

# OUTLINE



## 1 Recapitulando

Drive - Colab - Github

## 2 Open Babel

Casos de aplicación

## 3 Inputs para cálculo cuántico

Generación de coordenadas cartesianas y manipulación de diedros

## 4 Estereoisomería

@ y @@ en la representación SMILES

## 5 Descriptores y Bases de datos

CHEMBL, PeruNPDB, tmQM, PunoNPDB

# RECAPITULANDO

## Drive - Colab - Github

- » Estamos empleando estas 3 plataformas continuamente ¿Ya estás familiarizado?
- » ¿Ya ejecuté los colabs de la primera clase? ¿los guardé en github o en drive?
- » ¿Sí en colab optimizo 100 moléculas, estas se guardan automáticamente en drive?
- » Si editamos un colab en grupo ¿Qué cuidados debemos tener para guardar?



# OUTLINE



## 1 Recapitulando

Drive - Colab - Github

## 2 Open Babel

Casos de aplicación

## 3 Inputs para cálculo cuántico

Generación de coordenadas cartesianas y manipulación de diedros

## 4 Estereoisomería

@ y @@ en la representación SMILES

## 5 Descriptores y Bases de datos

CHEMBL, PeruNPDB, tmQM, PunoNPDB

# OPEN BABEL

¿Es solamente para convertir entre formatos?

- » Sintaxis y caso de aplicación N°1 en el siguiente **ejemplo**
- » Asegúrate de guardar tus moléculas en PDBQT

## Atento Experimentalista

*Reposicionamiento de fármacos*

Dispón de un grupo de fármacos en SMILES u otro formato.

Genera sus estructuras 3D y pre-optimízalas.

Prepara el target y a dockear!

# OUTLINE



## 1 Recapitulando

Drive - Colab - Github

## 2 Open Babel

Casos de aplicación

## 3 Inputs para cálculo cuántico

Generación de coordenadas cartesianas y manipulación de diedros

## 4 Estereoisomería

@ y @@ en la representación SMILES

## 5 Descriptores y Bases de datos

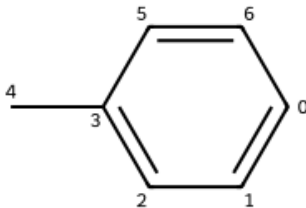
CHEMBL, PeruNPDB, tmQM, PunoNPDB

# VISUALIZACIÓN DE ESTRUCTURAS EN 2D

En el segundo **notebook** vamos a generar estructuras 3D a partir de SMILES, trabajar con confórmeros y emplear

## Importando librerías

```
# Tolueno  
smiles = "c1ccc(C)cc1"  
# Convierte a elemento de RDKit  
mol = Chem.MolFromSmiles(smiles)  
# Dibuja la estructura  
Draw.MolsToGridImage([mol])
```

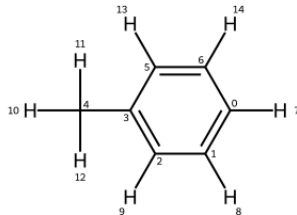


# HIDRÓGENOS EXPLÍCITOS

La estructura anterior tiene hidrógenos implícitos, pero para generar la estructura 3D requerimos todos los átomos de forma explícita.

## Importando librerías

```
# Agrega hidrógenos
mol_h = Chem.AddHs(mol)
# Imprime la molécula
Draw.MolsToGridImage([mol_h])
```



# ARCHIVO DE ENTRADA (INPUT)

El input para el cálculo con química cuántica es una estructura 3D. Agregar coordenadas (x,y,z) a cada átomo genera un confórmero.

Molécula en 2D o con Z=0

```
Chem.MolToMolBlock(mol_h)
```

RDKit								2D			
15	15	0	0	0	0	0	0	0	0	0999	V2000
	1.5000		0.0000		0.0000		0.0000	C	0	0	
	0.7500		-1.2990		0.0000		0.0000	C	0	0	
	-0.7500		-1.2990		0.0000		0.0000	C	0	0	
	-1.5000		0.0000		0.0000		0.0000	C	0	0	
	-3.0000		0.0000		0.0000		0.0000	C	0	0	
	-0.7500		1.2990		0.0000		0.0000	C	0	0	
	0.7500		1.2990		0.0000		0.0000	C	0	0	
	3.0000		0.0000		0.0000		0.0000	H	0	0	
	1.5000		-2.5981		0.0000		0.0000	H	0	0	
	-1.5000		-2.5981		0.0000		0.0000	H	0	0	
	-4.5000		0.0000		0.0000		0.0000	H	0	0	
	-3.0000		1.5000		0.0000		0.0000	H	0	0	
	-3.0000		-1.5000		0.0000		0.0000	H	0	0	
	-1.5000		2.5981		0.0000		0.0000	H	0	0	
	1.5000		2.5981		0.0000		0.0000	H	0	0	
1	2	2	0								
2	3	1	0								
3	4	2	0								
4	5	1	0								



```
# La estructura 3D es un confórmero aleatorio
Chem.AllChem.EmbedMolecule(mol_h)
# Revisando las coordenadas:
Chem.MolToMolBlock(mol_h)
# Guardamos el input
Chem.MolToXYZFile(mol_h,"toluene.xyz")
```

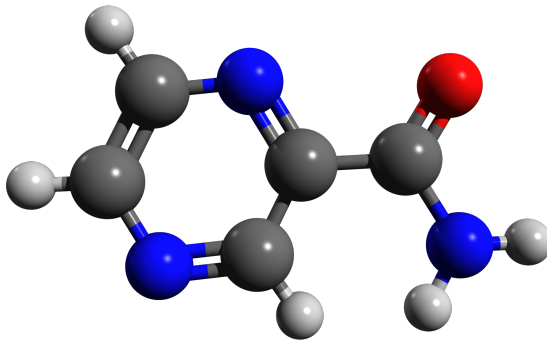
Información complementaria:

- » Documentación de RDKit
- » Generación de confórmers con RDKit
- » Mejorando la generación de confórmers.

RDKit				3D						
15	15	0	0	0	0	0	0	0	0999	V200
	2.0557			0.0019			-0.0276	C		0
	1.3603			-1.1811			-0.1707	C		0
	-0.0182			-1.1834			-0.1460	C		0
	-0.7198			0.0040			0.0229	C		0
	-2.2091			-0.0135			0.0476	C		0
	0.0197			1.1516			0.1606	C		0
	1.3854			1.2098			0.1435	C		0
	3.1375			0.0062			-0.0466	H		0
	1.9399			-2.0834			-0.3000	H		0
	-0.5784			-2.1006			-0.2566	H		0
	-2.5863			-0.8443			-0.6125	H		0
	-2.6092			0.9214			-0.4374	H		0
	-2.6121			-0.0904			1.0771	H		0
	-0.5342			2.0876			0.2937	H		0
	1.9690			2.1143			0.2519	H		0
1	2	2	0							
2	3	1	0							
3	4	2	0							
4	5	1	0							

# EJERCICIO 1: MI PRIMERA AUTOMATIZACIÓN

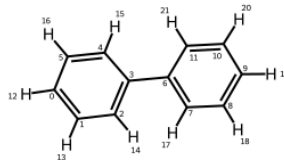
- » Crea la representación SMILES para la pirazinamida y guarda las coordenadas en PZA.xyz



## EJERCICIO 2: MANIPULANDO DIEDROS

### Creamos la estructura 3D

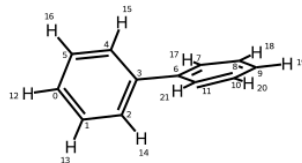
```
smiles = "C1=CC=C(C=C1)C2=CC=CC=C2"  
mol = Chem.MolFromSmiles(smiles)  
mol = Chem.AddHs(mol)  
Chem.AllChem.EmbedMolecule(mol)  
Chem.MolToXYZFile(mol, "initial.xyz")
```



## EJERCICIO 2: MANIPULANDO DIEDROS

### Modificamos el diedro

```
# Configuramos 90° para el diedro
Chem.AllChem.SetDihedralDeg
(mol.GetConformer(0),4,3,6,11,90.0)
# Guardamos el nuevo conformero
Chem.MolToXYZFile(mol, "90.xyz")
# Imprimimos
```



## EJERCICIO 2: MANIPULANDO DIEDROS

### Obtenemos confórmeros con diferentes diedros

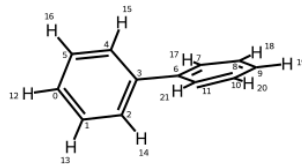
```
diedros = [0, 30, 60, 90, 120, 150,  
180, 210, 240, 270, 300, 330]
```

```
for i in diedros:
```

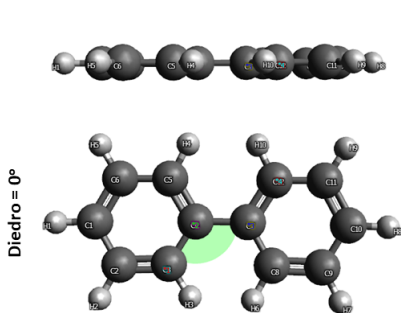
```
    Chem.AllChem.SetDihedralDeg
```

```
    (mol.GetConformer(0),4,3,6,11,i)
```

```
    Chem.MolToXYZFile(mol, f"{i}.xyz")
```

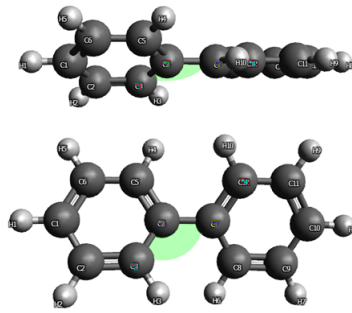


# EJERCICIO 2: MANIPULANDO DIEDROS



Dihedral Angle  
Distance(Å)

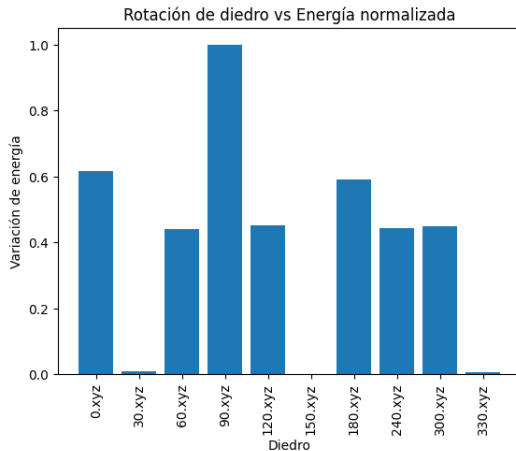
-181°P  
181°P  
1.387 Å 1.451 Å 1.574 Å



Dihedral Angle  
Distance(Å)

-181°P  
181°P  
1.387 Å 1.451 Å 1.574 Å

# DIEDRO VS ENERGÍA SCF



# OUTLINE



## 1 Recapitulando

Drive - Colab - Github

## 2 Open Babel

Casos de aplicación

## 3 Inputs para cálculo cuántico

Generación de coordenadas cartesianas y manipulación de diedros

## 4 Estereoisomería

@ y @@ en la representación SMILES

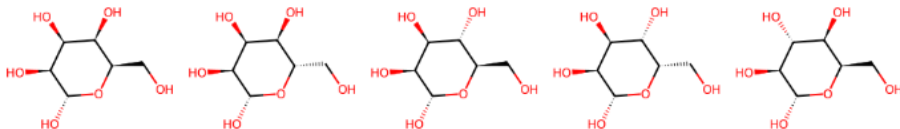
## 5 Descriptores y Bases de datos

CHEMBL, PeruNPDB, tmQM, PunoNPDB



# EJERCICIO 3

» ¿Cómo generamos estereoisómeros? veamos un caso de aplicación con la glucosa



# OUTLINE



## 1 Recapitulando

Drive - Colab - Github

## 2 Open Babel

Casos de aplicación

## 3 Inputs para cálculo cuántico

Generación de coordenadas cartesianas y manipulación de diedros

## 4 Estereoisomería

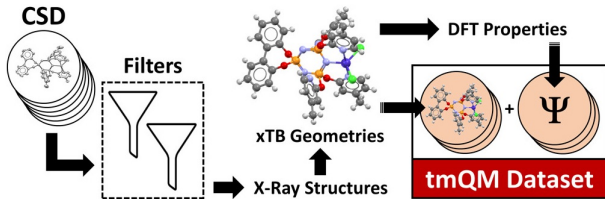
@ y @@ en la representación SMILES

## 5 Descriptores y Bases de datos

CHEMBL, PeruNPDB, tmQM, PunoNPDB

# EXPLOREMOS ALGUNAS BASES DE DATOS

- » "CHEMBL: ChEMBL: a large-scale bioactivity database for drug discovery". [Colab](#) de consulta.
- » "PeruNPDB: the Peruvian Natural Products Database for in silico drug screening". Obtención de la data mediante [web scrapping](#).
- » "tmQM Dataset — Quantum Geometries and Properties of 86k Transition Metal Complexes". [Colab](#) para el análisis exploratorio de los datos (EDA)
- » "PunoNPDB: base de datos de productos naturales en Puno-Perú" (Guadalupe Enriquez-Anco)





CURSO DE

**QUÍMICA**

**PARA COMPUTACIONAL**

**PARA EXPERIMENTALISTAS**

PARTE I