|  |  |
| --- | --- |
| **config\_ONOS\_controller.py module** | |
| **S\_ONOSController(cmd.Cmd)**  Console interface를 위해 cmd 모듈을 상속 ( 모듈 경로 : /usr/lib/python2.7)  Controller에 관련된 명령들을 수행  입력된 문자열에 해당하는 명령을 수행 | |
| Method  List | **C\_Start(ip, port)**  입력 받은 인자를 저장.  ip를 이용해 특정한 위치에(/conf) ip.json DB를 생성  ip.json랑 똑같은 파일이 존재하고 있으면 그 DB를 update한다  Console interface를 시작 |
| **do\_show(command)**  console에 show <command> 형식의 입력을 처리  정보 출력에 관련 된 명령을 실행   |  |  | | --- | --- | | 커맨드 목록 | | | switches | 스위치들의 정보를 출력 | | hosts | 스위치에 연결된 호스트들의 정보를 출력 | | topology | 스위치와 호스트들이 연결 된 정보를 출력 | |
| **do\_switch(switch\_name)**  console에 switch <switch\_name> 형식의 입력을 처리  입력 된 switch\_name이 존재하는지 확인  입력 된 switch\_name에 해당하는 스위치의 console을 실행 |
| **do\_alias(controller\_name) #TODO**  controller를 특정 지을 수 있는(Unique) 이름을 입력 받아 현재 컨트롤러의 이름으로 사용 |
| **do\_help()**  controller console의 입력형식, 명령어 등 사용법을 출력 |
| **complete\_show(text, line, begidx, endidx)**  show command에 대한 help와 option을 출력 |
| **complete\_switch(text, line, begidx, endidx)**  flow command에 대한 help를 출력 |
| **complete\_alias(text, line, begidx, endidx) #TODO**  alias command에 대한 help를 출력 |

|  |  |
| --- | --- |
| **specific\_switch\_cmd.py module** | |
| **S\_ODLSwtich(cmd.Cmd)**  Console interface를 위해 cmd 모듈을 상속 ( 모듈 경로 : /usr/lib/python2.7)  Switch에 관련된 명령들을 수행  입력된 문자열에 해당하는 명령을 수행 | |
| Method  List | **S\_Start(url, dpid, sw\_alias, file\_path)**  입력 받은 인자를 저장  스위치 정보 출력  Console interface를 시작 |
| **do\_ports(command)**  console에 ports 형식의 입력을 처리  정보 출력에 관련 된 명령을 실행 |
| **do\_flow(command)**  console에 flow 형식의 입력을 처리  정보 출력에 관련 된 명령을 실행 |
| **do\_alias(switch\_name)**  switch를 특정 지을 수 있는(Unique) 이름을 입력 받아 현재 switch의 이름으로 사용 |
| **do\_help() #TODO**  switch console의 입력형식, 명령어 등 사용법을 출력 |
| **complete\_ports(text, line, begidx, endidx) #TODO**  show command에 대한 help와 option을 출력 |
| **complete\_flow(text, line, begidx, endidx) #TODO**  flow command에 대한 help를 출력 |
| **complete\_alias(text, line, begidx, endidx) #TODO**  alias command에 대한 help를 출력 |

|  |  |
| --- | --- |
| **refresh.py module**  ONOS의 RestAPI를 call 하여 결과 값을 반환 | |
| Method List | **refresh\_switches(url)**  url를 이용하여 controller가 관리하는 모든 스위치 정보 반환  retrun data\_switches |
| **Refresh\_topology(url)**  url를 이용하여 controller가 관리하는 네트워크 topology 정보 반환  retrun data\_topology |
| **refresh\_hosts(url)**  url를 이용하여 controller가 관리하는 네트워크의 모든 호스트들의 정보 반환  retrun data\_hosts |
| **refresh\_specific\_switch(url, switch\_dpid)**  url 및 switch\_dpid를 이용하여 특정한 스위치 정보 반환  retrun data\_specific\_switch |
| **refresh\_specific\_switch\_port(url, switch\_dpid)**  url 및 switch\_dpid를 이용하여 특정한 스위치의 모든 포트 정보 반환  retrun data\_port |
| **refresh\_specific\_switch\_flows(url, switch\_dpid)**  url 및 switch\_dpid를 이용하여 특정한 스위치를 거쳐가고 있는 flow 정보 반환  retrun data\_flows |
|  | |
| **global\_value.py module**  **global하게 사용될 값들을 저장** | |
| Value List | **LOCAL\_PATH**  프로그램을 실행한 위치 정보  = os.getcwd() |
| **CONF\_PATH**  생한 json DB들을 저장할 위치  = LOCAL\_PATH+’/conf’ |
| **REG\_CONF\_ALIAS**  등록한 controller 정보 저장 json DB  = ‘/registered\_controller.json’ |
| **CONTROLLER\_ALIAS**  Controller의 바뀐 이름 저장 json DB.  = ‘/controller\_alias.json’ |
| **FLOW\_PATH**  Flow 정보 저장 위치  = LOCAL\_PATH+’/flows’ |
| **devices = ‘**devices’ |
| **topology** = ‘topology’ |
| **hosts** = ‘hosts’ |
|  | **ports** = ‘ports’ |
|  | **flows** = ‘flows’ |
| **utils.py module**  “json 형식 file” create, update, read, write 관련된 methods | |
| Method  List | **make\_alias\_json(url, ip)**  입력 받은 url 및 ip 정보들을 이용해 해당한 controller에 대한 json 형식의 데이터 생성  json 데이터 반환 |
| **update\_alias\_json(url, ip, file\_path)**  입력 받은 url, ip, 및 이미 만들어진 DB를 이용해 해당한 controller에 대한 데이터 update  update된 데이터 반환 |
| **config\_json\_read(file\_path)**  json형식으로 되어 있는 string를 python 형식으리 데이터로 바뀜  python 형식으로 된 데이터 반환 |
| **config\_json\_write(file\_path, json\_data)**  python 형식으로 되어 있는 데이터를 json 형식으로 바뀌어서 특정한 위치에 저장 |

|  |  |
| --- | --- |
| **ONOS\_result\_print.py module**  Python 데이터 형식으로 되어 있는 데이터들을 입력으로 받아 console에 출력 | |
|  | **print\_all\_switches(data)**  controller가 관리하는 모든 스위치를 출력 |
| **print\_topology(data)**  controller가 관리하는 네트워크의 topology 정보 출력 |
| **print\_hosts(data)**  controller가 관리하는 네트워크에 연결되어 있는 모든 호스트들에 대한 정보를 출력 |
| **print\_specific\_switch(data)**  특정한 스위치에 대한 정보를 출력 |
| **print\_specific\_switch\_ports(data)**  특정한 스위치의 포트 정보를 출력 |
| **print\_specific\_switch\_flow(data)**  특정한 스위치의 flow 정보를 출력 |