

Lab 11

Question 1: Create a simple Node script that converts 'www.miu.edu' domain name to the equivalent IP address. (Search and learn 'dns' module, resolve4) - For this exercise, you need to look up Node.js API by yourself

```
const dns=require('dns')

//finding address for the valid url
dns.resolve('www.miu.edu',(err,address)=>{
  if(address){
    console.log(address);
  }
  if(err){
    console.log(err);
  }
});

//finding address for non-existence url
dns.resolve('www.miu.edux',(err,address)=>{
  if(address){
    console.log(address);
  }
  if(err){
    console.log("Address not found for the given url.");
  }
});
```

Question 2: Create a web server that's going to send a response of big image (bigger then 3MB) to any client that sends a request to your specified server:port. Use the best way for performance. (Try to solve this in many different ways)

```
const http = require("http");
const fs = require("fs");
const util = require("util");
const server = http
.createServer((req, res) => {
  console.log(`${req.url} ${req.method}`);
  res.setHeader("Content-Type", "image/jpg");
  res.statusCode = 200;
  //method one, async call back
  fs.readFile('./desc.jpg',(err,data)=>{
    res.end(data);
  });
  //method two: using pipe
  fs.createReadStream("./desc.jpg").pipe(res);
  //method Three: using sync
  res.end(fs.readFileSync("./source.jpg"));
  //method Four: using promise
  const promise = util.promisify(fs.readFile);
  promise("./source.jpg")
    .then((data) => {
      res.end(data);
    })
    .catch((err) => res.end(err));
})
.listen(3000);
console.log("Listening...");
```