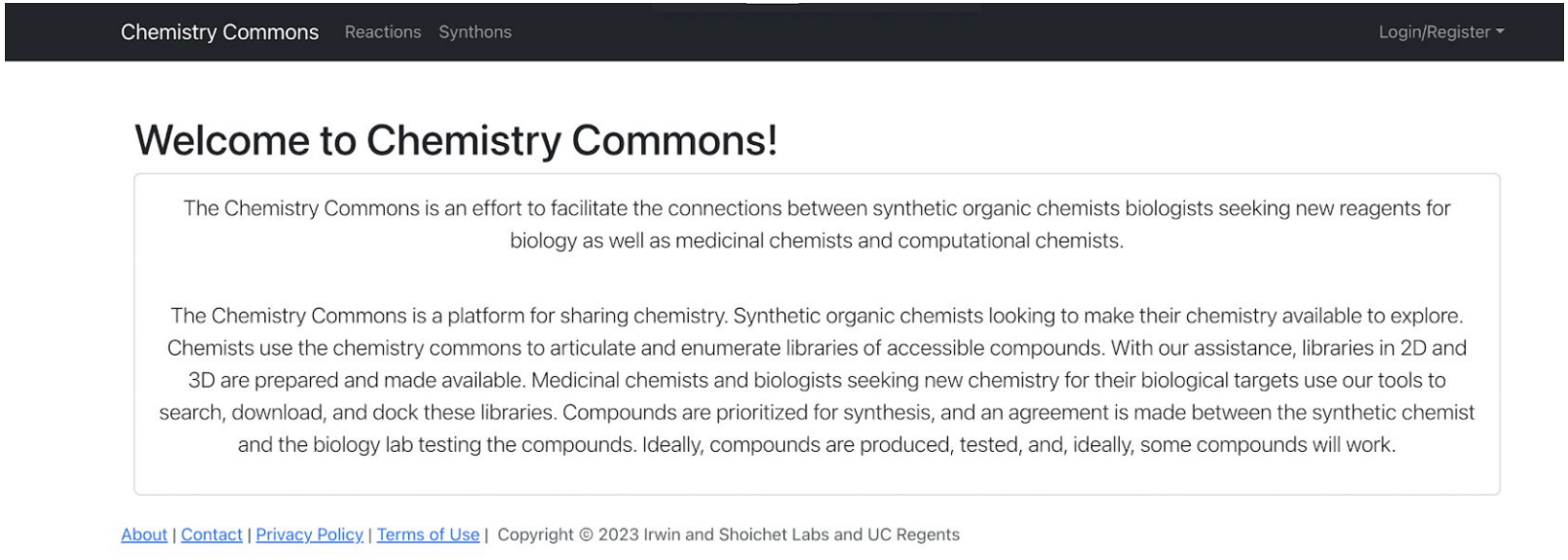
Adding Reactions into the Chemistry Commons, Retrosynthetic Analoging, and Reaction Enumeration

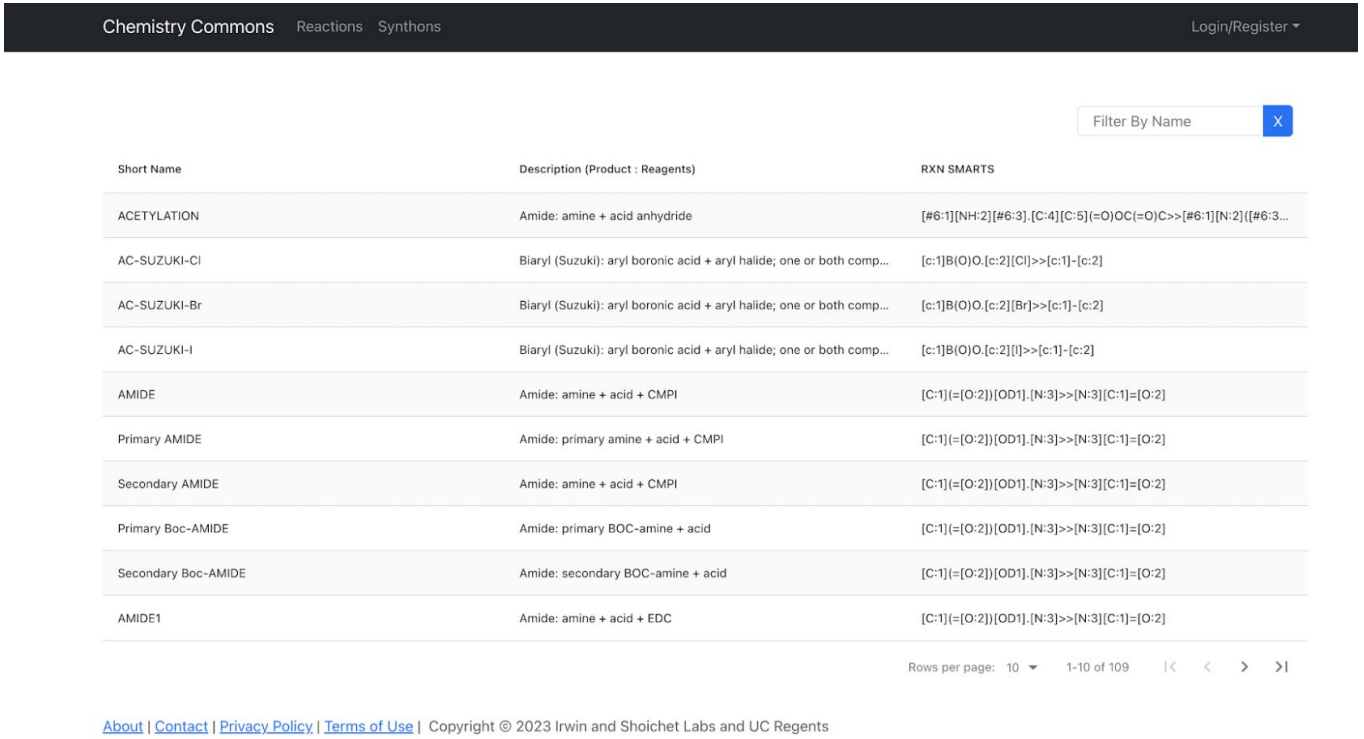
UGI
By Maksim DuNova

Chemistry Commons

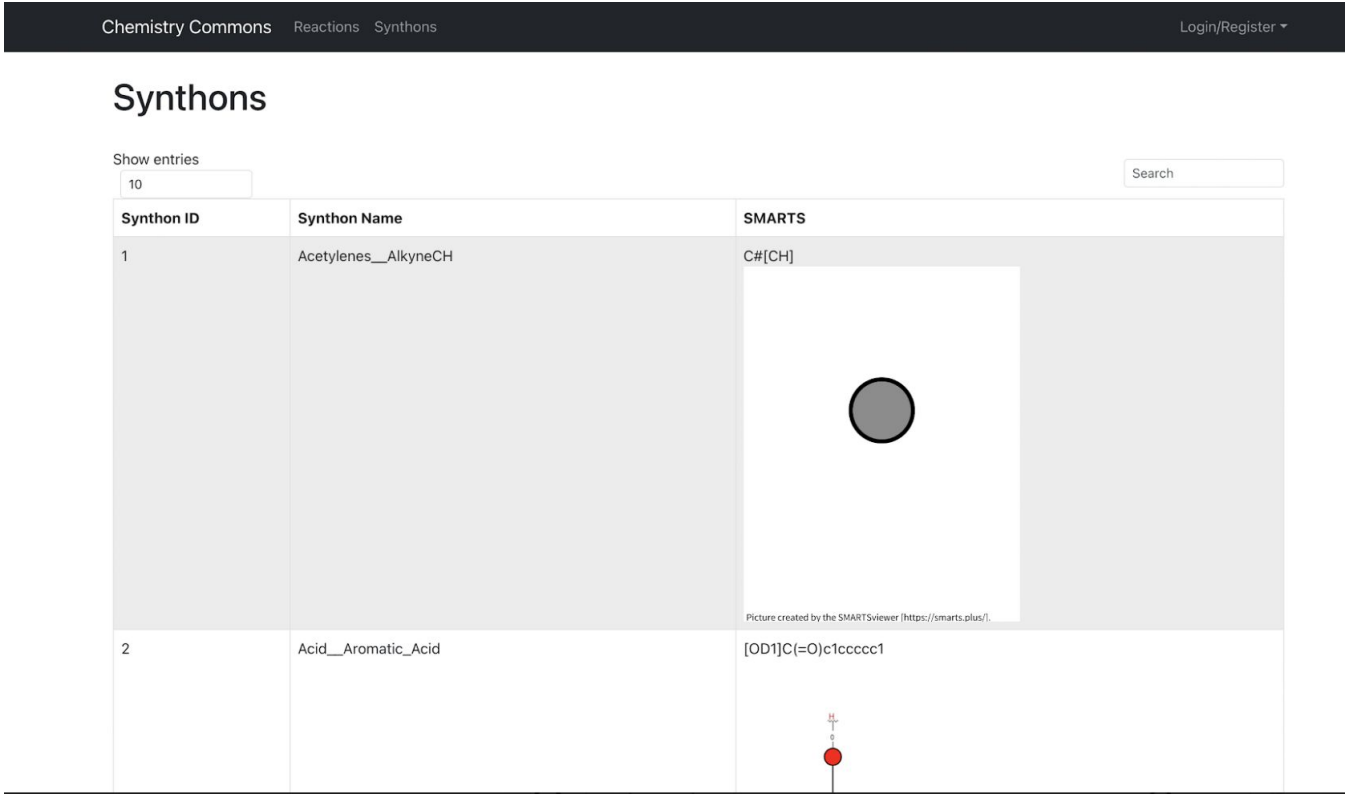
The goal of Chemistry Commons is to make an interactive space for chemists to facilitate collaboration on reaction curation and to share their own reactions for the purpose of exploring chemical space!



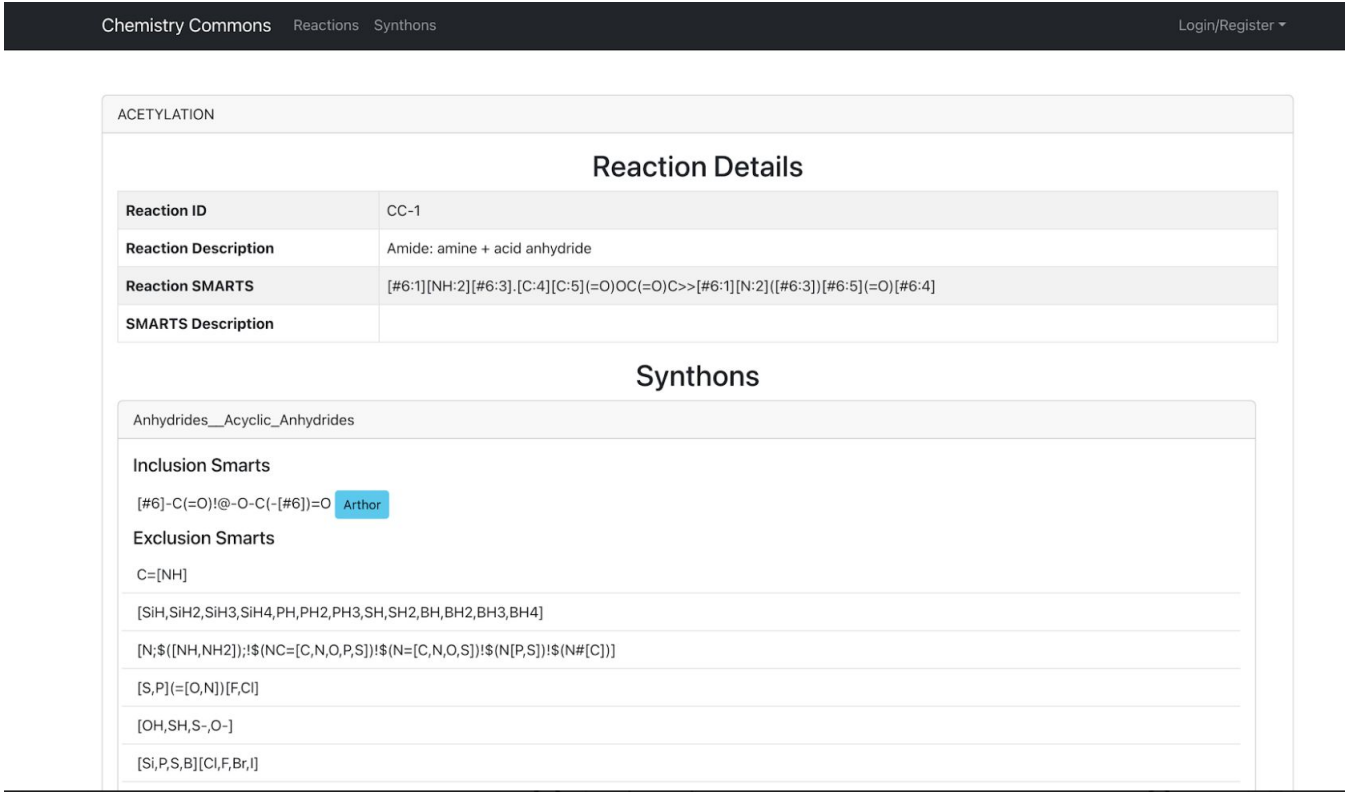
Reaction Database



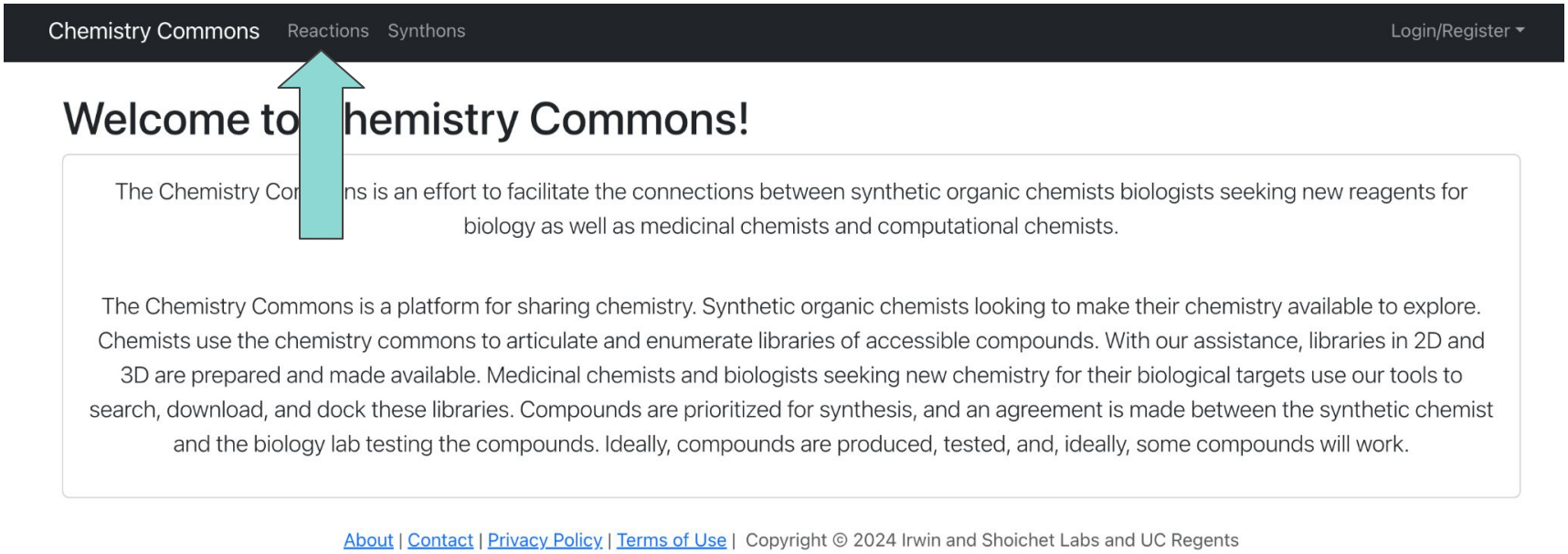
Synthon Database



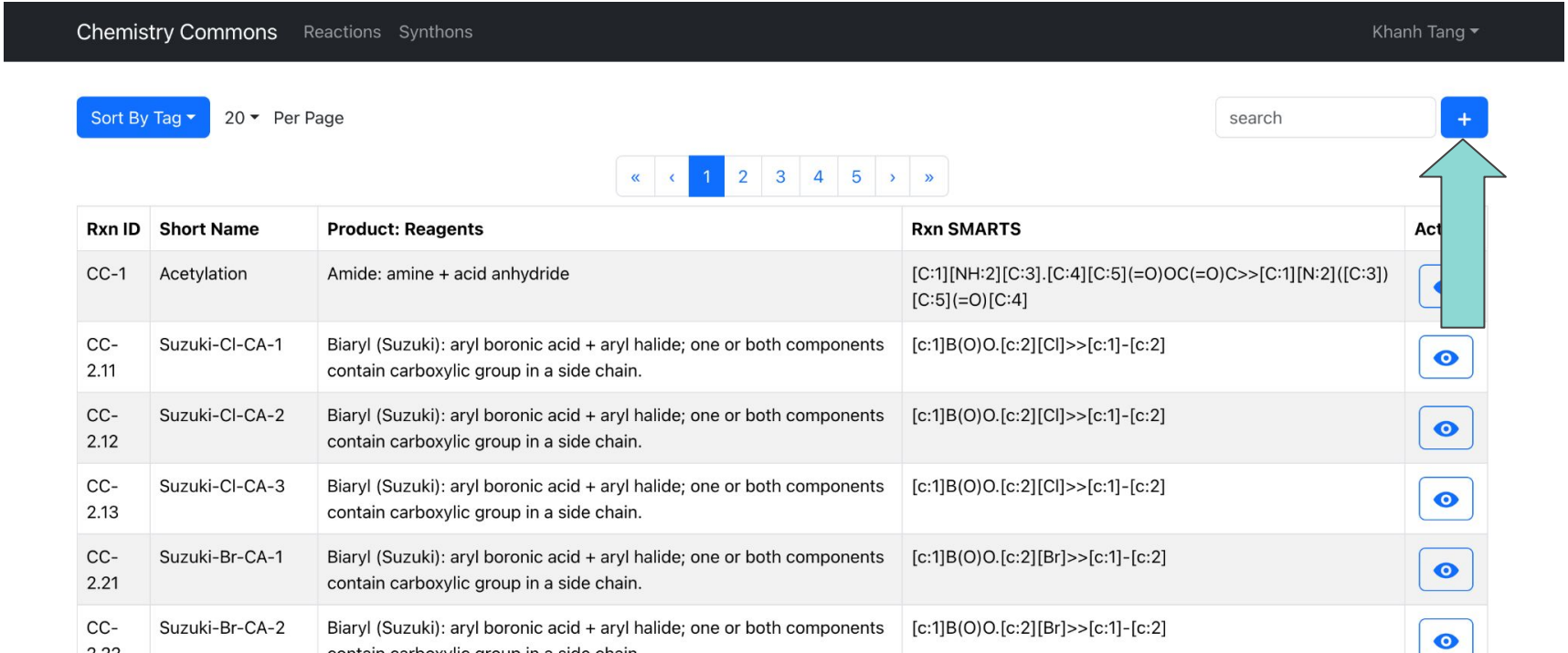
Reaction Details



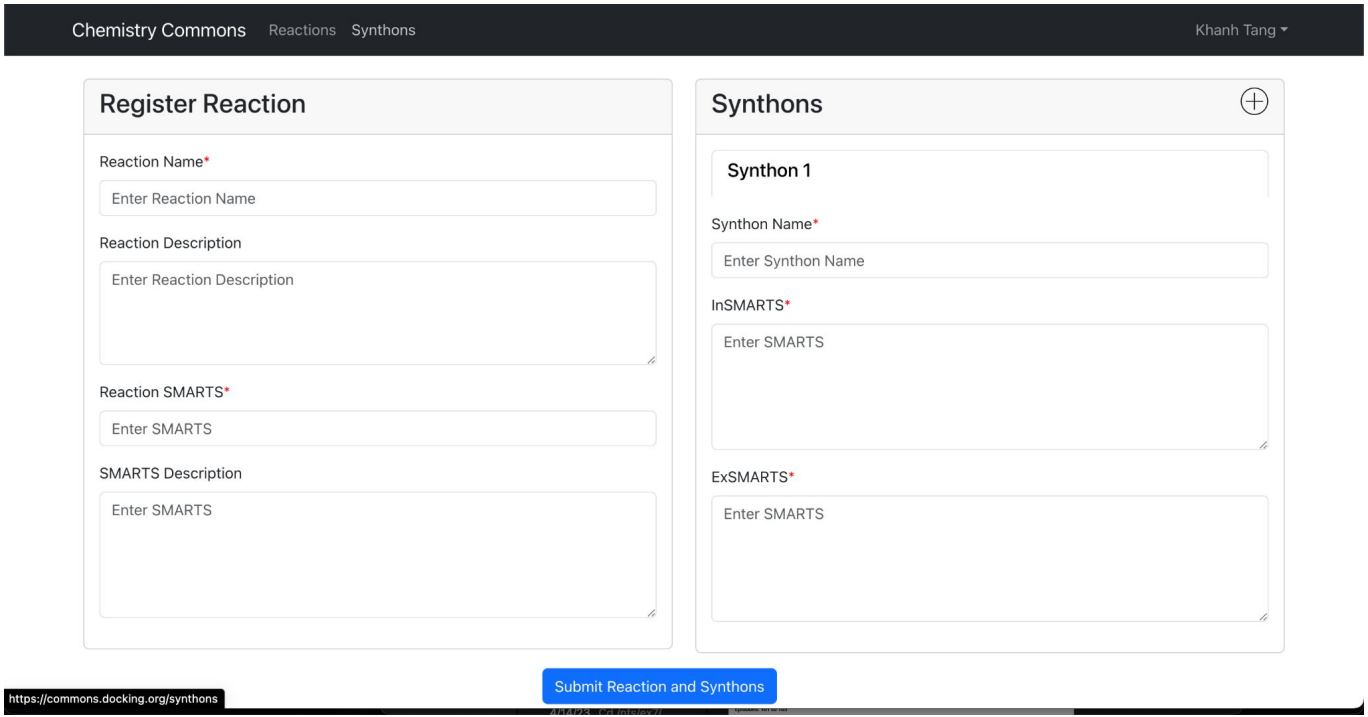
How to add a Reaction into the Chemistry Commons.



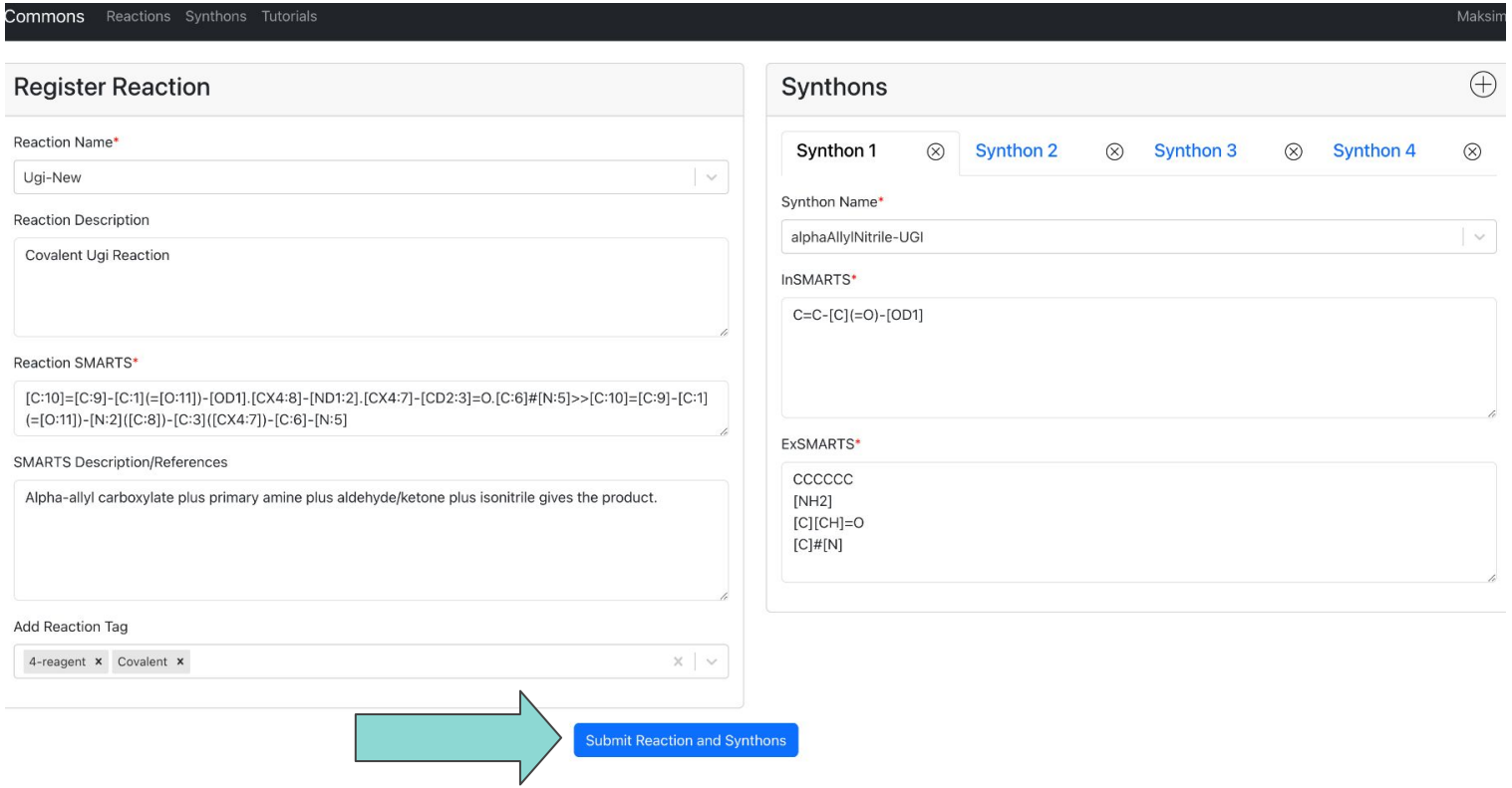
How to add a Reaction into the Chemistry Commons.



Enter in Reaction Details and Submit!



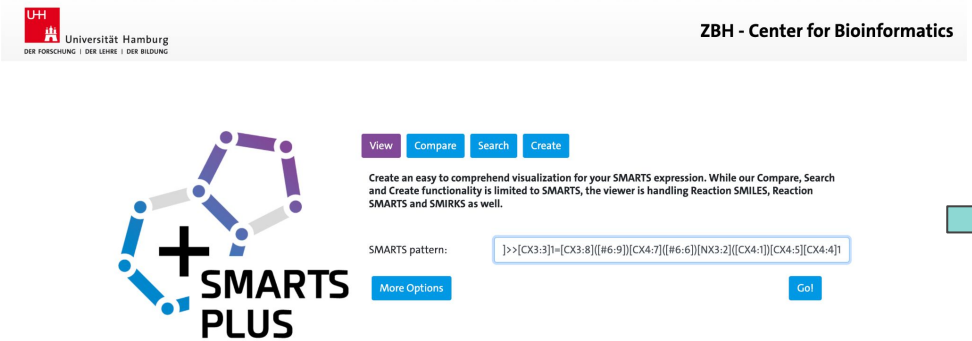
Enter in Reaction Details and Submit!

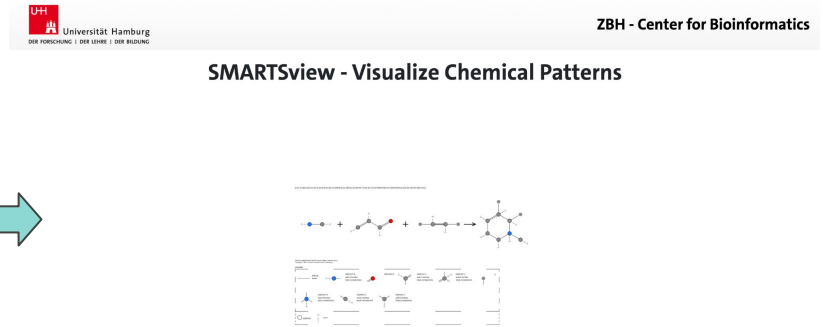


First Step is to learn SMARTS Resources

- Smarts Plus
- Daylight SMARTS

These are useful resources in helping learn the syntax of SMARTS, visualize SMARTS, as well as seeing if the SMARTS patterns are valid or not

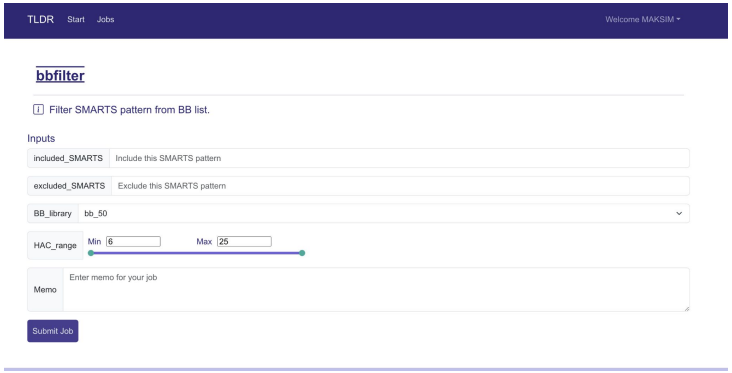


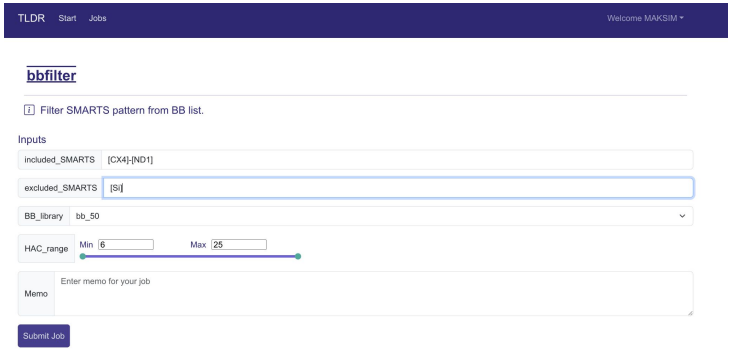


First Step is to learn SMARTS Resources

- [BBFilter module on tldr.docking.org]

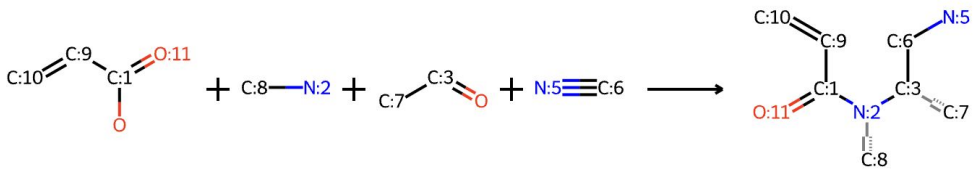
  This is a module on from our own tool that helps verify your inclusion and exclusion SMARTS





Transforming the Reaction into SMARTS

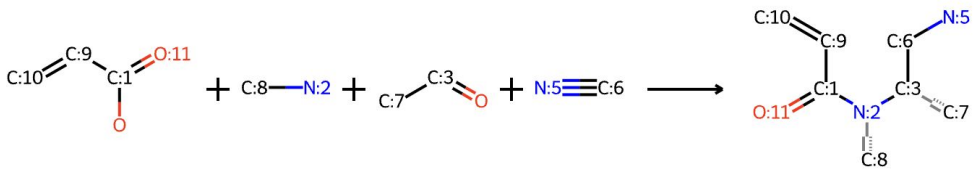
- Figure out the reaction that you want to transform into SMARTS!
- In this example we are transforming, UGI
- UGI is a 4-Component Reaction

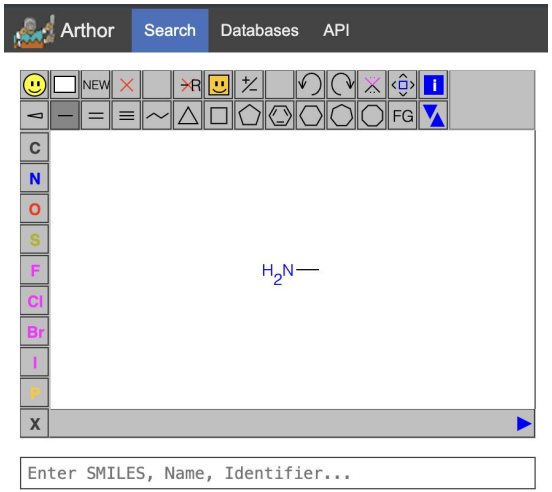


Allyl Acid | Primary Amine | Aldehyde | Nitrile | Product

Transforming the Reaction into SMARTS: Finding the SMILES of each Reactant

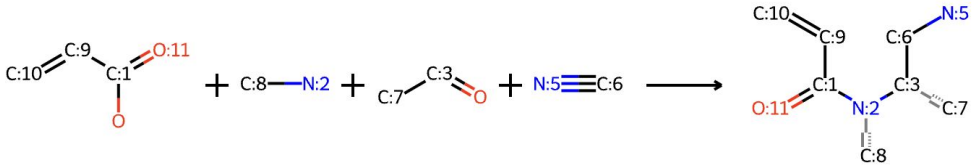
- First step is to figure out the SMILES of the reactant(s) and product
- You can do this simply by drawing the molecule in our tool Arthor and copying the SMILES output by the drawing tool





Transforming the Reaction into SMARTS: R Group SMARTS

- After you figure out the SMILES of each component, now specify the R-group bond order/connections using SMARTS notation if this is required for your reaction
- Example: C-N -> [CX4]-[NH2]



Allyl Acid | Primary Amine | Aldehyde | Nitrile | Product

Transforming the Reaction into SMARTS: Daylight Resource

- Again Daylight SMARTS is a good resource if you need to learn SMARTS notation

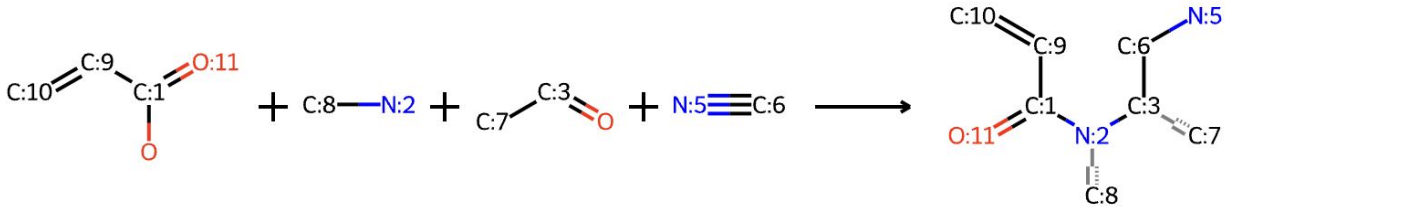
SMARTS Atomic Primitives

Symbol	Symbol name	Atomic property requirements	Default
*	wildcard	any atom	(no default)
a	aromatic	aromatic	(no default)
A	aliphatic	aliphatic	(no default)
D<n>	degree	<n> explicit connections	exactly one
H<n>	total-H-count	<n> attached hydrogens	exactly one[1]
h<n>	implicit-H-count	<n> implicit hydrogens	at least one
R<n>	ring membership	in <n> SSSR rings	any ring atom
r<n>	ring size	in smallest SSSR ring of size <n>	any ring atom[2]
v<n>	valence	total bond order <n>	exactly one[2]
X<n>	connectivity	<n> total connections	exactly one[2]
x<n>	ring connectivity	<n> total ring connections	at least one[2]
- <n>	negative charge	-<n> charge	-1 charge (-- is -2, etc)
+<n>	positive charge	+<n> formal charge	+1 charge (++ is +2, etc)
#n	atomic number	atomic number <n>	(no default)[2]
@	chirality	anticlockwise	anticlockwise, default class[2]
@@	chirality	clockwise	clockwise, default class[2]
@<c><n>	chirality	chiral class <c> chirality <n>	(nodefault)
@<c><n>?	chiral or unspec	chirality <c><n> or unspecified	(no default)
<n>	atomic mass	explicit atomic mass	unspecified mass

Transforming the Reaction into SMARTS: Inclusion SMARTS

- At the step you should have the some simplified SMARTS of each component

1. Allyl Acid **2. Primary Amine** **3. Aldehyde** **4. Nitrile**



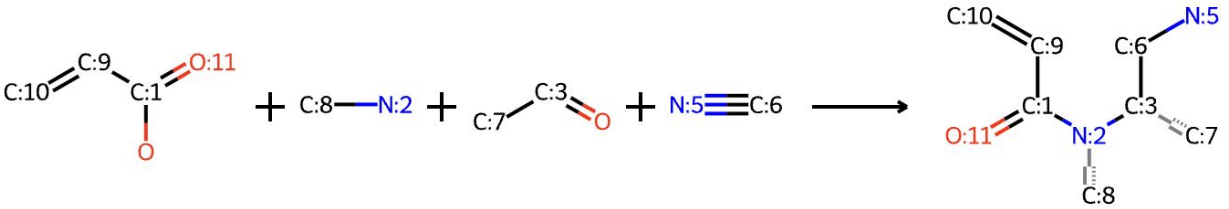
[C]=[C]-[C](=[O])-[O] [CX4]-[N] [CX4]-[C]=O [C]#[N]

[C]=[C]-[C](=[O])-[N]([C])-[C]([C])-[C]-[N]

Transforming the Reaction into SMARTS: Bond SMARTS

- Save the SMARTS from the previous step as these will be our inclusion SMARTS
- Next step is to specify bonds!

1. Allyl Acid **2. Primary Amine** **3. Aldehyde** **4. Nitrile**



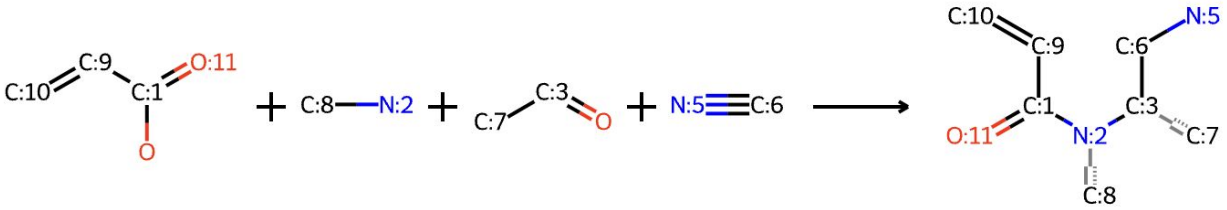
[C]=[C]-[C](=[O])-[OD1] [CX4]-[ND1] [CX4]-[CD2]=O [C]#[N]

[C]=[C]-[C](=[O])-[N]([C])-[C]([CX4])-[C]-[N]

Transforming the Reaction into SMARTS: Mapping Atoms

- Next is to **Map** the atoms in the reactants into the product (long step)

1. Allyl Acid **2. Primary Amine** **3. Aldehyde** **4. Nitrile**



[C:10]=[C:9]-[C:1](=[O:11])-[OD1]

[CX4:8]-[ND1:2]

[CX4:7]-[CD2:3]=O

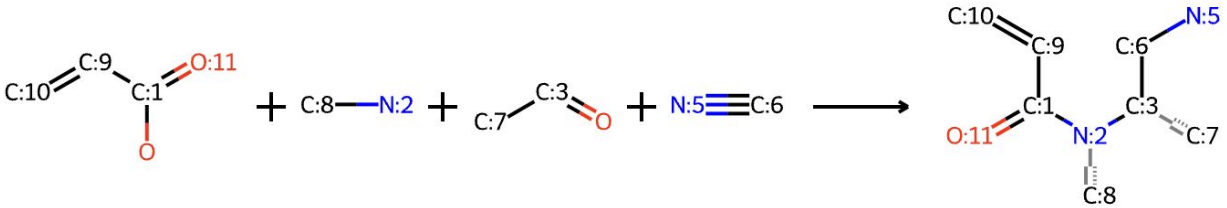
[C:6]#[N:5]

[C:10]=[C:9]-[C:1](=[O:11])-[N:2]([C:8])-[C:3]([CX4:7])-[C:6]-[N:5]

Transforming the Reaction into SMARTS: Combining it all together

- Now these parts can be all put together to get your reaction SMARTS!

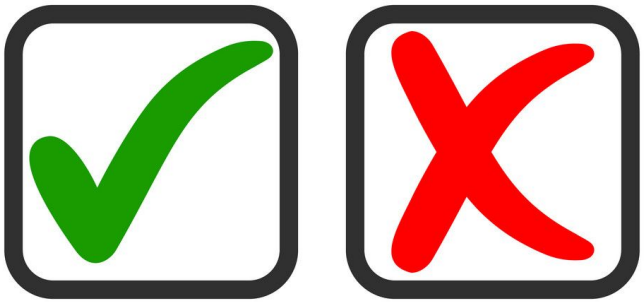
1. Allyl Acid **2. Primary Amine** **3. Aldehyde** **4. Nitrile**



[C:10]=[C:9]-[C:1](=[O:11])-[OD1].[CX4:8]-[ND1:2].[CX4:7]-[CD2:3]=O.[C:6]#[N:5]>>[C:10]=[C:9]-[C:1](=[O:11])-[N:2]([C:8])-[C:3]([CX4:7])-[C:6]-[N:5]

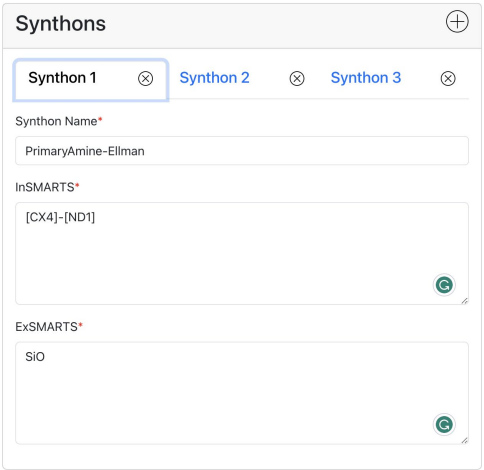
Transforming the Reaction into SMARTS: Filtering the Reagents

- Now it’s time to think about what should and should not be allowed for each reactant
- These will be our Inclusion/Exclusion SMARTS



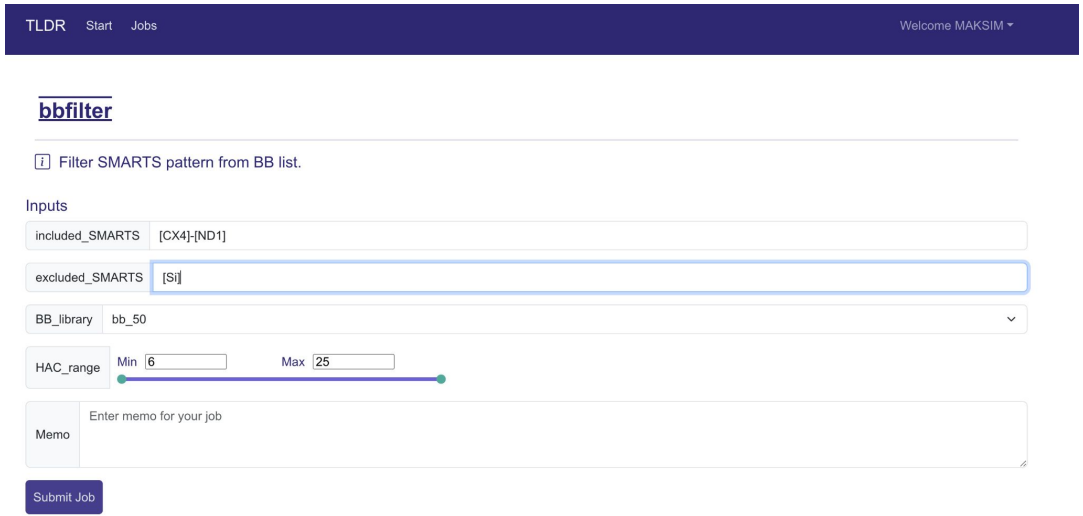
Transforming the Reaction into SMARTS: Inclusion/Exclusion SMARTS

- Inclusion SMARTS were already made earlier though making the reaction SMARTS!
- As for Exclusion SMARTS it can be and string of molecules that should not work as a reactant. Example: Anything with SiO in it



Transforming the Reaction into SMARTS: Testing Inclusion/Exclusion SMARTS

- You can use the bbfilter module in TLDR to test Inclusion/Exclusion SMARTS to see if you get sensible molecules or if you're filtering too many!



Enumeration and Debugging!

- Once you feel more or less confidant with your Reaction and Inclusion/Exclusion SMARTS it is time to run some Enumeration to test your SMARTS!

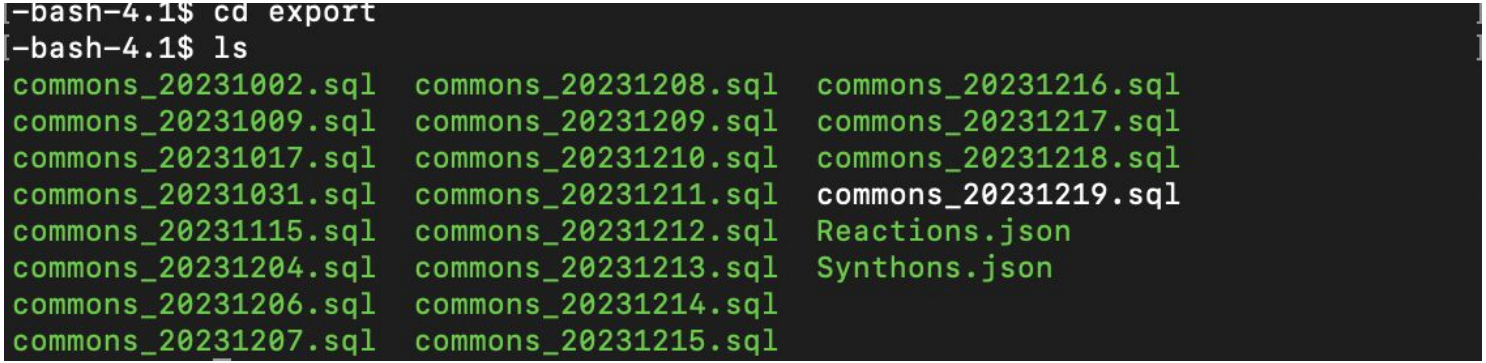
```
-bash-4.1$ ls
bb50     chbrbb1123         Filtered  NirMADE              scripts
BU       chbrbb1223         innovabb  otavabb              sialbb
chbr     ChemistryCommons   junk      README               sialbb1123
chbrbb   emolbb             MADE      REAL_Space_22Q3_29B  Test
-bash-4.1$ cd bb50
-bash-4.1$ ls
CC-10       CC-122.22  CC-140    CC-36.2  CC-4.4   CC-69.2  CC-80.2  CC-93.3
CC-108      CC-122.23  CC-142    CC-39.1  CC-53    CC-69.3  CC-92.1  CC-94.1
CC-122.11   CC-122.31  CC-142.1  CC-39.2  CC-68.1  CC-70.1  CC-92.2  CC-94.2
CC-122.12   CC-122.32  CC-143    CC-4.1   CC-68.2  CC-70.2  CC-92.3  CC-94.3
CC-122.13   CC-122.33  CC-148    CC-4.2   CC-68.3  CC-70.3  CC-93.1
CC-122.21   CC-138     CC-36.1   CC-4.3   CC-69.1  CC-80.1  CC-93.2
-bash-4.1$
```

Exporting Chemistry Commons Reaction

-Once you have made your reaction on Chemistry commons it’s time to export your JSON containing the reaction into the command line!

Finding Reaction and Synthon JSONs

-Once run the export script you can find the Reactions.json and Synthons.json in the export directory



Picking library to Enumerate

- From experience it's best to pick a larger library to Enumerate to make sure there are enough molecules in the database, such as **bb50**.
- However, if you need quick results I recommend **Sialbb**.

```
-bash-4.1$ cd /nfs/exj/Fe/Filtered/
-bash-4.1$ ls
bb50    emolbb   innovabb  otavabb-v            sialbb
chbrbb  indexed  MADE      REAL_Space_22Q3_29B  Test
-bash-4.1$
```

Filtering Synthons

- Go into the library of your choosing and run the filtering command below for each synthon

```
[-bash-4.1$ cd sialbb/                                                    ]
[-bash-4.1$ ls                                                            ]
 sialbb_synthon_100_ReagentsForOlefination-Wittig
 sialbb_synthon_101_ReagentsForOlefination-HornerWadsworthEmmonsReagents
 sialbb_synthon_102_SecondaryAmines-SecAminesAliphaticPolycyclic
 sialbb_synthon_103_SecondaryAmines-secHet-Anilines
 sialbb_synthon_104_SecondaryAmines-Cyc-Aliphatic
 sialbb_synthon_105_SecondaryAmines-Cyc-Benzylic
 sialbb_synthon_106_SecondaryAmines-SecAminesAliphaticFluorinated
 sialbb_synthon_107_SecondaryAmines-diArylAmines
 sialbb_synthon_108_SecondaryAmines-Cyc-Het-benzylic
 sialbb_synthon_109_SecondaryAmines-Cyc-Anilines
 sialbb_synthon_10_Anhydrides-AcyclicAnhydrides
```

```
sialbb_synthon_9_Acylhalides-N-Acylhalides
-bash-4.1$ bash /nfs/exj/Fe/scripts/bb_wrapper.bash /nfs/exj/Fe/Filtered/indexed
/<catalog>/<catalog>.smi <id> /nfs/exj/Fe/scripts/common_db/export/Synthons.json
```

Enumeration: Part 1 of 3

1. First step is to source the environment:

```
-bash-4.1$ source /nfs/home/ak87/exa/UCSF/SynthI/BESPOKE/arthor-env/bin/activate
```

2. Next step is to go into Library Enumeration Directory or create your own!

```
(arthor-env) -bash-4.1$ cd /nfs/exj/Fe
(arthor-env) -bash-4.1$ ls
bb50     chbrbb1123        Filtered  NirMADE             scripts
BU       chbrbb1223        innovabb  otavabb             sialbb
chbr     ChemistryCommons  junk      README              sialbb1123
chbrbb   emolbb            MADE      REAL_Space_22Q3_29B Test
(arthor-env) -bash-4.1$ cd sialbb
(arthor-env) -bash-4.1$ ls
CC-10       CC-122.22  CC-140   CC-4.1   CC-68.2  CC-70.2  CC-92.3  CC-94.3
CC-108      CC-122.23  CC-143   CC-4.2   CC-68.3  CC-70.3  CC-93.1
CC-122.11   CC-122.31  CC-36.1  CC-4.3   CC-69.1  CC-80.1  CC-93.2
CC-122.12   CC-122.32  CC-36.2  CC-4.4   CC-69.2  CC-80.2  CC-93.3
CC-122.13   CC-122.33  CC-39.1  CC-53    CC-69.3  CC-92.1  CC-94.1
```

Enumeration: Part 2 of 3

- Inside create a directory using the following Syntax: mkdir **CC-<Reaction ID>**
- Go inside the directory and run the first enumeration script:

```
(arthor-env) -bash-4.1$ python3 /nfs/exj/Fe/scripts/Enumeratorsplit.py -i <Synthonfile1> <Synthon
file2> -r CC-<Reaction ID> -o <library> -e /mnt/nfs/exj/Fe/Filtered/Test/EXAMPLE_Library_Enumerat
ion_File_hg2_noProtected_logP350_HAC35_RotB7_V3_A-AK1.json -a /nfs/exj/Fe/scripts/common_db/expor
t/Reactions.json
```

Enumeration: Part 3 of 3

- Now after first script is finished running run the final script

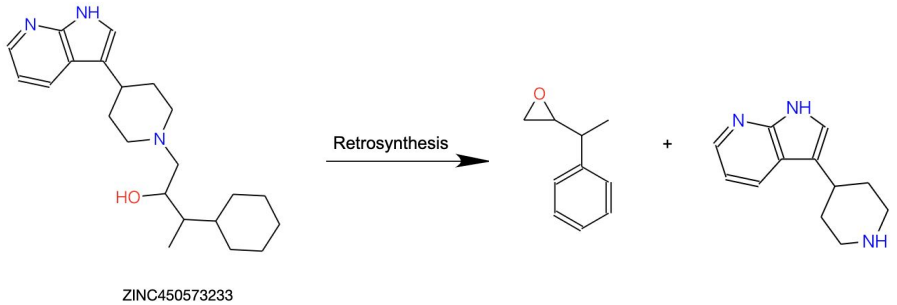
```
(arthor-env) -bash-4.1$ python3 /nfs/exj/Fe/scripts/Enumeratorcombine.py -i <pairing file> -r CC-
<Reaction ID> -o <library> -e /mnt/nfs/exj/Fe/Filtered/Test/EXAMPLE_Library_Enumeration_File_hg2_
noProtected_logP350_HAC35_RotB7_V3_A-AK1.json -a /nfs/exj/Fe/scripts/common_db/export/Reactions.j
son
```

Analog By Building Block Tool or AB3

-Retrosynthesis is the process of taking a molecule and breaking it apart into building blocks

-We can use this concept to access a new area of purchasable chemical space!

ZINC450573233 a hit for the σ2 receptor



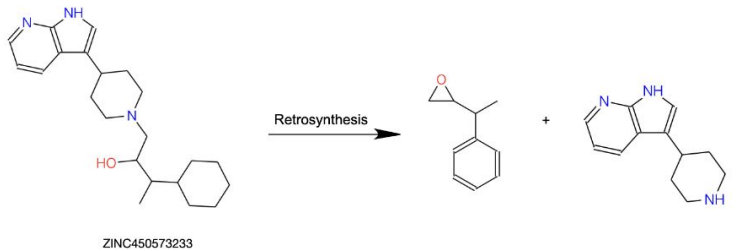
ZINC450573233

Alon A, Lyu J, Braz JM, Tummino TA, Craik V, O'Meara MJ, Webb CM, Radchenko DS, Moroz YS, Huang XP, Liu Y, Roth BL, Irwin JJ, Basbaum AI, Shoichet BK, Kruse AC. Structures of the σ receptor enable docking for bioactive ligand discovery. Nature. 2021 Dec;600(7890):759-764. doi: 10.1038/s41586-021-04175-x. Epub 2021 Dec 8. PMID: 34880501; PMCID: PMC8867396.

Analog By Building Block Tool or AB3

1. Use a sturdy retrosynthetic reaction to break down a molecule into building blocks
2. Find purchasable analogs of the building blocks by doing a similarity search
3. The respective building blocks can now be recombined with a forward reaction to find a final analog of the starting molecule

ZINC450573233 a hit for the σ2 receptor

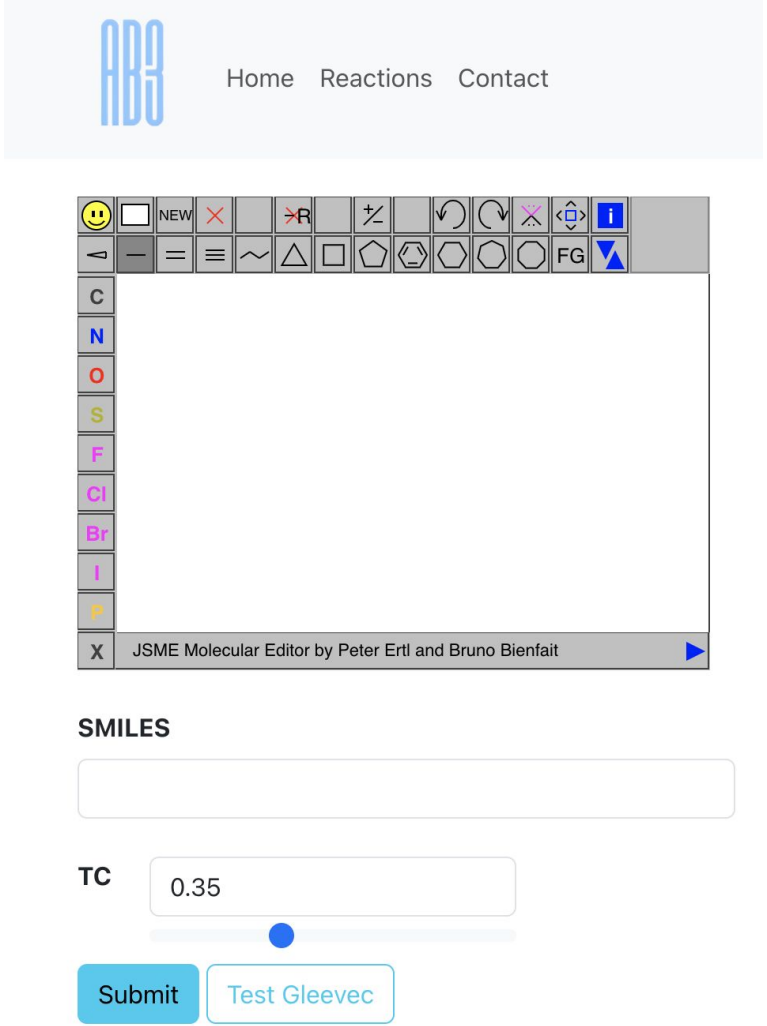


ZINC450573233

Alon A, Lyu J, Braz JM, Tummino TA, Craik V, O'Meara MJ, Webb CM, Radchenko DS, Moroz YS, Huang XP, Liu Y, Roth BL, Irwin JJ, Basbaum AI, Shoichet BK, Kruse AC. Structures of the σ receptor enable docking for bioactive ligand discovery. Nature. 2021 Dec;600(7890):759-764. doi: 10.1038/s41586-021-04175-x. Epub 2021 Dec 8. PMID: 34880501; PMCID: PMC8867396.

Interface

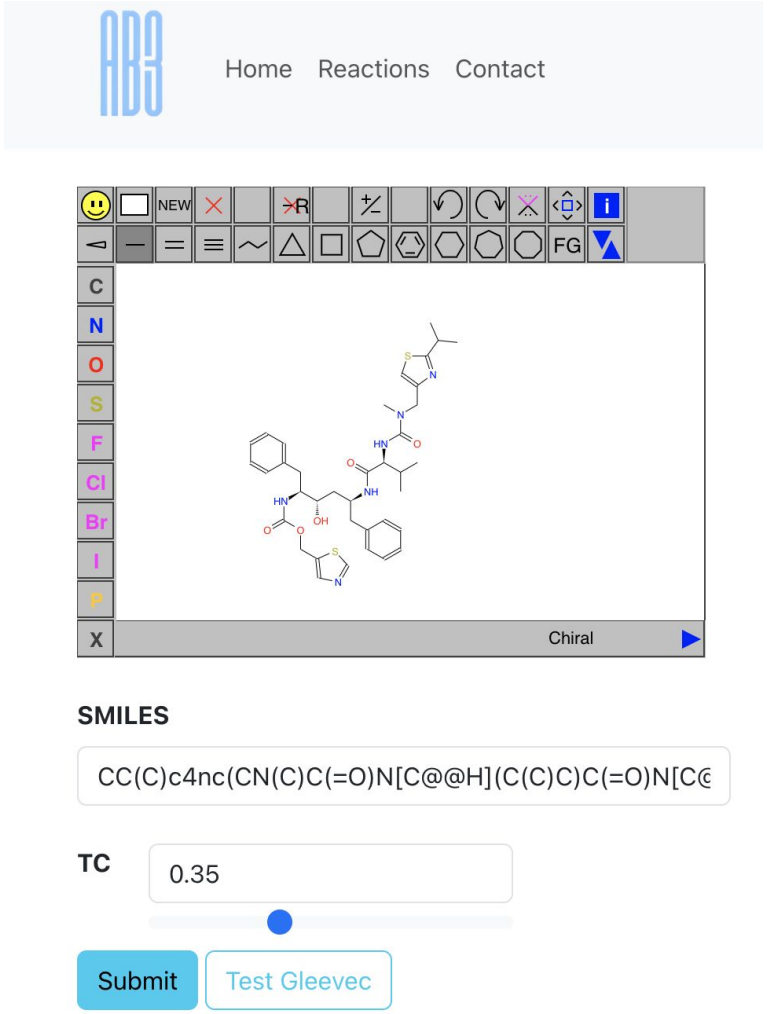
-Draw your molecule of interest or just put in the SMILES in the input box



Interface

-Draw your molecule of interest or just put in the SMILES in the input box

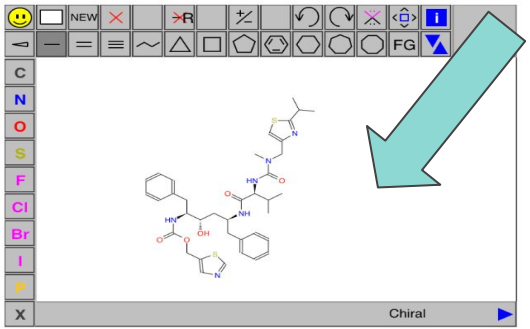
-Set your TC value and hit submit!



Finding Analogs of Novel Molecules: 1 of 8

Home Reactions Contact

Ritonavir a new FDA approved drug to treat HIV/AIDS

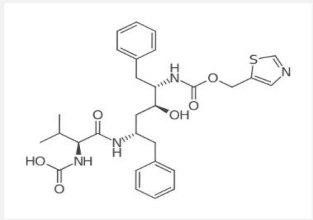
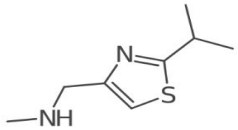


SMILES

CC(C)c4nc(CN(C)C(=O)N[C@@H](C(C)C)C(=O)N[C@

TC 0.35

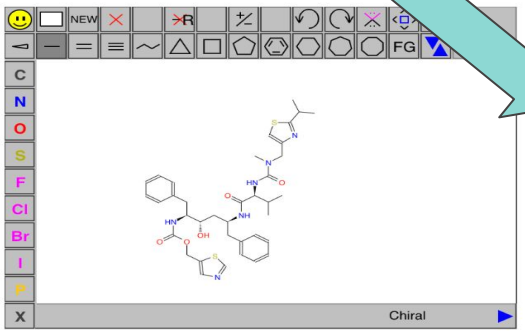
Submit Test Gleevec

Rxn/Rxn ID	BBs:16	BB SMILES	Analogs
AMIDE CC-4			
		CC(C)[C@H](NC(=O)O)C(=O)N[C@@H](Cc1ccccc1)C[C@H](O)[C@H](Cc1ccccc1)NC(=O)OCc1cncs1	Analogs
		CNCc1csc(C(C)C)n1	Analogs

Finding Analogs of Novel Molecules: 2 of 8

Home Reactions Contact

Retrosynthesis results on the right

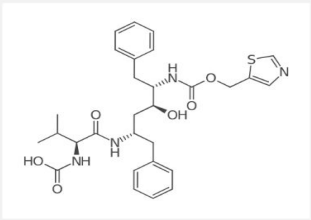
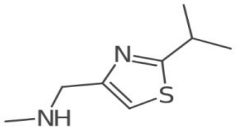


SMILES

CC(C)c4nc(CN(C)C(=O)N[C@@H](C(C)C)C(=O)N[C@

TC 0.35

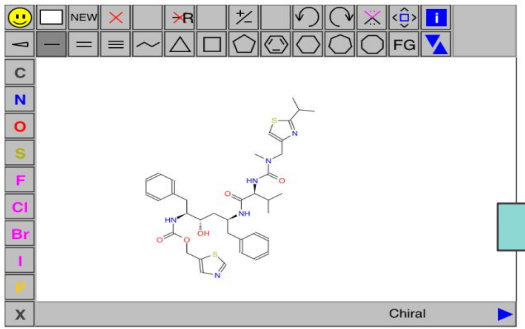
Submit Test Gleevec

Rxn/Rxn ID	BBs:16	BB SMILES	Analogs
AMIDE CC-4			
		CC(C)[C@H](NC(=O)O)C(=O)N[C@@H](Cc1ccccc1)C[C@H](O)[C@H](Cc1ccccc1)NC(=O)OCc1cncs1	Analogs
		CNCc1csc(C(C)C)n1	Analogs

Finding Analogs of Novel Molecules: 3 of 8

Home Reactions Contact

Each retrosynthesis has atleast two results

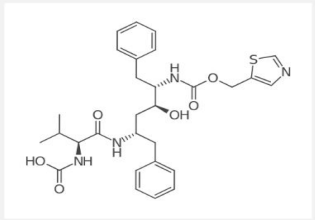
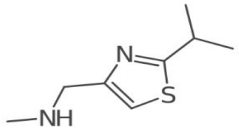


SMILES

CC(C)c4nc(CN(C)C(=O)N[C@@H](C(C)C)C(=O)N[C@

TC 0.35

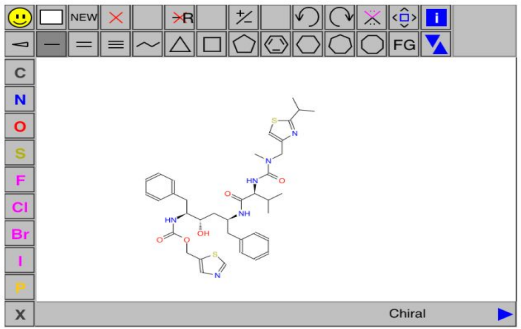
Submit Test Gleevec

Rxn/Rxn ID	BBs:16	BB SMILES	Analogs
AMIDE CC-4		CC(C)[C@H](NC(=O)O)C(=O)N[C@@H](Cc1ccccc1)C[C@H](O)[C@H](Cc1ccccc1)NC(=O)OCc1cncs1	Analogs
		CNCc1csc(C(C)C)n1	Analogs

Finding Analogs of Novel Molecules: 4 of 8

Home Reactions Contact

Download Retrosynthesis Results

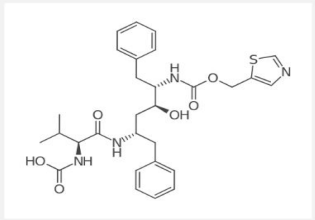
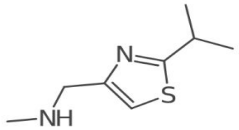


SMILES

CC(C)c4nc(CN(C)C(=O)N[C@@H](C(C)C)C(=O)N[C@

TC 0.35

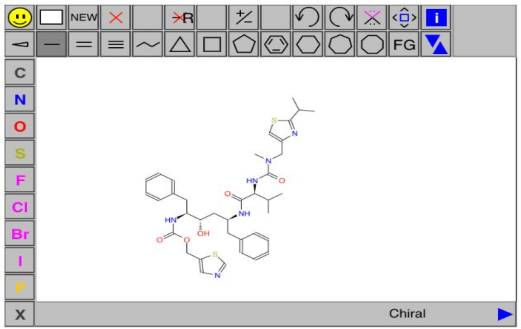
Submit Test Gleevec

Rxn/Rxn ID	BBs:16	BB SMILES	Analogs
AMIDE CC-4			
		CC(C)[C@H](NC(=O)O)C(=O)N[C@@H](Cc1ccccc1)C[C@H](O)[C@H](Cc1ccccc1)NC(=O)OCc1cncs1	Analogs
		CNCc1csc(C(C)C)n1	Analogs

Finding Analogs of Novel Molecules: 5 of 8

Home Reactions Contact

Press Analogs to View
Purchasable Analogs of BB

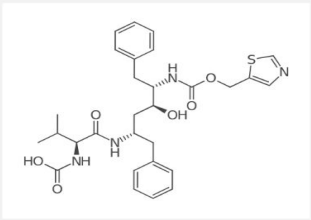
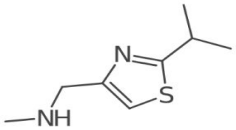


SMILES

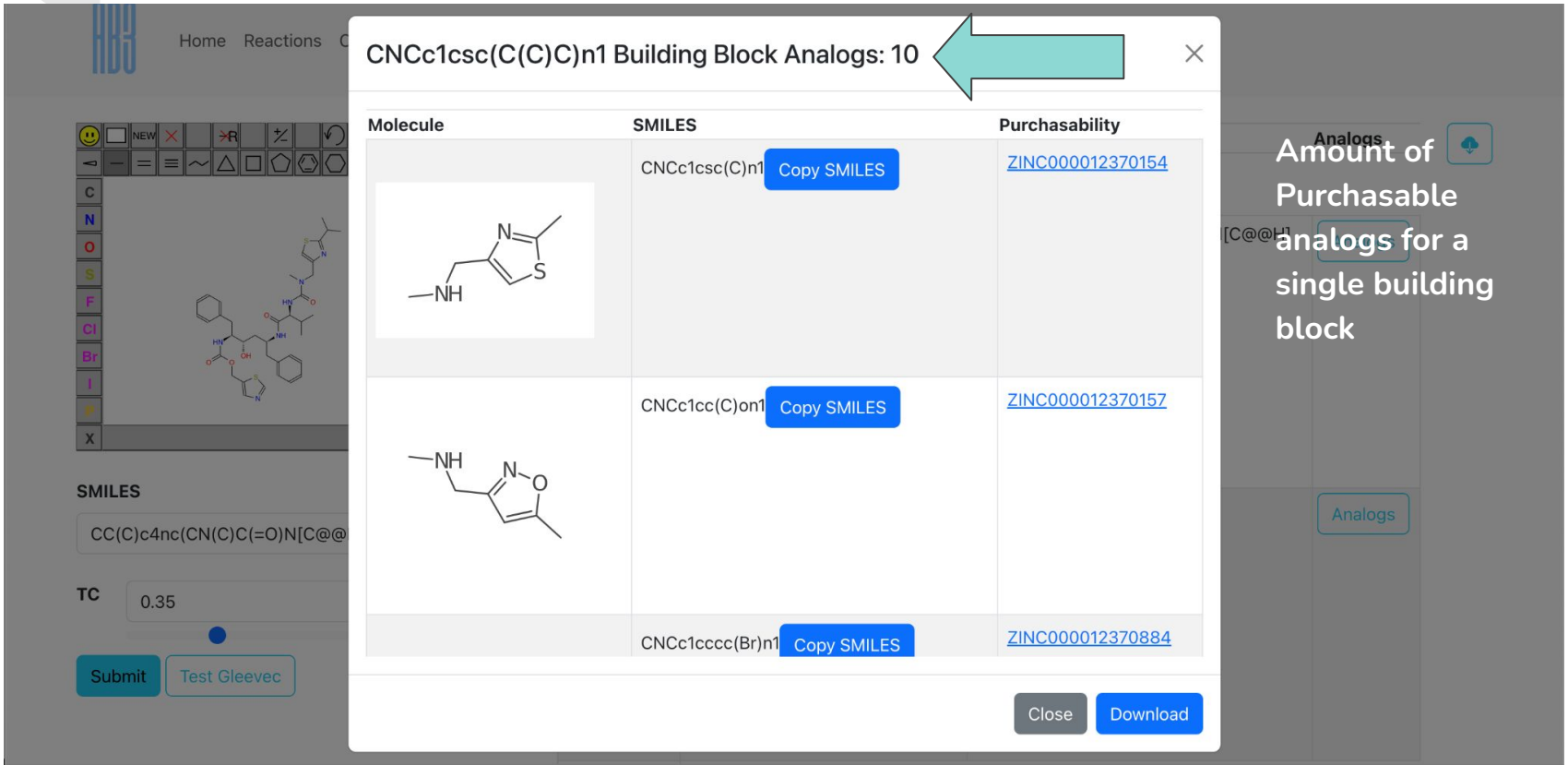
CC(C)c4nc(CN(C)C(=O)N[C@@H](C(C)C)C(=O)N[C@

TC 0.35

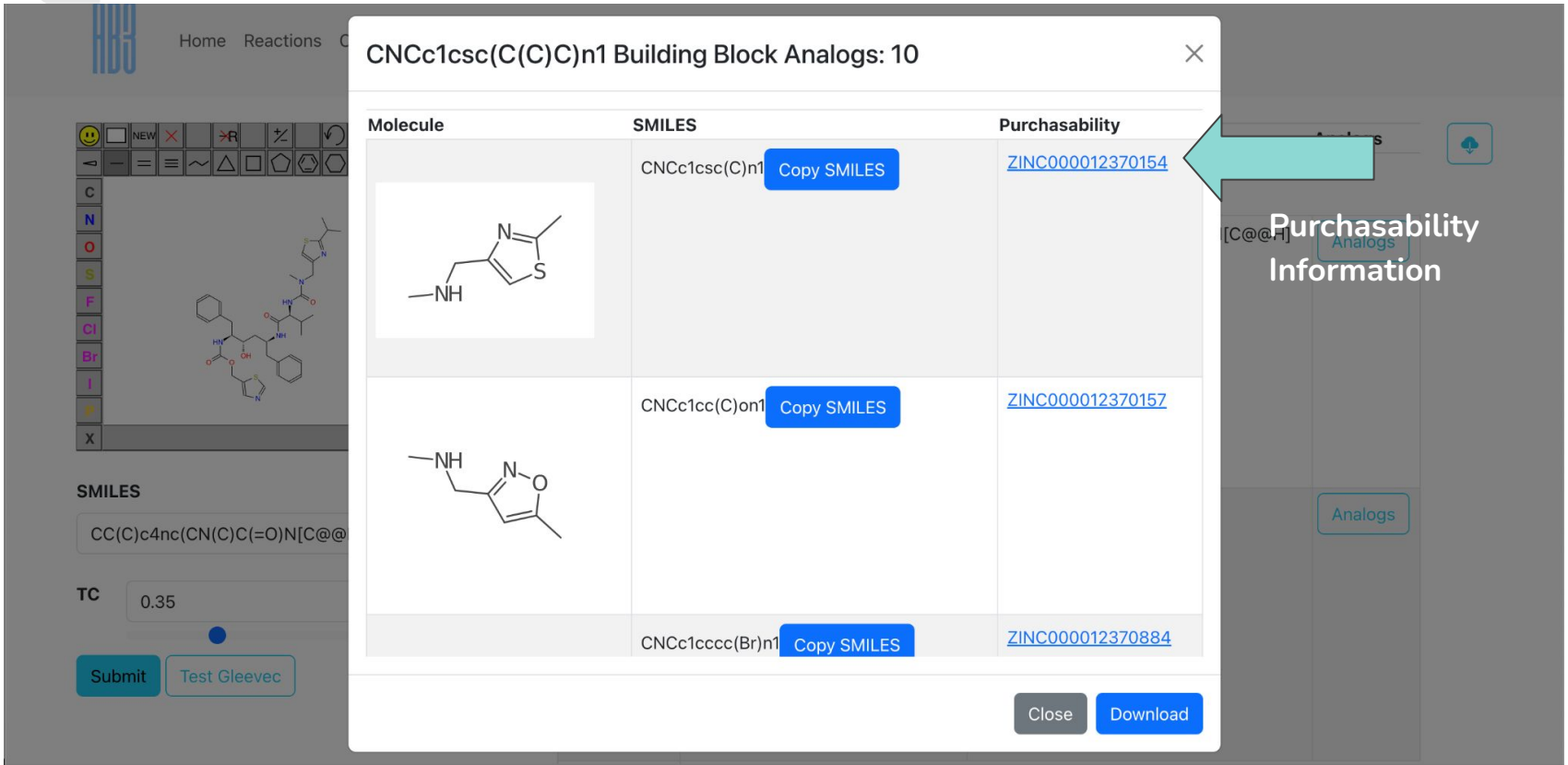
Submit Test Gleevec

Rxn/Rxn ID	BBs:16	BB SMILES	Analogs
AMIDE CC-4			
		CC(C)[C@H](NC(=O)O)C(=O)N[C@@H](Cc1ccccc1)C[C@H](O)[C@H](Cc1ccccc1)NC(=O)OCc1cncs1	Analogs
		CNCc1csc(C(C)C)n1	Analogs

Finding Analogs of Novel Molecules: 6 of 8



Finding Analogs of Novel Molecules: 7 of 8



Finding Analogs of Novel Molecules: 8 of 8