

§ SFDX force:flyingonthecloud:cheatsheet -p -m molina

ORGS

authorize an org using the web login flow
\$ sfdx force:auth:web:login

list all orgs you've created or authenticated to
\$ sfdx force:org:list

set config vars for the Salesforce CLI
\$ sfdx force:config:set defaultusername=me@my.org
\$ sfdx force:config:set defaultdevhubusername=me@myhub.org

set username aliases for the Salesforce CLI
\$ sfdx force:alias:set youralias=user@example.com

assign a permission set to one or more users of an org
\$ sfdx force:user:permset:assign -n permsetName

PACKAGING

install a package in the target org
\$ sfdx force:package:install

create a package version
\$ sfdx force:package:version:create -p package -a version

UP/DOWN

push source to a scratch org from the project
\$ sfdx force:source:push

pull source from the scratch org to the project
\$ sfdx force:source:pull

deploy metadata to an org using Metadata API
\$ sfdx force:mdapi:deploy -d deploydir

retrieve metadata from an org using Metadata API
\$ sfdx force:mdapi:retrieve -p packagenames -r retrievetargetdir

deploy a specific Apex class to a non-source-tracked org
\$ sfdx force:source:deploy -m ApexClass:MyApexClass

retrieve all metadata components listed in a manifest (package.xml) from a non-source-tracked org
\$ sfdx force:source:retrieve -x path/to/package.xml

convert source into Metadata API format
\$ sfdx force:source:convert -r path/to/source -d path/to/outputdir -n 'My Package'

convert metadata from the Metadata API format into the source format
\$ sfdx force:mdapi:convert -r path/to/metadata -d path/to/outputdir

LOCAL

create an Apex class
\$ sfdx force:apex:class:create -n YourClass

create an Apex trigger
\$ sfdx force:apex:trigger:create -n MyTrigger

create a bundle for an Aura component or a Lightning web component
\$ sfdx force:lightning:component:create -n mycomponent --type lwc

create a visualforce page
\$ sfdx force:visualforce:page:create -n mypage

create a new SFDX project
\$ sfdx force:project:create -n projectname -p defaultpackagedir -x

OTHER

create a record
\$ sfdx force:data:record:create -s Account -v "Name='Universal Containers'"

describe an object
\$ sfdx force:schema:object:describe -s MyObject__c

display current org's limits
\$ sfdx force:limits:api:display

