# PILOT STUDENTS – Oct/Nov 2015 (this applies to you)

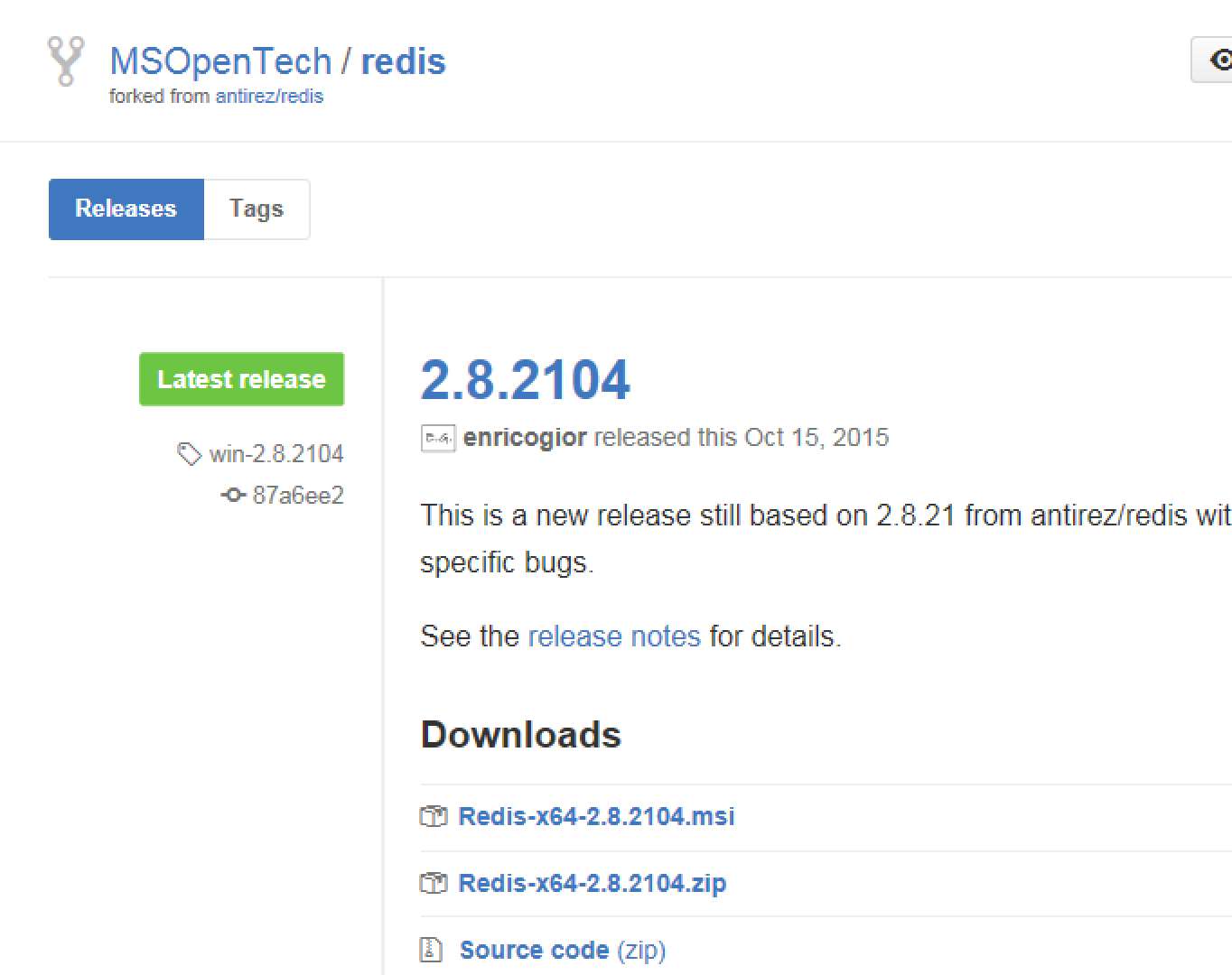
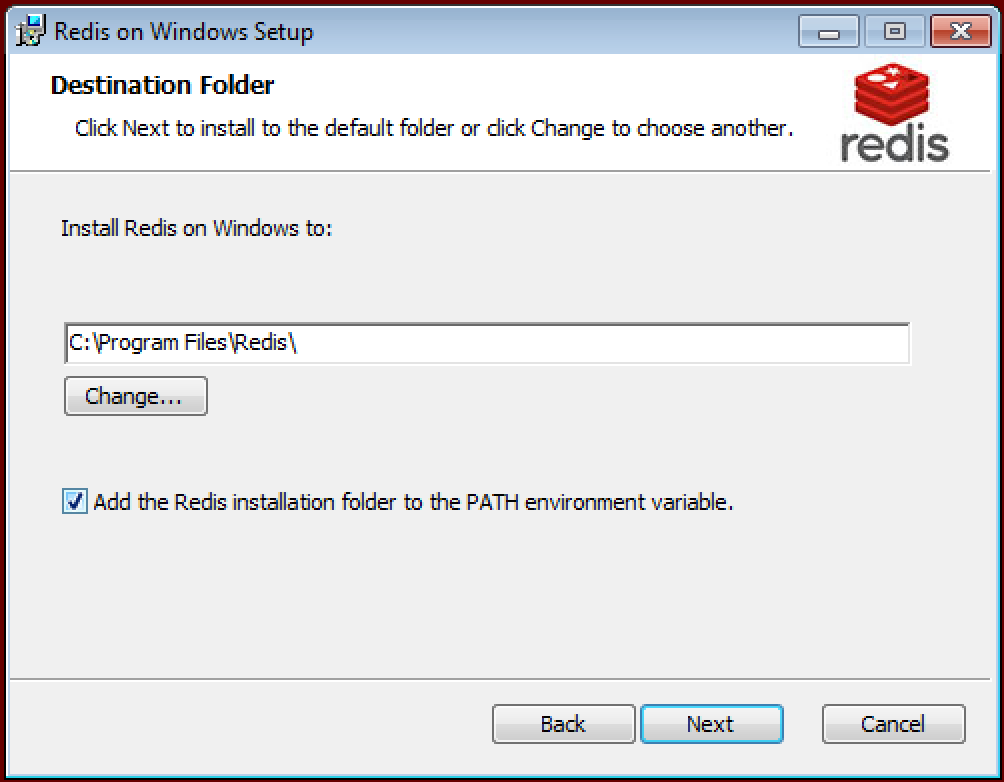
# (1) When you see an instruction to upload something, upload it to your desktop. If faculty needs it, they will ask for it. Contact info is in your FAQ.

# Intro to NoSQL Lab 2: KeyValue Stores / Redis HandsOn

## What Am I Supposed to Do?

1. Follow the instructions of the Lab.
2. Periodically, you’ll be instruction to **STOP** and do what is requested. Typically, we ask you to copy a screen shot. All requested deliverables should be uploaded in the training environment. **YOU MUST DO THIS IF YOU WANT TO PASS THE COURSE!**

## BEGIN LAB:

1. Make sure you have your node.js Setup from Lab 1 up and running. You’ll need the following packages unpacked and **ready** (i.E. npm install executed within their directory)
   1. nosql-lab-dataimport
   2. nosql-lab-redis
2. Install Redis
   1. Redis was ported to Windows by [Microsoft Open Technologies](https://msopentech.com/), Get Redis from their [Binary Releases on Github](https://github.com/MSOpenTech/redis/releases). Any > 2.8 build is sufficient, take the freshest one, get the MSI  
      
   2. Be sure to have it added to the path for added comfort
   3. Install ☺
   4. Ensure that redis is running by launching the redis-cli. You can use some commands like SET and GET to ensure it’s running – like for example this session:  
      127.0.0.1:6379> GET "test"  
      (nil)  
      127.0.0.1:6379> SET "test" "this is my funky testvalue"  
      OK  
      127.0.0.1:6379> GET "test"  
      "this is my funky testvalue"  
      127.0.0.1:6379> DEL "test"  
      (integer) 1  
      (end of sample)
   5. Check out the other cool features of redis @ redis.io
3. Import the data into redis
   1. In the Dataloader directory execute: node redis.js  
      (if you get strange errors re-check point 1 of this lab…there could be dependencies missing)
   2. Check in redis wether data is in (hint: there is a KEYS command)

## Perform the Lab

Launch the redis sample app

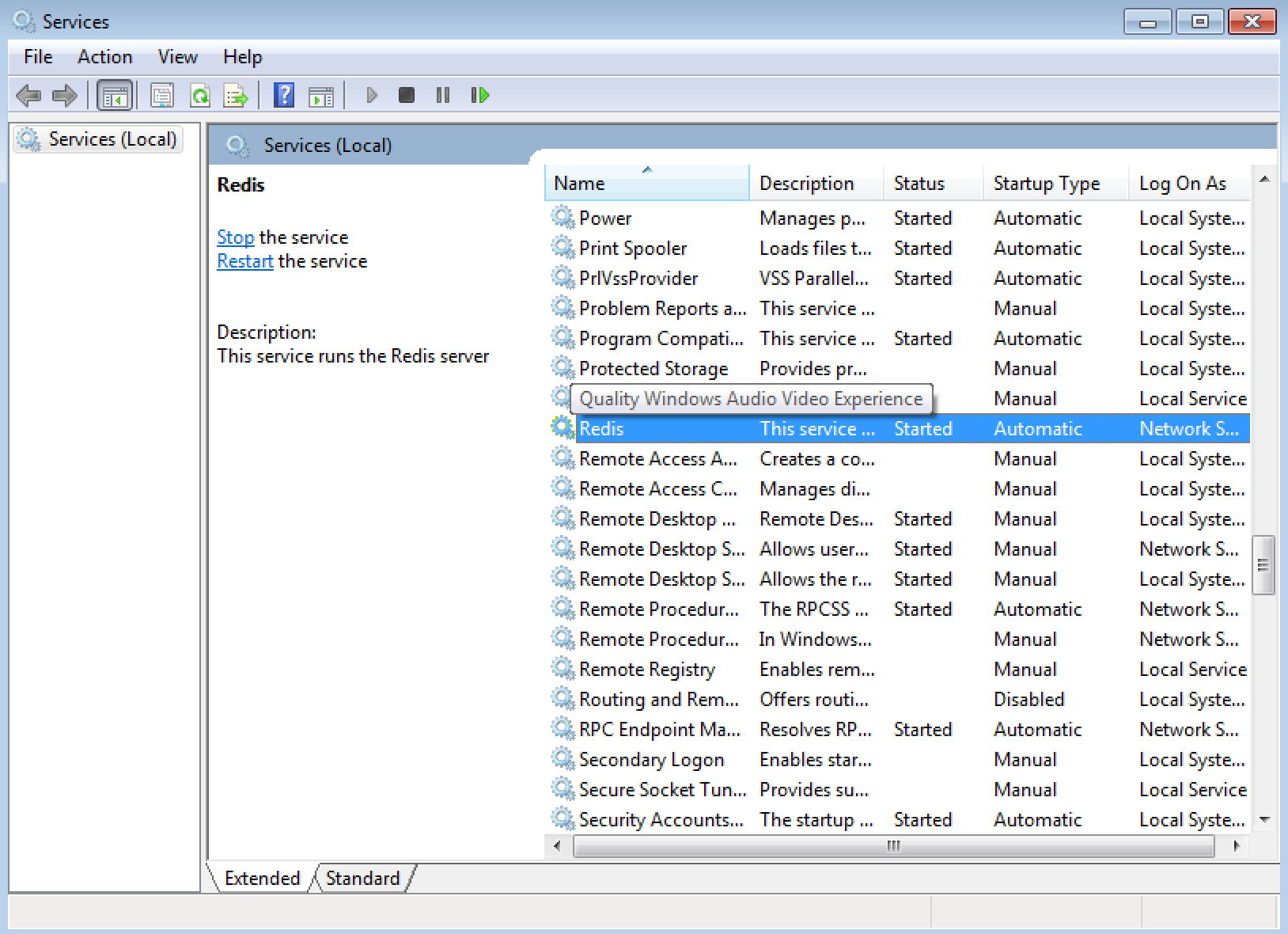
1. In the nosql-lab-redis folder execute: node app.js
2. You can now surf to the result of the app at <http://localhost:3000>
3. Check out what you get, get a feeling for it.
4. **STOP:** After checking the code and the data-schema: How would you proceed to implement a list of the newest 25 posts/questions?
5. **STOP:** Also describe how you would create a detailed view that features the post/question and then all the comments and answers
6. **STOP:** Modify the code so that you show the details of a post/question *including* the metadata – hence showing nicely title, when it was asked and the body. Upload a Screenshot of this as well as the descriptions of the other points

## Notes

As we're using Redis as an example for KeyValue systems, we'll stay with the base commands -- as you can see they are available since eternety (1.0):

* [GET](http://redis.io/commands/get) --> read a key's value
* [SET](http://redis.io/commands/set) --> set a key to a value
* [KEYS](http://redis.io/commands/KEYS) --> get all keys (don't do this on big stores)

## Installation Notes

Redis will be installed as a Windows Service and always running in the background. To stop it from starting automatically (but not uninstalling) you can disable Automatic Startup in Services Management. Launch it by starting services.msc and look for Redis: 

Just change the Startup Type to Manual ☺