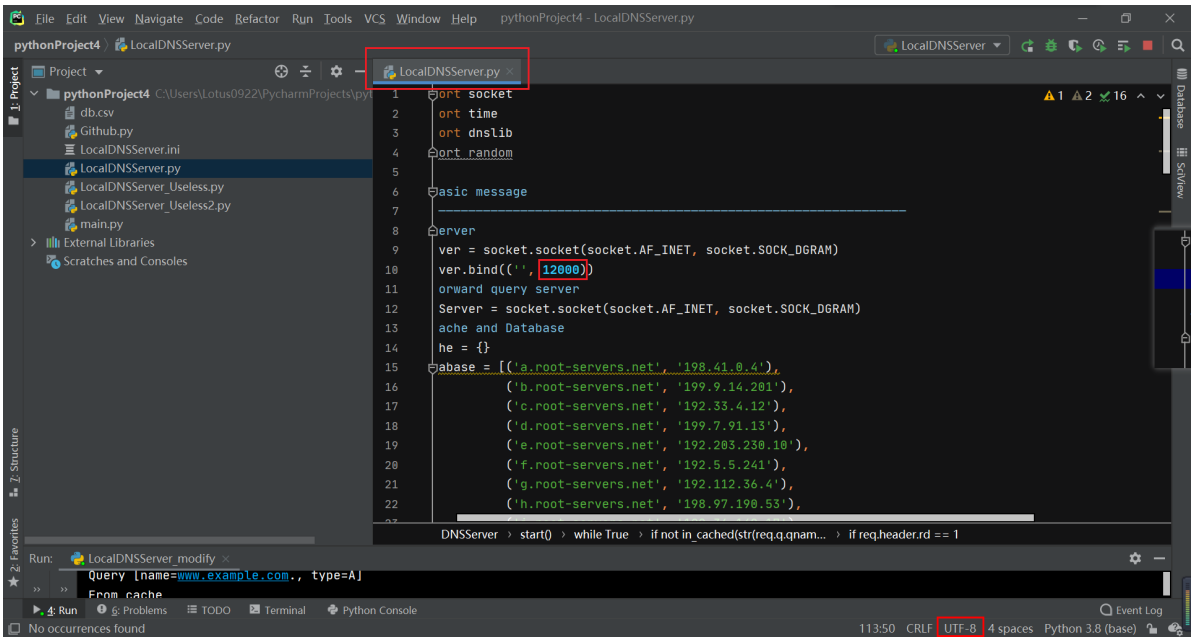
# CS305 Assignment5.3

The python source file should be named as `LocalDNSServer.py` using `UTF-8` as file coding, work on port `12000` of `127.0.0.1`, there is no need to zip the source file, Comments in code is MUST.




# 1

## (1)Listen and accept DNS queries from client

### support default query type: A type

```
D:\ChromeCoreDownloads\BIND9.11.23.x64>dig www.example.com @127.0.0.1 -p 12000
;; Warning: ID mismatch: expected ID 2992, got 0

; <<>> DiG 9.11.23 <<>> www.example.com @127.0.0.1 -p 12000
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 2992
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 13, ADDITIONAL: 27

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        1004    IN      A       93.184.216.34

;; AUTHORITY SECTION:
com.                    1429    IN      NS      a.gtld-servers.net.
com.                    1429    IN      NS      m.gtld-servers.net.
com.                    1429    IN      NS      f.gtld-servers.net.
com.                    1429    IN      NS      b.gtld-servers.net.
com.                    1429    IN      NS      j.gtld-servers.net.
com.                    1429    IN      NS      e.gtld-servers.net.
com.                    1429    IN      NS      k.gtld-servers.net.
com.                    1429    IN      NS      l.gtld-servers.net.
```

# (2)Maintain a Cache

## Cache should be read

```
# get DNS responce format from cache
def from_cache_get_DNS_format(qname, qtype, qheader):
    # create response message from cache
    res = dnslib.DNSRecord(
        dnslib.DNSHeader(id=qheader.id, q=qheader.q, r=cache[qname][qtype][2], auth=cache[qname][qtype][3],
                         ra=cache[qname][qtype][4]), q=dnslib.DNSQuestion(qname, qtype), rr=cache[qname][qtype][5],
        auth=cache[qname][qtype][6], ar=cache[qname][qtype][7])
    res.header.qr = 1
    cacheMessage = bytes(res.pack())
    return cacheMessage
```

## Cache should be write while response is coming

```
# add message to cache
def add_to_cache(qname, qtype, msg):
    record = dnslib.DNSRecord.parse(msg)

    # calculate the minimum ttl
    cur_t = int(time.time())  # record current time
    ttl = 1000000000000000
    for rr in record.rr:
        ttl = min(ttl, rr.ttl)

    # store all information in cache
    cache_format(cur_t, ttl, record, qname, qtype)


def cache_format(cur_t, ttl, record, qname, qtype):
    cache[str(qname)][qtype] = [cur_t, ttl, record.header.a, record.header.auth, record.header.ar,
                                record.rr, record.auth, record.ar]

```

## While the query is new, cache the query and response

```
class IterativeQuery(object):
    @staticmethod
    def start(msg, req):
        print('From iterative query')
        # forward query to a public dns server
        GoogleDNS, GooglePort = '8.8.8.8', 53
        farServer.sendto(msg, (GoogleDNS, GooglePort))
        msg = farServer.recv(2048)
        # add new response to cache
        add_to_cache(str(req.q.qname), req.q.qtype, msg)
        # use cache to create answer
        resmsg = from_cache_get_DNS_format(str(req.q.qname), req.q.qtype, req.header)
        return resmsg
```

```
            tempReq2 = dnslib.DNSRecord.parse(msg2)
            tempReq2.header.a += 1
            tmp.extend(tempReq2.rr)
            tempReq2.rr = tmp
            # add to cache
            msg2 = bytes(tempReq2.pack())
            add_to_cache(str(req.q.qname), req.q.qtype, msg2)
            resmsg = from_cache_get_DNS_format(str(req.q.qname), req.q.qtype, req.header)
            break
        recursiveSerer = str(tempReq.auth[random.randint(0,tempReq.header.auth - 1)].rdata)
    return resmsg
```

## (3)Maintain a list of records

About the domain name and IPv4 address of Root DNS servers

```
database = [('a.root-servers.net', '198.41.0.4'),
            ('b.root-servers.net', '199.9.14.201'),
            ('c.root-servers.net', '192.33.4.12'),
            ('d.root-servers.net', '199.7.91.13'),
            ('e.root-servers.net', '192.203.230.10'),
            ('f.root-servers.net', '192.5.5.241'),
            ('g.root-servers.net', '192.112.36.4'),
            ('h.root-servers.net', '198.97.190.53'),
            ('i.root-servers.net', '192.36.148.17'),
            ('j.root-servers.net', '192.58.128.30'),
            ('k.root-servers.net', '193.0.14.129'),
            ('l.root-servers.net', '199.7.83.42'),
            ('m.root-servers.net', '202.12.27.33')]
```

## (4) Search related DNS response in Cache according to the received query

```
D:\ChromeCoreDownloads\BIND9.11.23.x64>dig www.example.com @127.0.0.1 -p 12000
;; Warning: ID mismatch: expected ID 60133, got 0

; <<>> DiG 9.11.23 <<>> www.example.com @127.0.0.1 -p 12000
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60133
;; flags: qr rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        855     IN      A       93.184.216.34

;; Query time: 0 msec
;; SERVER: 127.0.0.1#12000(127.0.0.1)
;; WHEN: Fri Oct 23 21:56:42 中国标准时间 2020
;; MSG SIZE  rcvd: 49
```

```
local DNSSever start at: 127.0.0.1:12000
Query [name=www.example.com., type=A]
From recursive query
Query [name=www.example.com., type=A]
From cache
Query [name=www.example.com., type=A]
From cache
```

```python
# in cached
if in_cached(str(req.q.qname), req.q.qtype):  # check if the query is cached
    print('From cache')
    resmsg = from_cache_get_DNS_format(str(req.q.qname), req.q.qtype, req.header)  # get reply from cache
    server.sendto(resmsg, addr)
```

## (5) Invoke DNS queries to other DNS server if needed and cache the new response

```python
# Neither in cache nor in database
if not in_cached(str(req.q.qname), req.q.qtype) and not in_database(req.q.qname):
    if req.header.rd == 0:
        resmsg = IterativeQuery.start(msg, req)
    if req.header.rd == 1:
        resmsg = RecursiveQuery.start(msg, req)
```

# 2.

## dig

```
D:\ChromeCoreDownloads\BIND9.11.23.x64>dig www.example.com @127.0.0.1 -p 12000
;; Warning: ID mismatch: expected ID 2992, got 0

; <<>> DiG 9.11.23 <<>> www.example.com @127.0.0.1 -p 12000
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 2992
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 13, ADDITIONAL: 27

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        1004    IN      A       93.184.216.34

;; AUTHORITY SECTION:
com.                    1429    IN      NS      a.gtld-servers.net.
com.                    1429    IN      NS      m.gtld-servers.net.
com.                    1429    IN      NS      f.gtld-servers.net.
com.                    1429    IN      NS      b.gtld-servers.net.
com.                    1429    IN      NS      j.gtld-servers.net.
com.                    1429    IN      NS      e.gtld-servers.net.
com.                    1429    IN      NS      k.gtld-servers.net.
com.                    1429    IN      NS      l.gtld-servers.net.
com.                    1429    IN      NS      c.gtld-servers.net.
com.                    1429    IN      NS      h.gtld-servers.net.
```

The first time invoke a query to www.example.com

```
D:\ChromeCoreDownloads\BIND9.11.23.x64>dig www.example.com @127.0.0.1 -p 12000
;; Warning: ID mismatch: expected ID 60133, got 0

; <<>> DiG 9.11.23 <<>> www.example.com @127.0.0.1 -p 12000
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60133
;; flags: qr rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        855     IN      A       93.184.216.34

;; Query time: 0 msec
;; SERVER: 127.0.0.1#12000(127.0.0.1)
;; WHEN: Fri Oct 23 21:56:42 中国标准时间 2020
;; MSG SIZE  rcvd: 49
```

## Wireshark

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 662 | 2020-10-24 14:14:34.830459 | 10.17.96.210 | a.root-servers.net | DNS | 98 | 128 | Standard query 0x6ac9 A www.example.com OPT |
| 663 | 2020-10-24 14:14:34.893079 | a.root-servers.net | 10.17.96.210 | DNS | 882 | 47 | Standard query response 0x6ac9 A www.example.com NS a.gtld-servers.net NS b.gtld-servers. |
| 666 | 2020-10-24 14:14:34.910183 | 10.17.96.210 | a.gtld-servers.net | DNS | 98 | 128 | Standard query 0x6ac9 A www.example.com OPT |
| 667 | 2020-10-24 14:14:34.926068 | a.gtld-servers.net | 10.17.96.210 | DNS | 134 | 51 | Standard query response 0x6ac9 A www.example.com NS a.iana-servers.net NS b.iana-servers. |
| 668 | 2020-10-24 14:14:34.927372 | 10.17.96.210 | b.iana-servers.net | DNS | 98 | 128 | Standard query 0x6ac9 A www.example.com OPT |
| 671 | 2020-10-24 14:14:35.103957 | b.iana-servers.net | 10.17.96.210 | DNS | 150 | 44 | Standard query response 0x6ac9 A www.example.com A 93.184.216.34 NS a.iana-servers.net NS |
| 966 | 2020-10-24 14:14:38.933329 | 10.17.96.210 | c.root-servers.net | DNS | 101 | 128 | Standard query 0xae52 A www.sustech.edu.cn OPT |
| 968 | 2020-10-24 14:14:39.114861 | c.root-servers.net | 10.17.96.210 | DNS | 444 | 49 | Standard query response 0xae52 A www.sustech.edu.cn NS ns.cernet.net NS c.dns.cn NS e.dns.c |
| 971 | 2020-10-24 14:14:39.125924 | 10.17.96.210 | b.dns.cn | DNS | 101 | 128 | Standard query 0xae52 A www.sustech.edu.cn OPT |
| 974 | 2020-10-24 14:14:39.160906 | b.dns.cn | 10.17.96.210 | DNS | 293 | 54 | Standard query response 0xae52 A www.sustech.edu.cn NS ns2.cuhk.hk NS ns2.cernet.net NS dns |
| 978 | 2020-10-24 14:14:39.178105 | 10.17.96.210 | deneb.dfn.de | DNS | 101 | 128 | Standard query 0xae52 A www.sustech.edu.cn OPT |
| 984 | 2020-10-24 14:14:39.381377 | deneb.dfn.de | 10.17.96.210 | DNS | 213 | 235 | Standard query response 0xae52 A www.sustech.edu.cn NS ns1.sustech.edu.cn NS ns2.sustech.ed |
| 987 | 2020-10-24 14:14:39.385868 | 10.17.96.210 | ns1.sustc.edu.cn | DNS | 101 | 128 | Standard query 0xae52 A www.sustech.edu.cn OPT |
| 988 | 2020-10-24 14:14:39.389003 | ns1.sustc.edu.cn | 10.17.96.210 | DNS | 243 | 62 | Standard query response 0xae52 A www.sustech.edu.cn CNAME sustech.edu.cn A 172.18.1.3 NS ns |

# 3

## The first query

```
D:\ChromeCoreDownloads\BIND9.11.23.x64>dig www.example.com @127.0.0.1 -p
12000

; <<>> DiG 9.11.23 <<>> www.example.com @127.0.0.1 -p 12000
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 2429
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        86400   IN      A       93.184.216.34

;; Query time: 335 msec
;; SERVER: 127.0.0.1#12000(127.0.0.1)
;; WHEN: Sat Oct 24 14:18:04 中国标准时间 2020
;; MSG SIZE  rcvd: 60
```

## The second query

```
D:\ChromeCoreDownloads\BIND9.11.23.x64>dig www.example.com @127.0.0.1 -p
12000

; <<>> DiG 9.11.23 <<>> www.example.com @127.0.0.1 -p 12000
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 9418
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        86400   IN      A       93.184.216.34

;; Query time: 0 msec
;; SERVER: 127.0.0.1#12000(127.0.0.1)
;; WHEN: Sat Oct 24 14:18:22 中国标准时间 2020
;; MSG SIZE  rcvd: 60
```

```
local DNSSever start at: 127.0.0.1:12000
Query [name=www.example.com., type=A]
From recursive query
Query [name=www.example.com., type=A]
From cache
```