OpenShift Client (oc)

Usage:

```
oc COMMAND_TYPE
Where,
     COMMAND_TYPE
                        BASIC, BUILD/DEPLOY, MANAGEMENT, TROUBLESHOOTING, SETTINGS, ADVANCED, OTHERS.
     BASIC
                        login, new-project, new-app, status, project, projects, explain
                        new-build, start-build, cancel-build, rollout, rollback, cancel-build,
     BUILD/DEPLOY
                        import-image, tag
     MANAGEMENT
                        create, apply, get, describe, edit, set, label, annotate, expose, delete,
                        scale, autoscale, secrets, serviceaccounts
     TROUBLESHOOTING
                        logs, rsh, rsync, port-forward, debug, exec, proxy, attach, run, cp, wait
     SETTINGS
                        logout, config, whoami, completion
     ADVANCED
                        adm, replace, patch, process, extract, observe, policy, auth, image, registry,
                        idle, api-versions, api-resources, cluster-info, diff, kustomize
     OTHERS
                        ex, help, plugin, version
     For a description of all the subcommand listed above, type oc --help
     Best method for getting help: oc COMMAND --help
```

Configuration Files:

```
~/.kube/config
```

Most often used common syntax:

Basic Operations:

Login Operations:

```
oc login [-u USER] [-p PASSWORD] API_URL
oc whoami [--show-console] [--show-token] # --show-token = -t
oc logout
oc login -u devuser https://api.mycluster.example.com:6443
```

Project Operations:

Creating Application Resources:

Create App using existing Image/ImageStream

Create App using Source-To-Image (S2I/STI) or Dockerfile

```
oc new-app -i php https://github.com/user/myapp#branch --context-dir mydb
oc new-app -i php:7.1 https://github.com/user/yourapp
oc new-app php:7.1~https://github.com/user/superapp
```

Create App using Template

API Resources (Resource Types) Operations:

```
oc api-resources
                                   # List all resource types. Any namespaced resource
                                   command, accepts -n PROJECT option
oc explain RESOURCE[.FIELD]{[.FIELD]}...
                                                 # Learn resource structure
oc get RESOURCE [NAME] [-n PROJECT] [-o (json|yaml)|wide] # Show resource info
oc describe RESOURCE NAME
                                                # Show more info
oc edit RESOURCE NAME [-o json]
                                                 # Edit resource definition
oc patch ....
                                                # Refer to "oc help patch"
                                                # refer to "oc help set RESOURCE"
oc set env|probe|volumes|sa|triggers|...
oc delete RESOURCE NAME
oc delete all --all
                                                # delete all resource from project
oc delete all -l LABEL
                                                # delete all resources having LABEL
```

Scaling Pods:

Image Operation:

```
oc import-image NAME [--confirm] --from IMAGE_URI [--insecure]
```

Troubleshooting:

```
oc logs bc|dc|build|pod NAME [-f]
oc get events
oc rsh POD_NAME [CMD]
oc cp FROM TO

where,
    FROM = TO = [POD_NAME:]PATH
```

```
oc cp ./file1 mypod-1-ab123:/mnt/testing/fileX
oc cp mypod-1-12345:/index.html /tmp/backup.html
```

Creating Resources:

```
oc expose dc/myapp
oc expose svc/myapp --name myroute --hostname myroute.apps.example.com
```