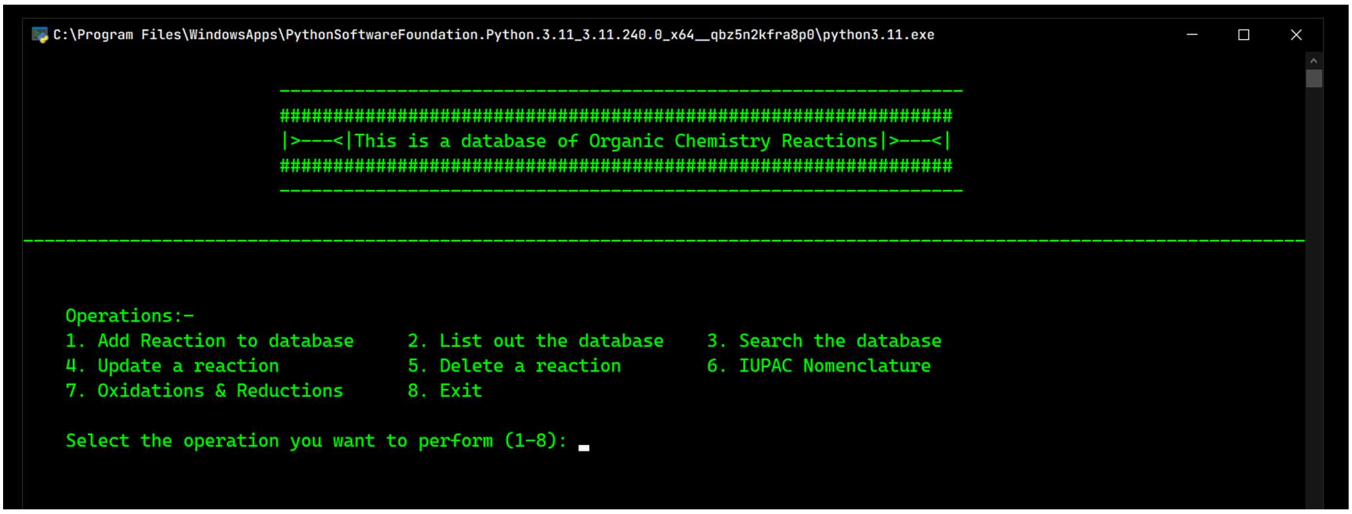
```
class Interface():
    def operations():
        helpers.page_break()
        print('''
    Operations:-
    1. Add Reaction to database     2. List out the database    3.
Search the database
    4. Update a reaction            5. Delete a reaction        6. IUPAC
Nomenclature
    7. Oxidations & Reductions      8. Exit

    Select the operation you want to perform (1-8): ''', end='')

        try:
            i = int(input())
            helpers.page_break()
            if i == 1:
                return Ops.insert_rx()
            elif i == 2:
                return Ops.list_db()
            elif i == 3:
                return Ops.search_db()
            elif i == 4:
                return Ops.update_db()
            elif i == 5:
                return Ops.delete_rx()
            elif i == 6:
                return Ops.Naming()
            elif i == 7:
                return Ops.OR()
            elif i == 8:
                global consent
                consent = False
                return

        except TypeError:
            print("There was error in the input. Execute again.")
            return

    def _init_():
        print('''
                        ------------------------------------------------
----------------

###############################################################
                        |>---<|This is a database of Organic Chemistry
Reactions|>---<|

###############################################################
                        ------------------------------------------------
----------------
    ''')
        global consent
        consent = True
        while consent == True:
            Interface.operations()
```

```
#INITIATION OF CODE

Interface._init_()
```

—---**CODE END**—---

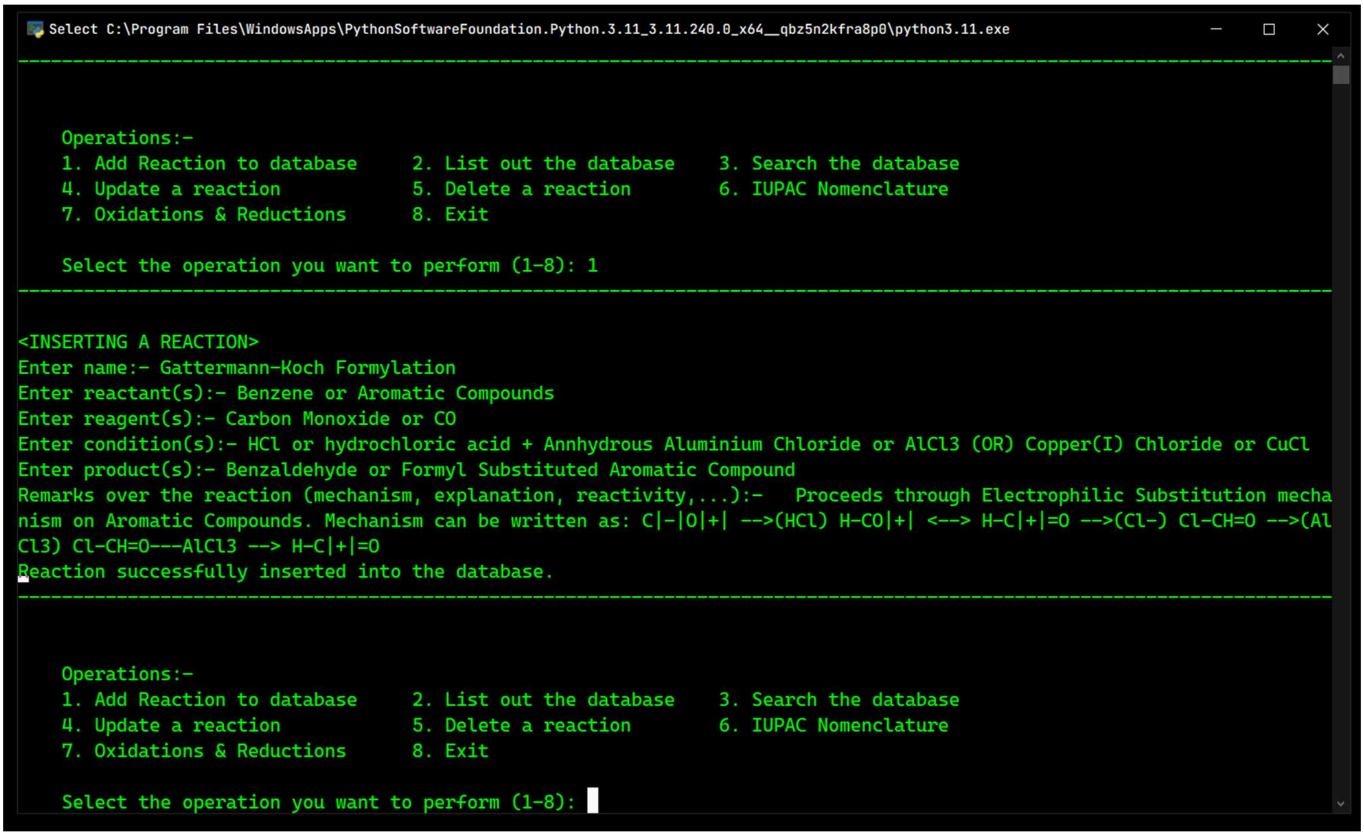
OUTPUT

- Start



```
C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe

                        ----------------------------------------------------------------
                        ###############################################################
                        |>---<|This is a database of Organic Chemistry Reactions|>---<|
                        ###############################################################
                        ----------------------------------------------------------------

------------------------------------------------------------------------------------------------------------------------


    Operations:-
    1. Add Reaction to database      2. List out the database     3. Search the database
    4. Update a reaction             5. Delete a reaction         6. IUPAC Nomenclature
    7. Oxidations & Reductions       8. Exit

    Select the operation you want to perform (1-8): _
```

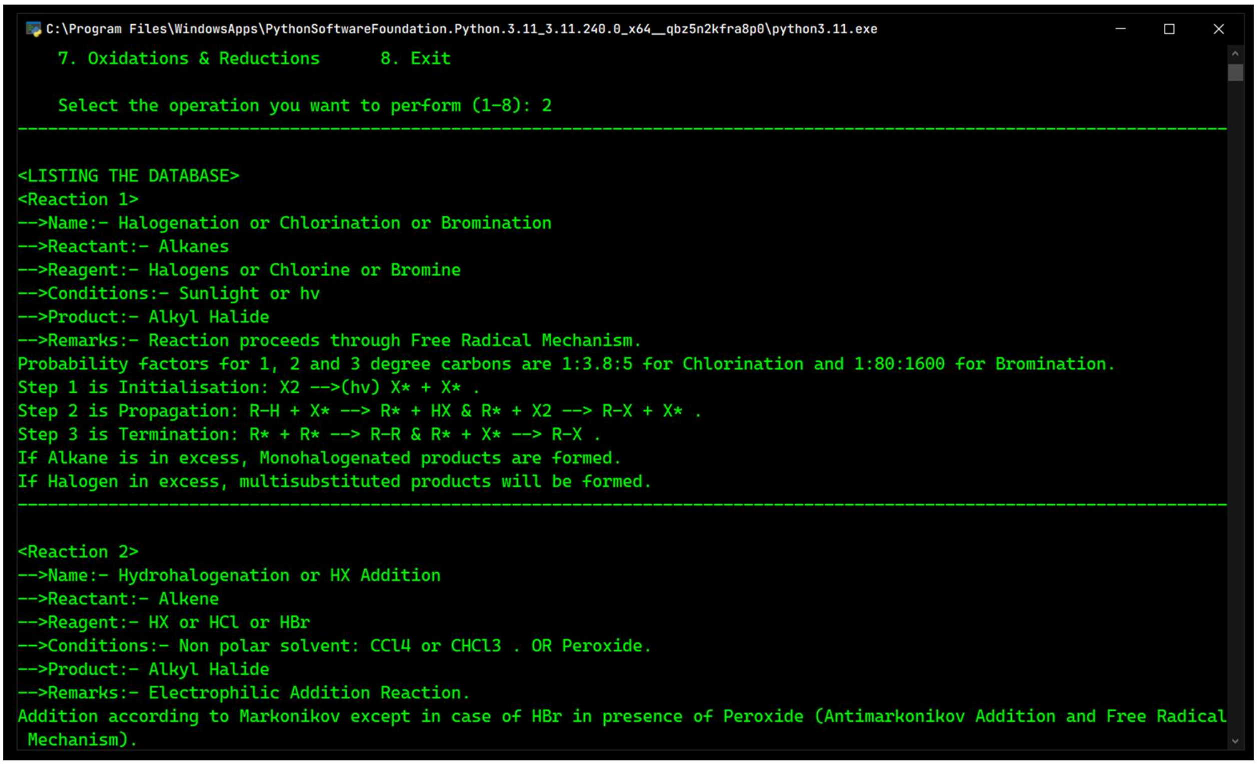
- Operation 1



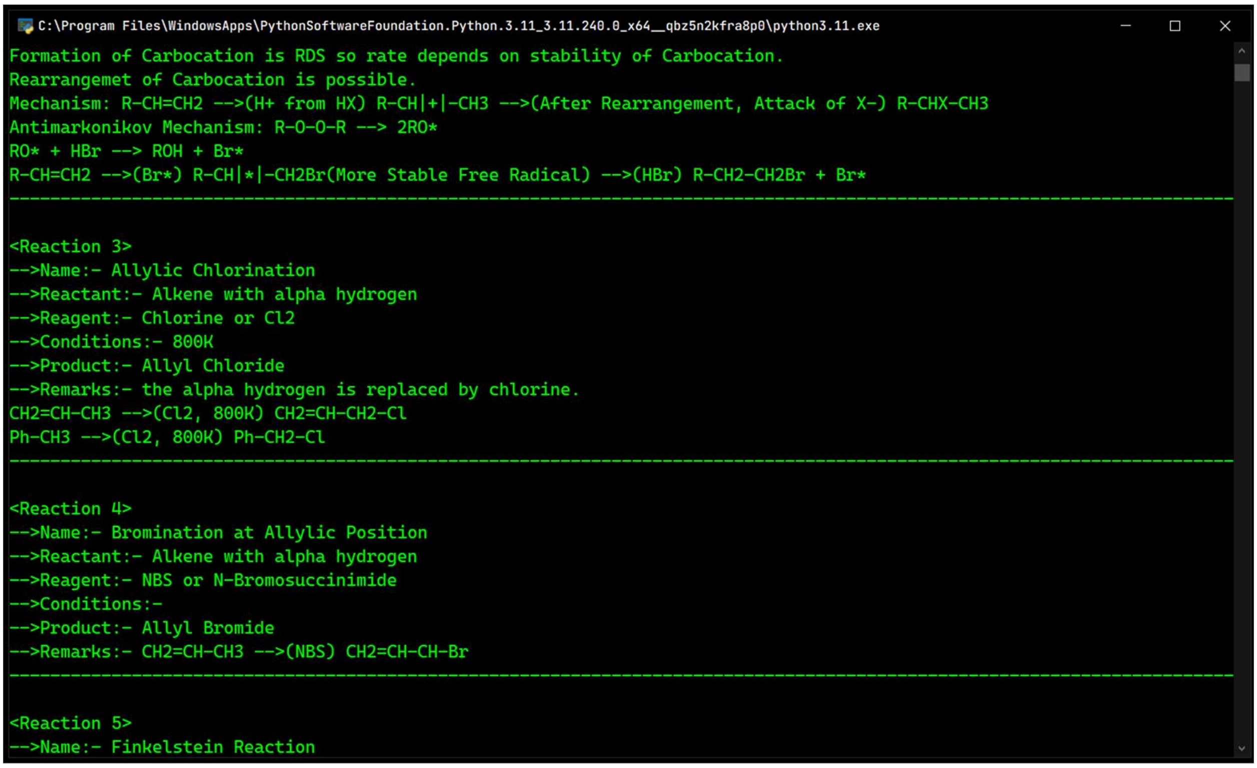
Select C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe

```
------------------------------------------------------------------------------------------------------------------------



    Operations:-
    1. Add Reaction to database     2. List out the database    3. Search the database
    4. Update a reaction            5. Delete a reaction        6. IUPAC Nomenclature
    7. Oxidations & Reductions      8. Exit

    Select the operation you want to perform (1-8): 1
------------------------------------------------------------------------------------------------------------------------

<INSERTING A REACTION>
Enter name:- Gattermann-Koch Formylation
Enter reactant(s):- Benzene or Aromatic Compounds
Enter reagent(s):- Carbon Monoxide or CO
Enter condition(s):- HCl or hydrochloric acid + Annhydrous Aluminium Chloride or AlCl3 (OR) Copper(I) Chloride or CuCl
Enter product(s):- Benzaldehyde or Formyl Substituted Aromatic Compound
Remarks over the reaction (mechanism, explanation, reactivity,...):-   Proceeds through Electrophilic Substitution mecha
nism on Aromatic Compounds. Mechanism can be written as: C|-|O|+| -->(HCl) H-CO|+| <--> H-C|+|=O -->(Cl-) Cl-CH=O -->(Al
Cl3) Cl-CH=O---AlCl3 --> H-C|+|=O
Reaction successfully inserted into the database.
------------------------------------------------------------------------------------------------------------------------



    Operations:-
    1. Add Reaction to database     2. List out the database    3. Search the database
    4. Update a reaction            5. Delete a reaction        6. IUPAC Nomenclature
    7. Oxidations & Reductions      8. Exit

    Select the operation you want to perform (1-8):
```

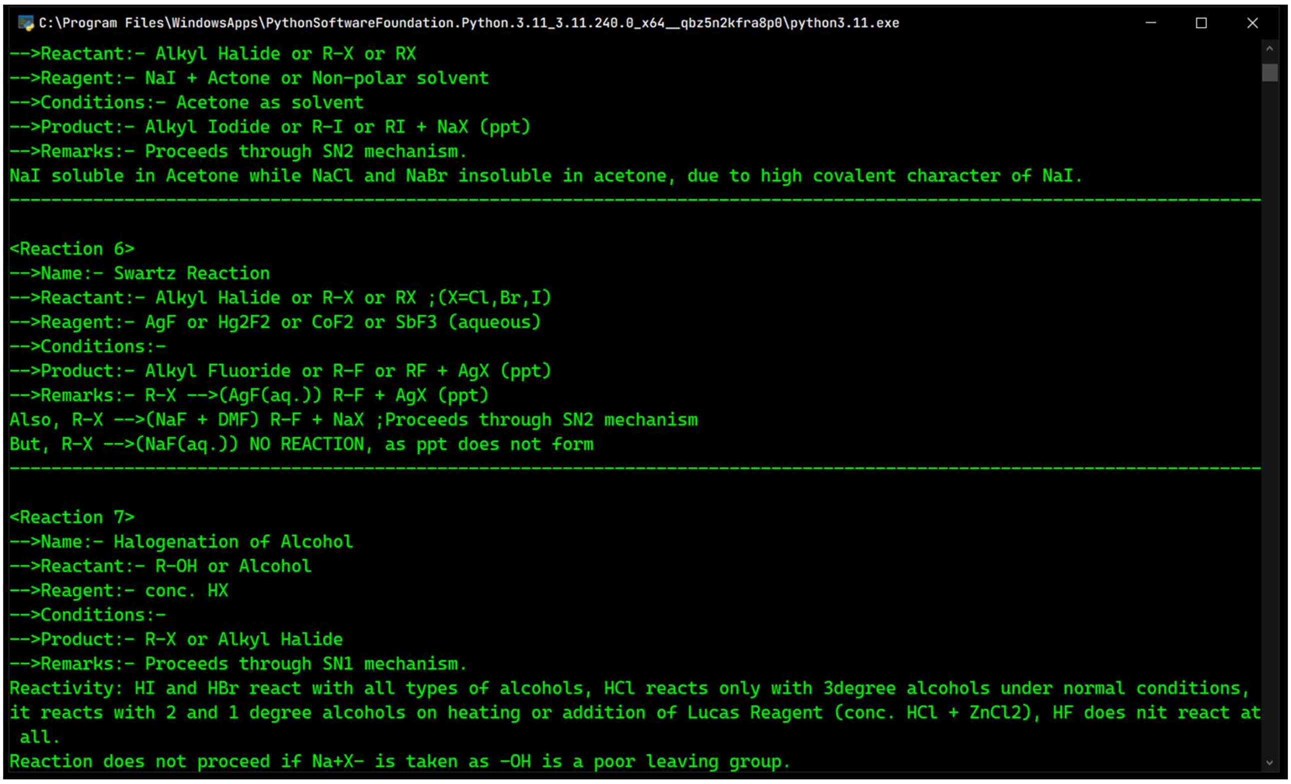
● Operation 2


C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe

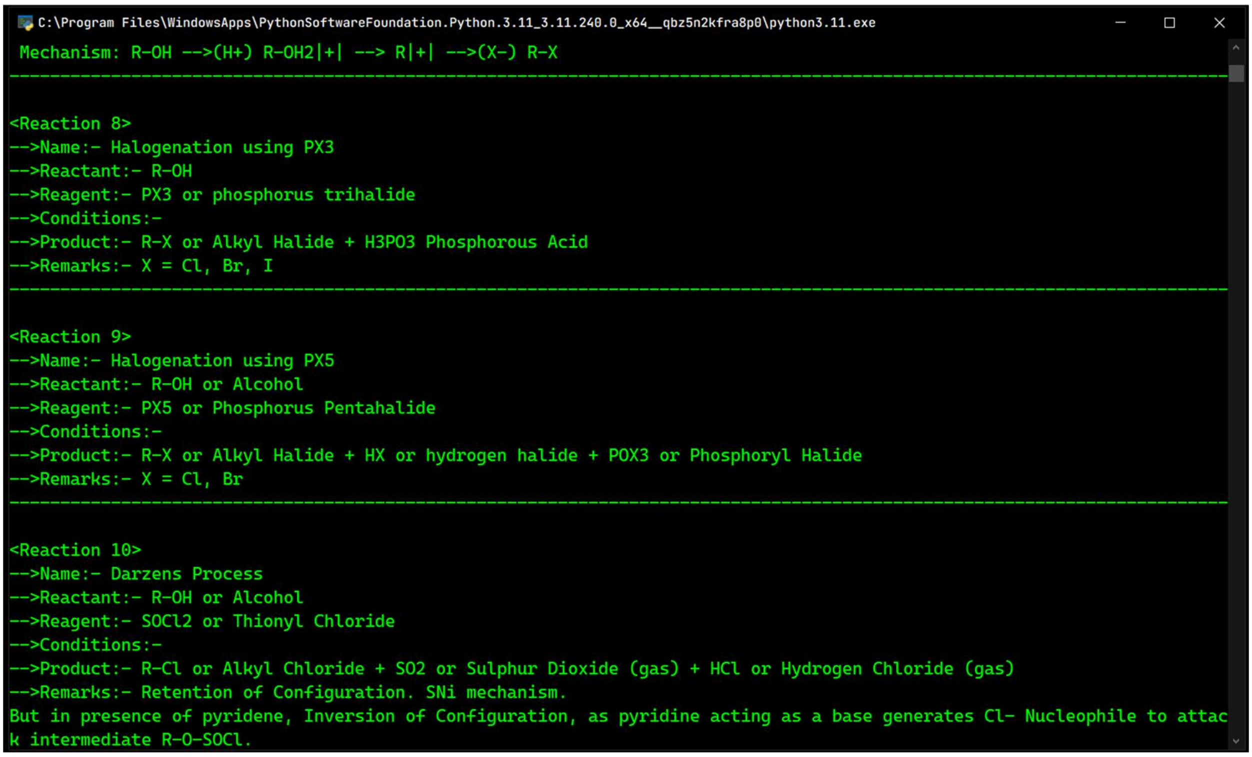
```
    7. Oxidations & Reductions      8. Exit

    Select the operation you want to perform (1-8): 2
------------------------------------------------------------------------------------------------------------------------

<LISTING THE DATABASE>
<Reaction 1>
-->Name:- Halogenation or Chlorination or Bromination
-->Reactant:- Alkanes
-->Reagent:- Halogens or Chlorine or Bromine
-->Conditions:- Sunlight or hv
-->Product:- Alkyl Halide
-->Remarks:- Reaction proceeds through Free Radical Mechanism.
Probability factors for 1, 2 and 3 degree carbons are 1:3.8:5 for Chlorination and 1:80:1600 for Bromination.
Step 1 is Initialisation: X2 -->(hv) X* + X* .
Step 2 is Propagation: R-H + X* --> R* + HX & R* + X2 --> R-X + X* .
Step 3 is Termination: R* + R* --> R-R & R* + X* --> R-X .
If Alkane is in excess, Monohalogenated products are formed.
If Halogen in excess, multisubstituted products will be formed.
------------------------------------------------------------------------------------------------------------------------

<Reaction 2>
-->Name:- Hydrohalogenation or HX Addition
-->Reactant:- Alkene
-->Reagent:- HX or HCl or HBr
-->Conditions:- Non polar solvent: CCl4 or CHCl3 . OR Peroxide.
-->Product:- Alkyl Halide
-->Remarks:- Electrophilic Addition Reaction.
Addition according to Markonikov except in case of HBr in presence of Peroxide (Antimarkonikov Addition and Free Radical
 Mechanism).
```


C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe

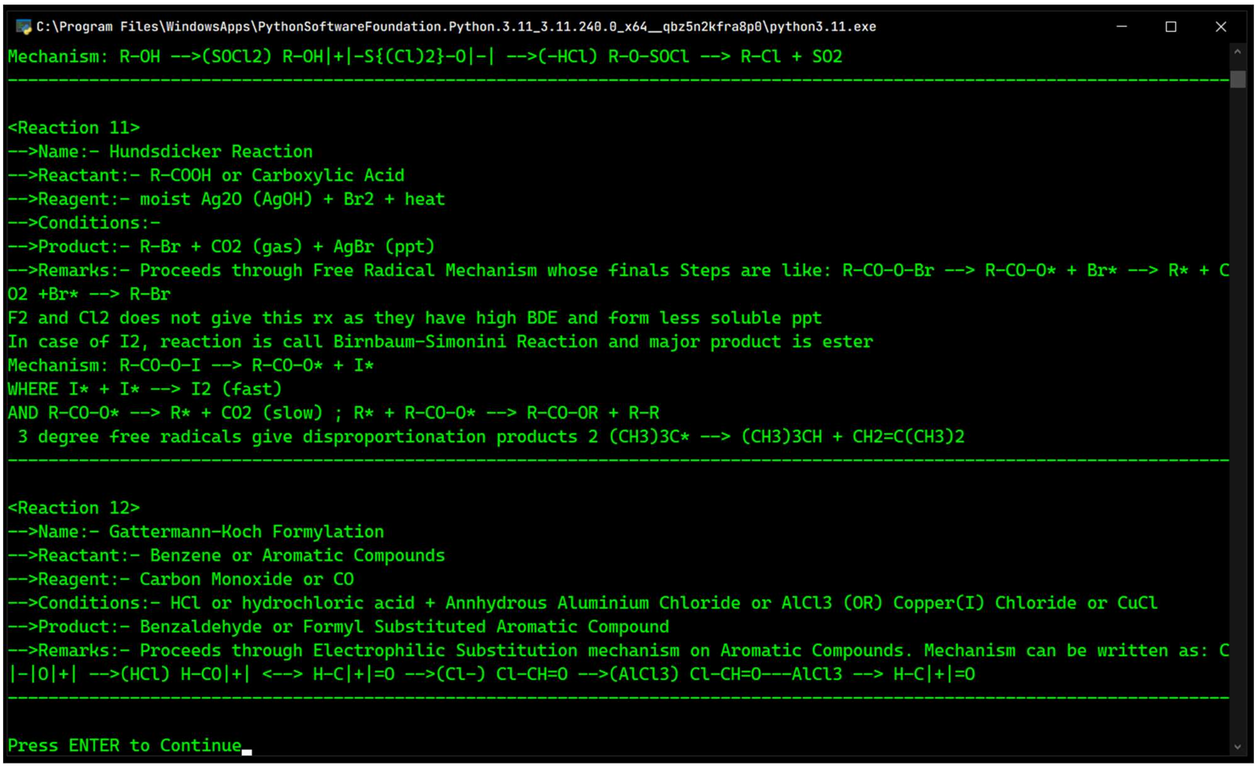
```
Formation of Carbocation is RDS so rate depends on stability of Carbocation.
Rearrangemet of Carbocation is possible.
Mechanism: R-CH=CH2 -->(H+ from HX) R-CH|+|-CH3 -->(After Rearrangement, Attack of X-) R-CHX-CH3
Antimarkonikov Mechanism: R-O-O-R --> 2RO*
RO* + HBr --> ROH + Br*
R-CH=CH2 -->(Br*) R-CH|*|-CH2Br(More Stable Free Radical) -->(HBr) R-CH2-CH2Br + Br*
------------------------------------------------------------------------------------------------------------------------

<Reaction 3>
-->Name:- Allylic Chlorination
-->Reactant:- Alkene with alpha hydrogen
-->Reagent:- Chlorine or Cl2
-->Conditions:- 800K
-->Product:- Allyl Chloride
-->Remarks:- the alpha hydrogen is replaced by chlorine.
CH2=CH-CH3 -->(Cl2, 800K) CH2=CH-CH2-Cl
Ph-CH3 -->(Cl2, 800K) Ph-CH2-Cl
------------------------------------------------------------------------------------------------------------------------

<Reaction 4>
-->Name:- Bromination at Allylic Position
-->Reactant:- Alkene with alpha hydrogen
-->Reagent:- NBS or N-Bromosuccinimide
-->Conditions:-
-->Product:- Allyl Bromide
-->Remarks:- CH2=CH-CH3 -->(NBS) CH2=CH-CH-Br
------------------------------------------------------------------------------------------------------------------------

<Reaction 5>
-->Name:- Finkelstein Reaction
```


C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe

```
-->Reactant:- Alkyl Halide or R-X or RX
-->Reagent:- NaI + Actone or Non-polar solvent
-->Conditions:- Acetone as solvent
-->Product:- Alkyl Iodide or R-I or RI + NaX (ppt)
-->Remarks:- Proceeds through SN2 mechanism.
NaI soluble in Acetone while NaCl and NaBr insoluble in acetone, due to high covalent character of NaI.
------------------------------------------------------------------------------------------------------------------------


<Reaction 6>
-->Name:- Swartz Reaction
-->Reactant:- Alkyl Halide or R-X or RX ;(X=Cl,Br,I)
-->Reagent:- AgF or Hg2F2 or CoF2 or SbF3 (aqueous)
-->Conditions:-
-->Product:- Alkyl Fluoride or R-F or RF + AgX (ppt)
-->Remarks:- R-X -->(AgF(aq.)) R-F + AgX (ppt)
Also, R-X -->(NaF + DMF) R-F + NaX ;Proceeds through SN2 mechanism
But, R-X -->(NaF(aq.)) NO REACTION, as ppt does not form
------------------------------------------------------------------------------------------------------------------------


<Reaction 7>
-->Name:- Halogenation of Alcohol
-->Reactant:- R-OH or Alcohol
-->Reagent:- conc. HX
-->Conditions:-
-->Product:- R-X or Alkyl Halide
-->Remarks:- Proceeds through SN1 mechanism.
Reactivity: HI and HBr react with all types of alcohols, HCl reacts only with 3degree alcohols under normal conditions,
it reacts with 2 and 1 degree alcohols on heating or addition of Lucas Reagent (conc. HCl + ZnCl2), HF does nit react at
 all.
Reaction does not proceed if Na+X- is taken as -OH is a poor leaving group.
```

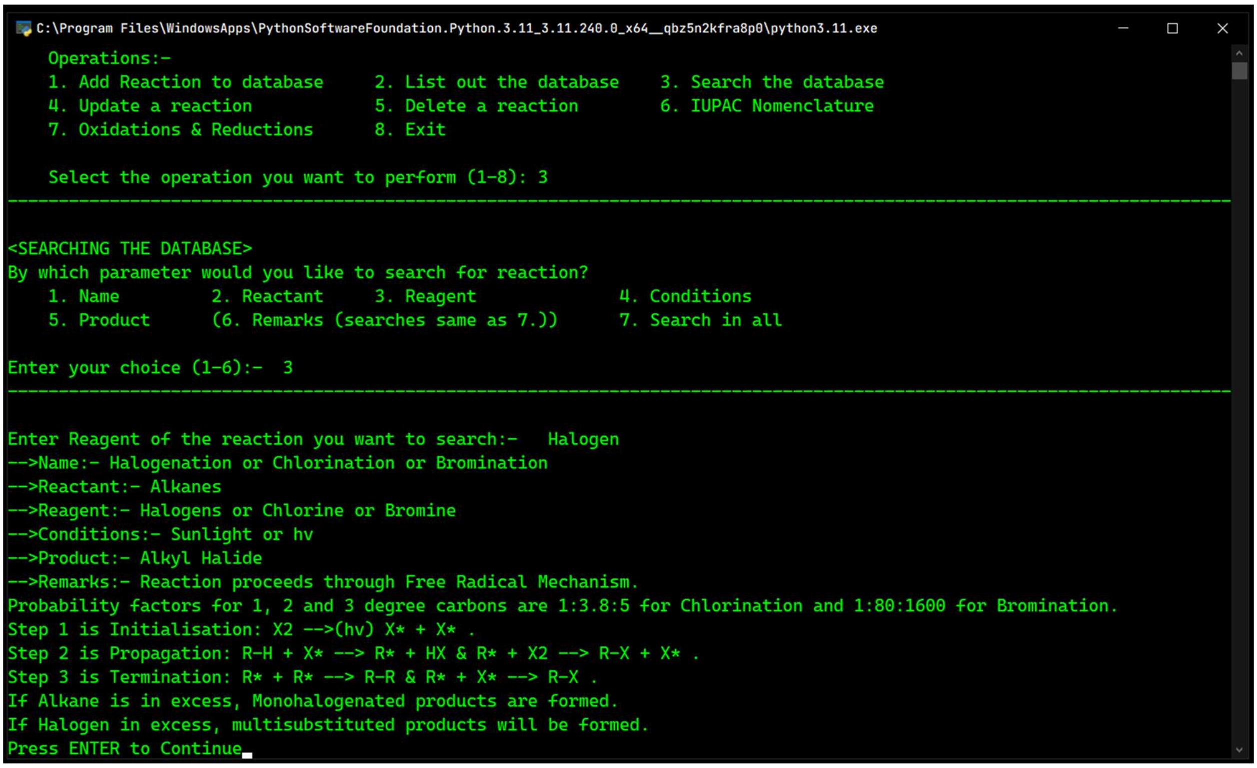

C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe

```
 Mechanism: R-OH -->(H+) R-OH2|+| --> R|+| -->(X-) R-X
------------------------------------------------------------------------------------------------------------------------


<Reaction 8>
-->Name:- Halogenation using PX3
-->Reactant:- R-OH
-->Reagent:- PX3 or phosphorus trihalide
-->Conditions:-
-->Product:- R-X or Alkyl Halide + H3PO3 Phosphorous Acid
-->Remarks:- X = Cl, Br, I
------------------------------------------------------------------------------------------------------------------------


<Reaction 9>
-->Name:- Halogenation using PX5
-->Reactant:- R-OH or Alcohol
-->Reagent:- PX5 or Phosphorus Pentahalide
-->Conditions:-
-->Product:- R-X or Alkyl Halide + HX or hydrogen halide + POX3 or Phosphoryl Halide
-->Remarks:- X = Cl, Br
------------------------------------------------------------------------------------------------------------------------


<Reaction 10>
-->Name:- Darzens Process
-->Reactant:- R-OH or Alcohol
-->Reagent:- SOCl2 or Thionyl Chloride
-->Conditions:-
-->Product:- R-Cl or Alkyl Chloride + SO2 or Sulphur Dioxide (gas) + HCl or Hydrogen Chloride (gas)
-->Remarks:- Retention of Configuration. SNi mechanism.
But in presence of pyridene, Inversion of Configuration, as pyridine acting as a base generates Cl- Nucleophile to attac
k intermediate R-O-SOCl.
```


C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe

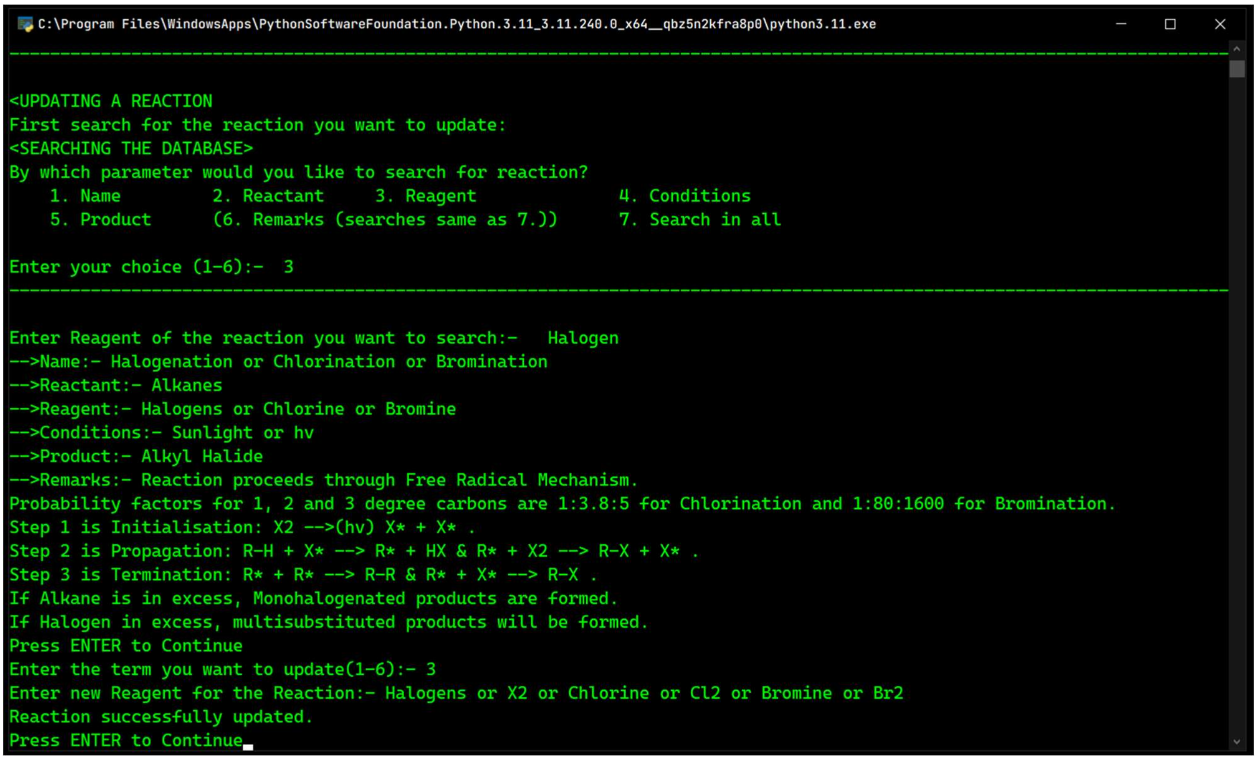
```
Mechanism: R-OH -->(SOCl2) R-OH|+|-S{(Cl)2}-O|-| -->(-HCl) R-O-SOCl --> R-Cl + SO2
------------------------------------------------------------------------------------------------------------------------

<Reaction 11>
-->Name:- Hundsdicker Reaction
-->Reactant:- R-COOH or Carboxylic Acid
-->Reagent:- moist Ag2O (AgOH) + Br2 + heat
-->Conditions:-
-->Product:- R-Br + CO2 (gas) + AgBr (ppt)
-->Remarks:- Proceeds through Free Radical Mechanism whose finals Steps are like: R-CO-O-Br --> R-CO-O* + Br* --> R* + C
O2 +Br* --> R-Br
F2 and Cl2 does not give this rx as they have high BDE and form less soluble ppt
In case of I2, reaction is call Birnbaum-Simonini Reaction and major product is ester
Mechanism: R-CO-O-I --> R-CO-O* + I*
WHERE I* + I* --> I2 (fast)
AND R-CO-O* --> R* + CO2 (slow) ; R* + R-CO-O* --> R-CO-OR + R-R
 3 degree free radicals give disproportionation products 2 (CH3)3C* --> (CH3)3CH + CH2=C(CH3)2
------------------------------------------------------------------------------------------------------------------------

<Reaction 12>
-->Name:- Gattermann-Koch Formylation
-->Reactant:- Benzene or Aromatic Compounds
-->Reagent:- Carbon Monoxide or CO
-->Conditions:- HCl or hydrochloric acid + Annhydrous Aluminium Chloride or AlCl3 (OR) Copper(I) Chloride or CuCl
-->Product:- Benzaldehyde or Formyl Substituted Aromatic Compound
-->Remarks:- Proceeds through Electrophilic Substitution mechanism on Aromatic Compounds. Mechanism can be written as: C
|-|O|+| -->(HCl) H-CO|+| <--> H-C|+|=O -->(Cl-) Cl-CH=O -->(AlCl3) Cl-CH=O---AlCl3 --> H-C|+|=O
------------------------------------------------------------------------------------------------------------------------

Press ENTER to Continue
```

● Operation 3


C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe

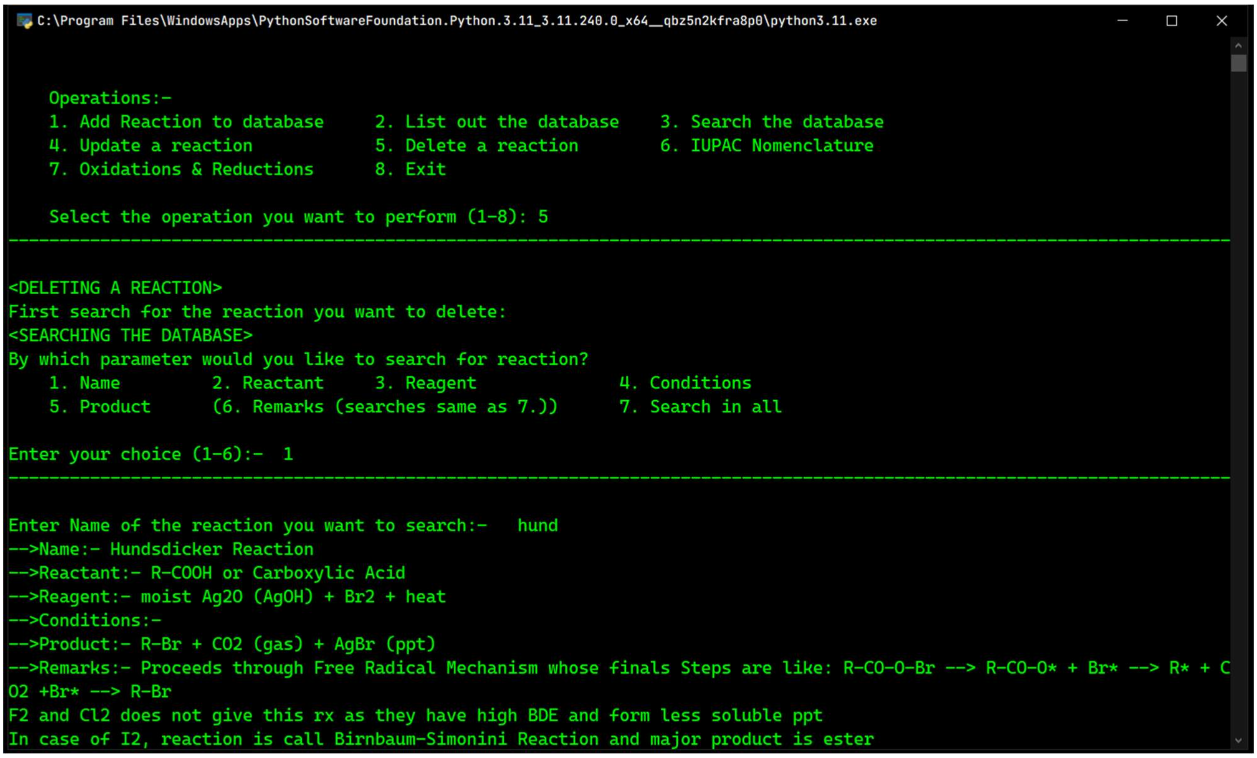
```
    Operations:-
    1. Add Reaction to database     2. List out the database    3. Search the database
    4. Update a reaction            5. Delete a reaction        6. IUPAC Nomenclature
    7. Oxidations & Reductions      8. Exit

    Select the operation you want to perform (1-8): 3
------------------------------------------------------------------------------------------------------------------------

<SEARCHING THE DATABASE>
By which parameter would you like to search for reaction?
    1. Name         2. Reactant     3. Reagent              4. Conditions
    5. Product      (6. Remarks (searches same as 7.))      7. Search in all

Enter your choice (1-6):-  3
------------------------------------------------------------------------------------------------------------------------

Enter Reagent of the reaction you want to search:-   Halogen
-->Name:- Halogenation or Chlorination or Bromination
-->Reactant:- Alkanes
-->Reagent:- Halogens or Chlorine or Bromine
-->Conditions:- Sunlight or hv
-->Product:- Alkyl Halide
-->Remarks:- Reaction proceeds through Free Radical Mechanism.
Probability factors for 1, 2 and 3 degree carbons are 1:3.8:5 for Chlorination and 1:80:1600 for Bromination.
Step 1 is Initialisation: X2 -->(hv) X* + X* .
Step 2 is Propagation: R-H + X* --> R* + HX & R* + X2 --> R-X + X* .
Step 3 is Termination: R* + R* --> R-R & R* + X* --> R-X .
If Alkane is in excess, Monohalogenated products are formed.
If Halogen in excess, multisubstituted products will be formed.
Press ENTER to Continue
```

● Operation 4

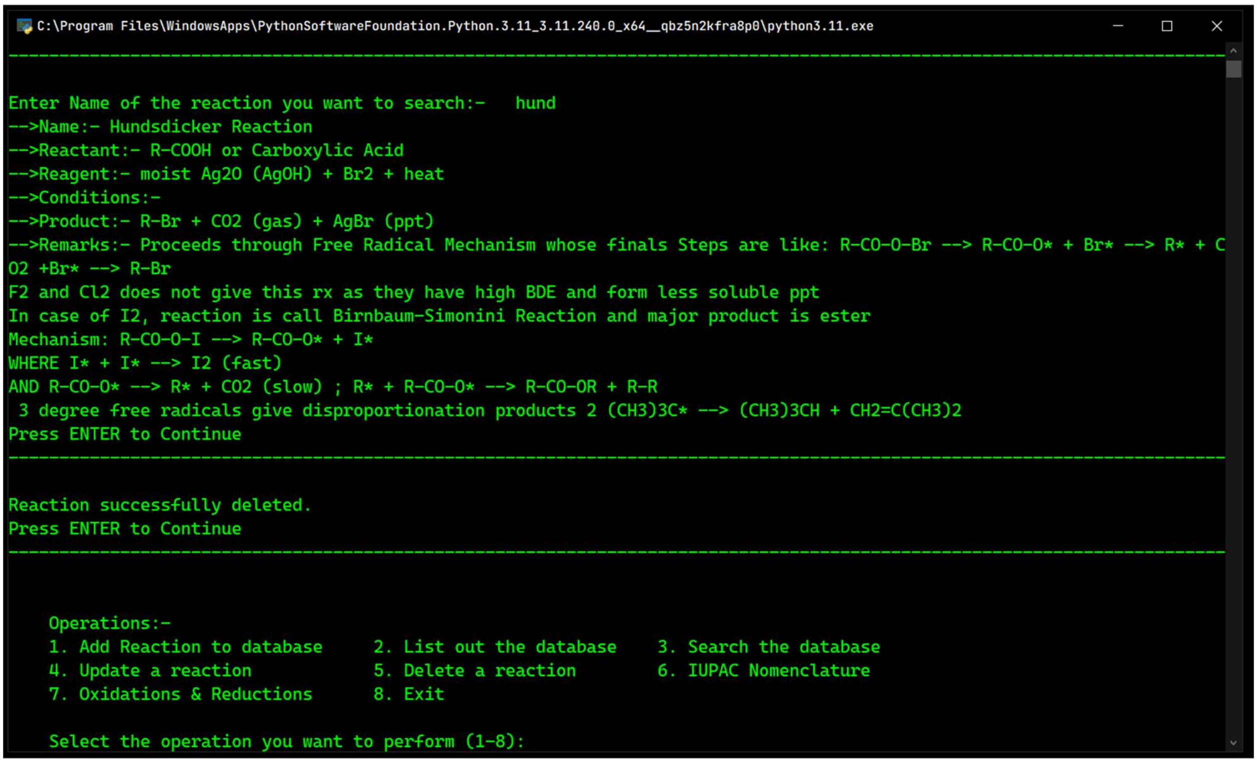

C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe

```
------------------------------------------------------------------------------------------------------------------------


<UPDATING A REACTION
First search for the reaction you want to update:
<SEARCHING THE DATABASE>
By which parameter would you like to search for reaction?
    1. Name         2. Reactant     3. Reagent              4. Conditions
    5. Product      (6. Remarks (searches same as 7.))      7. Search in all

Enter your choice (1-6):-  3
------------------------------------------------------------------------------------------------------------------------


Enter Reagent of the reaction you want to search:-   Halogen
-->Name:- Halogenation or Chlorination or Bromination
-->Reactant:- Alkanes
-->Reagent:- Halogens or Chlorine or Bromine
-->Conditions:- Sunlight or hv
-->Product:- Alkyl Halide
-->Remarks:- Reaction proceeds through Free Radical Mechanism.
Probability factors for 1, 2 and 3 degree carbons are 1:3.8:5 for Chlorination and 1:80:1600 for Bromination.
Step 1 is Initialisation: X2 -->(hv) X* + X* .
Step 2 is Propagation: R-H + X* --> R* + HX & R* + X2 --> R-X + X* .
Step 3 is Termination: R* + R* --> R-R & R* + X* --> R-X .
If Alkane is in excess, Monohalogenated products are formed.
If Halogen in excess, multisubstituted products will be formed.
Press ENTER to Continue
Enter the term you want to update(1-6):- 3
Enter new Reagent for the Reaction:- Halogens or X2 or Chlorine or Cl2 or Bromine or Br2
Reaction successfully updated.
Press ENTER to Continue
```

●Operation 5

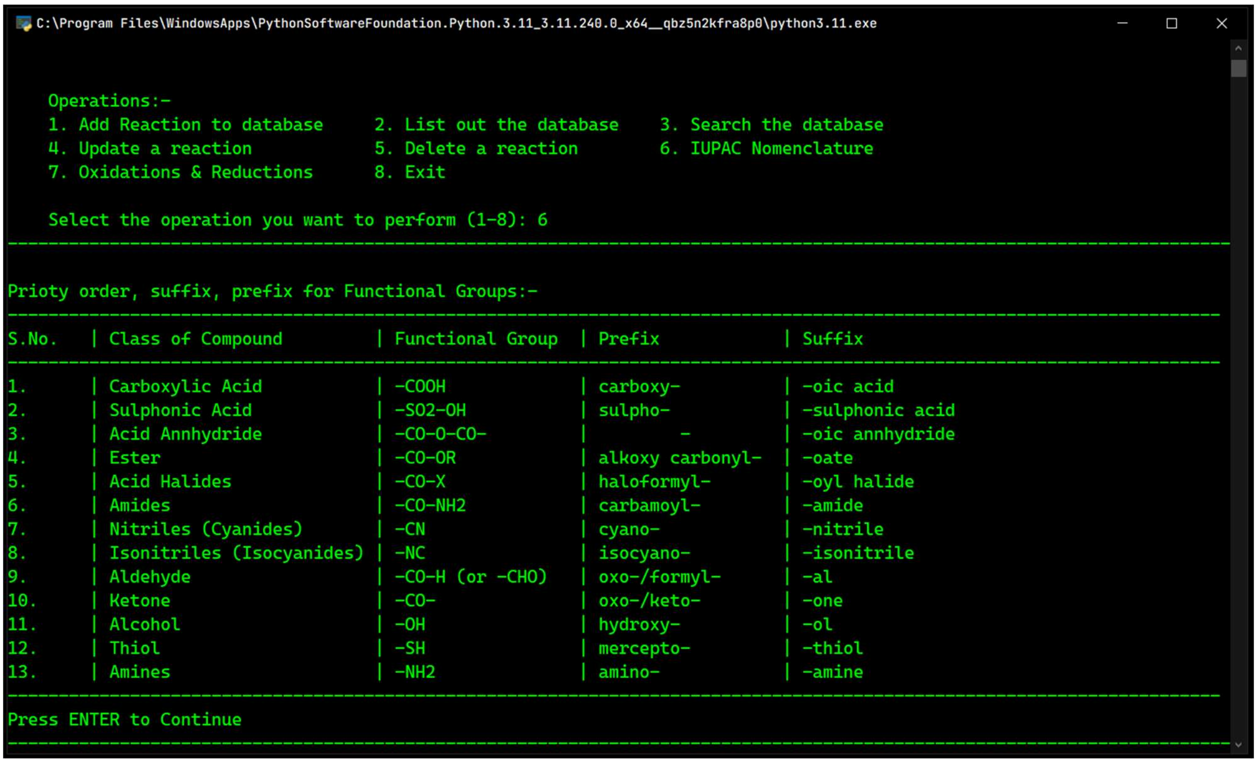

C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe

```
    Operations:-
    1. Add Reaction to database     2. List out the database      3. Search the database
    4. Update a reaction            5. Delete a reaction          6. IUPAC Nomenclature
    7. Oxidations & Reductions      8. Exit

    Select the operation you want to perform (1-8): 5
------------------------------------------------------------------------------------------------------------------------

<DELETING A REACTION>
First search for the reaction you want to delete:
<SEARCHING THE DATABASE>
By which parameter would you like to search for reaction?
    1. Name         2. Reactant     3. Reagent              4. Conditions
    5. Product      (6. Remarks (searches same as 7.))      7. Search in all

Enter your choice (1-6):-  1
------------------------------------------------------------------------------------------------------------------------

Enter Name of the reaction you want to search:-   hund
-->Name:- Hundsdicker Reaction
-->Reactant:- R-COOH or Carboxylic Acid
-->Reagent:- moist Ag2O (AgOH) + Br2 + heat
-->Conditions:-
-->Product:- R-Br + CO2 (gas) + AgBr (ppt)
-->Remarks:- Proceeds through Free Radical Mechanism whose finals Steps are like: R-CO-O-Br --> R-CO-O* + Br* --> R* + C
O2 +Br* --> R-Br
F2 and Cl2 does not give this rx as they have high BDE and form less soluble ppt
In case of I2, reaction is call Birnbaum-Simonini Reaction and major product is ester
```

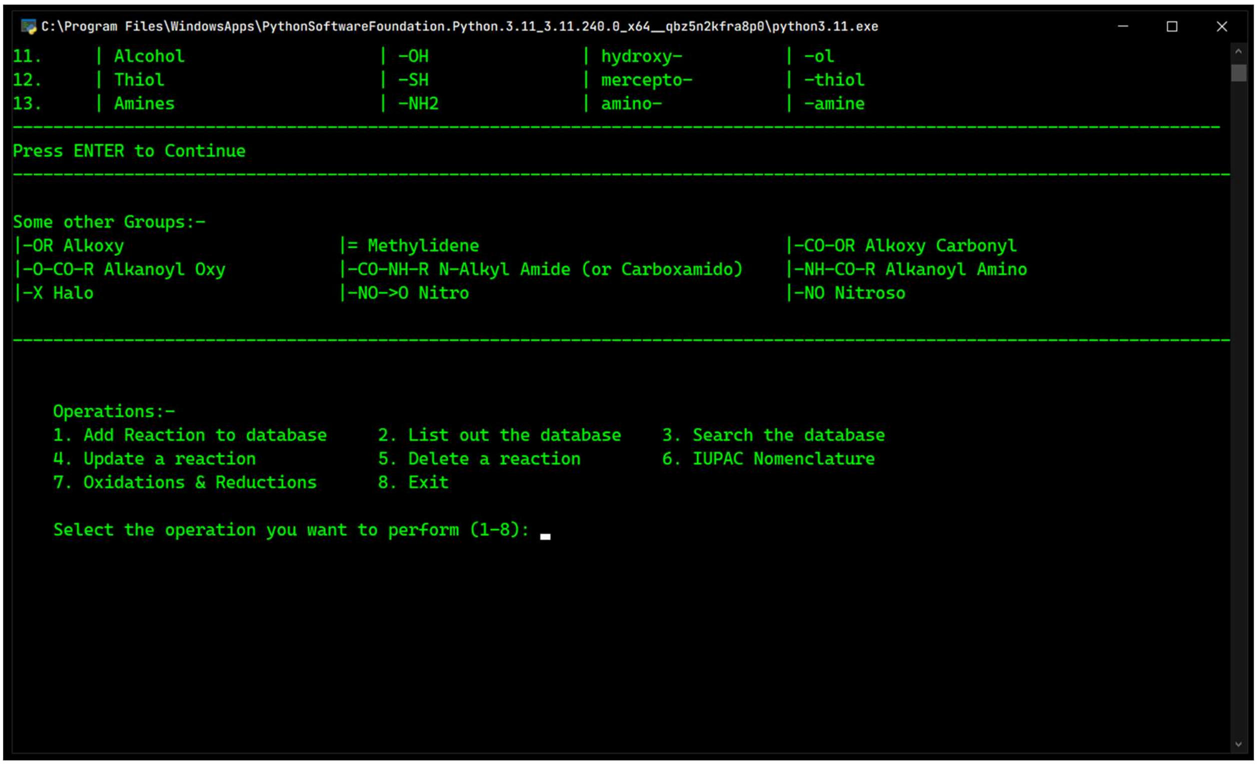

C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe

```
------------------------------------------------------------------------------------------------------------------------

Enter Name of the reaction you want to search:-   hund
-->Name:- Hundsdicker Reaction
-->Reactant:- R-COOH or Carboxylic Acid
-->Reagent:- moist Ag2O (AgOH) + Br2 + heat
-->Conditions:-
-->Product:- R-Br + CO2 (gas) + AgBr (ppt)
-->Remarks:- Proceeds through Free Radical Mechanism whose finals Steps are like: R-CO-O-Br --> R-CO-O* + Br* --> R* + C
O2 +Br* --> R-Br
F2 and Cl2 does not give this rx as they have high BDE and form less soluble ppt
In case of I2, reaction is call Birnbaum-Simonini Reaction and major product is ester
Mechanism: R-CO-O-I --> R-CO-O* + I*
WHERE I* + I* --> I2 (fast)
AND R-CO-O* --> R* + CO2 (slow) ; R* + R-CO-O* --> R-CO-OR + R-R
 3 degree free radicals give disproportionation products 2 (CH3)3C* --> (CH3)3CH + CH2=C(CH3)2
Press ENTER to Continue
------------------------------------------------------------------------------------------------------------------------

Reaction successfully deleted.
Press ENTER to Continue
------------------------------------------------------------------------------------------------------------------------


    Operations:-
    1. Add Reaction to database     2. List out the database      3. Search the database
    4. Update a reaction            5. Delete a reaction          6. IUPAC Nomenclature
    7. Oxidations & Reductions      8. Exit

    Select the operation you want to perform (1-8):
```

●Operation 6

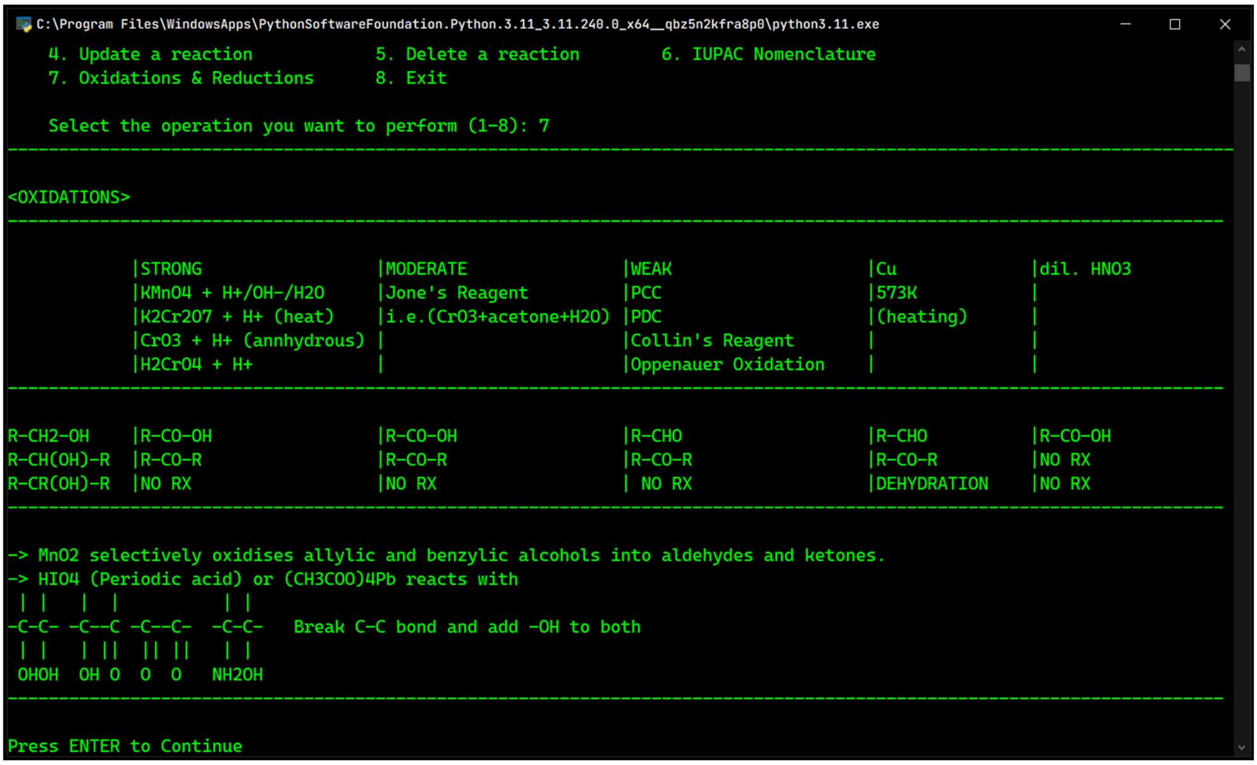


```
C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe


    Operations:-
    1. Add Reaction to database     2. List out the database    3. Search the database
    4. Update a reaction            5. Delete a reaction        6. IUPAC Nomenclature
    7. Oxidations & Reductions      8. Exit

    Select the operation you want to perform (1-8): 6
------------------------------------------------------------------------------------------------------------------------

Prioty order, suffix, prefix for Functional Groups:-
-----------------------------------------------------------------------------------------------------------------------
S.No.   | Class of Compound            | Functional Group  | Prefix              | Suffix
-----------------------------------------------------------------------------------------------------------------------
1.      | Carboxylic Acid              | -COOH             | carboxy-            | -oic acid
2.      | Sulphonic Acid               | -SO2-OH           | sulpho-             | -sulphonic acid
3.      | Acid Annhydride              | -CO-O-CO-         |         -           | -oic annhydride
4.      | Ester                        | -CO-OR            | alkoxy carbonyl-    | -oate
5.      | Acid Halides                 | -CO-X             | haloformyl-         | -oyl halide
6.      | Amides                       | -CO-NH2           | carbamoyl-          | -amide
7.      | Nitriles (Cyanides)          | -CN               | cyano-              | -nitrile
8.      | Isonitriles (Isocyanides)    | -NC               | isocyano-           | -isonitrile
9.      | Aldehyde                     | -CO-H (or -CHO)   | oxo-/formyl-        | -al
10.     | Ketone                       | -CO-              | oxo-/keto-          | -one
11.     | Alcohol                      | -OH               | hydroxy-            | -ol
12.     | Thiol                        | -SH               | mercepto-           | -thiol
13.     | Amines                       | -NH2              | amino-              | -amine
-----------------------------------------------------------------------------------------------------------------------
Press ENTER to Continue
------------------------------------------------------------------------------------------------------------------------
```

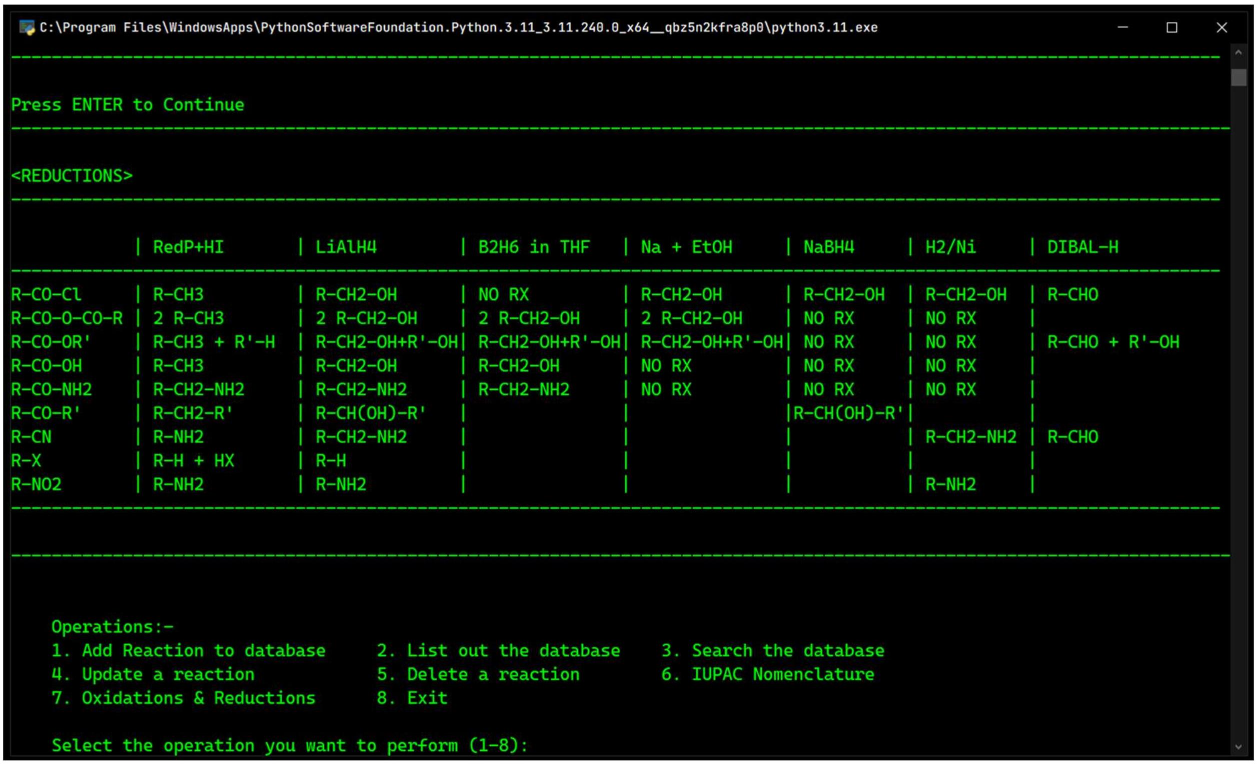


```
C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe
11.     | Alcohol                      | -OH               | hydroxy-            | -ol
12.     | Thiol                        | -SH               | mercepto-           | -thiol
13.     | Amines                       | -NH2              | amino-              | -amine
-----------------------------------------------------------------------------------------------------------------------
Press ENTER to Continue
------------------------------------------------------------------------------------------------------------------------

Some other Groups:-
|-OR Alkoxy                     |= Methylidene                          |-CO-OR Alkoxy Carbonyl
|-O-CO-R Alkanoyl Oxy           |-CO-NH-R N-Alkyl Amide (or Carboxamido) |-NH-CO-R Alkanoyl Amino
|-X Halo                        |-NO->O Nitro                           |-NO Nitroso

------------------------------------------------------------------------------------------------------------------------


    Operations:-
    1. Add Reaction to database     2. List out the database    3. Search the database
    4. Update a reaction            5. Delete a reaction        6. IUPAC Nomenclature
    7. Oxidations & Reductions      8. Exit

    Select the operation you want to perform (1-8): _
```

●Operation 7



```
C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe
    4. Update a reaction            5. Delete a reaction        6. IUPAC Nomenclature
    7. Oxidations & Reductions      8. Exit

    Select the operation you want to perform (1-8): 7
------------------------------------------------------------------------------------------------------------------------

<OXIDATIONS>
-----------------------------------------------------------------------------------------------------------------------

            |STRONG                 |MODERATE               |WEAK                   |Cu              |dil. HNO3
            |KMnO4 + H+/OH-/H2O     |Jone's Reagent         |PCC                    |573K            |
            |K2Cr2O7 + H+ (heat)    |i.e.(CrO3+acetone+H2O) |PDC                    |(heating)       |
            |CrO3 + H+ (annhydrous) |                       |Collin's Reagent       |                |
            |H2CrO4 + H+            |                       |Oppenauer Oxidation    |                |
-----------------------------------------------------------------------------------------------------------------------

R-CH2-OH    |R-CO-OH                |R-CO-OH                |R-CHO                  |R-CHO           |R-CO-OH
R-CH(OH)-R  |R-CO-R                 |R-CO-R                 |R-CO-R                 |R-CO-R          |NO RX
R-CR(OH)-R  |NO RX                  |NO RX                  | NO RX                 |DEHYDRATION     |NO RX
-----------------------------------------------------------------------------------------------------------------------

-> MnO2 selectively oxidises allylic and benzylic alcohols into aldehydes and ketones.
-> HIO4 (Periodic acid) or (CH3COO)4Pb reacts with
 | |   |  |          | |
-C-C- -C--C -C--C-  -C-C-   Break C-C bond and add -OH to both
 | |   | ||  || ||   | |
 OHOH  OH O  O  O   NH2OH
-----------------------------------------------------------------------------------------------------------------------

Press ENTER to Continue
```



```
C:\Program Files\WindowsApps\PythonSoftwareFoundation.Python.3.11_3.11.240.0_x64__qbz5n2kfra8p0\python3.11.exe
-----------------------------------------------------------------------------------------------------------------------

Press ENTER to Continue
------------------------------------------------------------------------------------------------------------------------

<REDUCTIONS>
-----------------------------------------------------------------------------------------------------------------------

            | RedP+HI       | LiAlH4        | B2H6 in THF   | Na + EtOH     | NaBH4     | H2/Ni     | DIBAL-H
-----------------------------------------------------------------------------------------------------------------------
R-CO-Cl     | R-CH3         | R-CH2-OH      | NO RX         | R-CH2-OH      | R-CH2-OH  | R-CH2-OH  | R-CHO
R-CO-O-CO-R | 2 R-CH3       | 2 R-CH2-OH    | 2 R-CH2-OH    | 2 R-CH2-OH    | NO RX     | NO RX     |
R-CO-OR'    | R-CH3 + R'-H  | R-CH2-OH+R'-OH| R-CH2-OH+R'-OH| R-CH2-OH+R'-OH| NO RX     | NO RX     | R-CHO + R'-OH
R-CO-OH     | R-CH3         | R-CH2-OH      | R-CH2-OH      | NO RX         | NO RX     | NO RX     |
R-CO-NH2    | R-CH2-NH2     | R-CH2-NH2     | R-CH2-NH2     | NO RX         | NO RX     | NO RX     |
R-CO-R'     | R-CH2-R'      | R-CH(OH)-R'   |               |               |R-CH(OH)-R'|           |
R-CN        | R-NH2         | R-CH2-NH2     |               |               |           | R-CH2-NH2 | R-CHO
R-X         | R-H + HX      | R-H           |               |               |           |           |
R-NO2       | R-NH2         | R-NH2         |               |               |           | R-NH2     |
-----------------------------------------------------------------------------------------------------------------------

------------------------------------------------------------------------------------------------------------------------


    Operations:-
    1. Add Reaction to database     2. List out the database    3. Search the database
    4. Update a reaction            5. Delete a reaction        6. IUPAC Nomenclature
    7. Oxidations & Reductions      8. Exit

    Select the operation you want to perform (1-8):
```

BIBLIOGRAPHY

1. Computer Science with python class 12 Sumita Arora
2. https://stackoverflow.com/
3. https://www.geeksforgeeks.org/
4. https://dev.mysql.com/doc/refman/8.0/en/