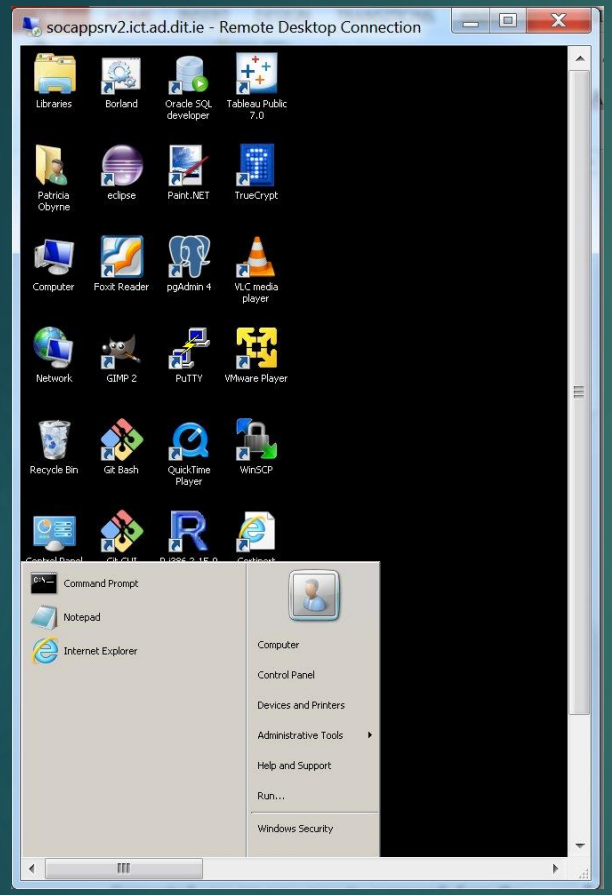
Using MongoDB

Using the MongoDB server

- User accounts have been set up for you on the ICT domain.
- Your username should be the first initial of your first name, followed by your lastname all in lower case.
  - E.g. Alan Turing's would be **aturing.**
- Your password is your student number.
  - E.g. Alan Turing's would be **c3312345**
- Your database name is the same as your username.
  - E.g. Alan Turing's would be **aturing.**

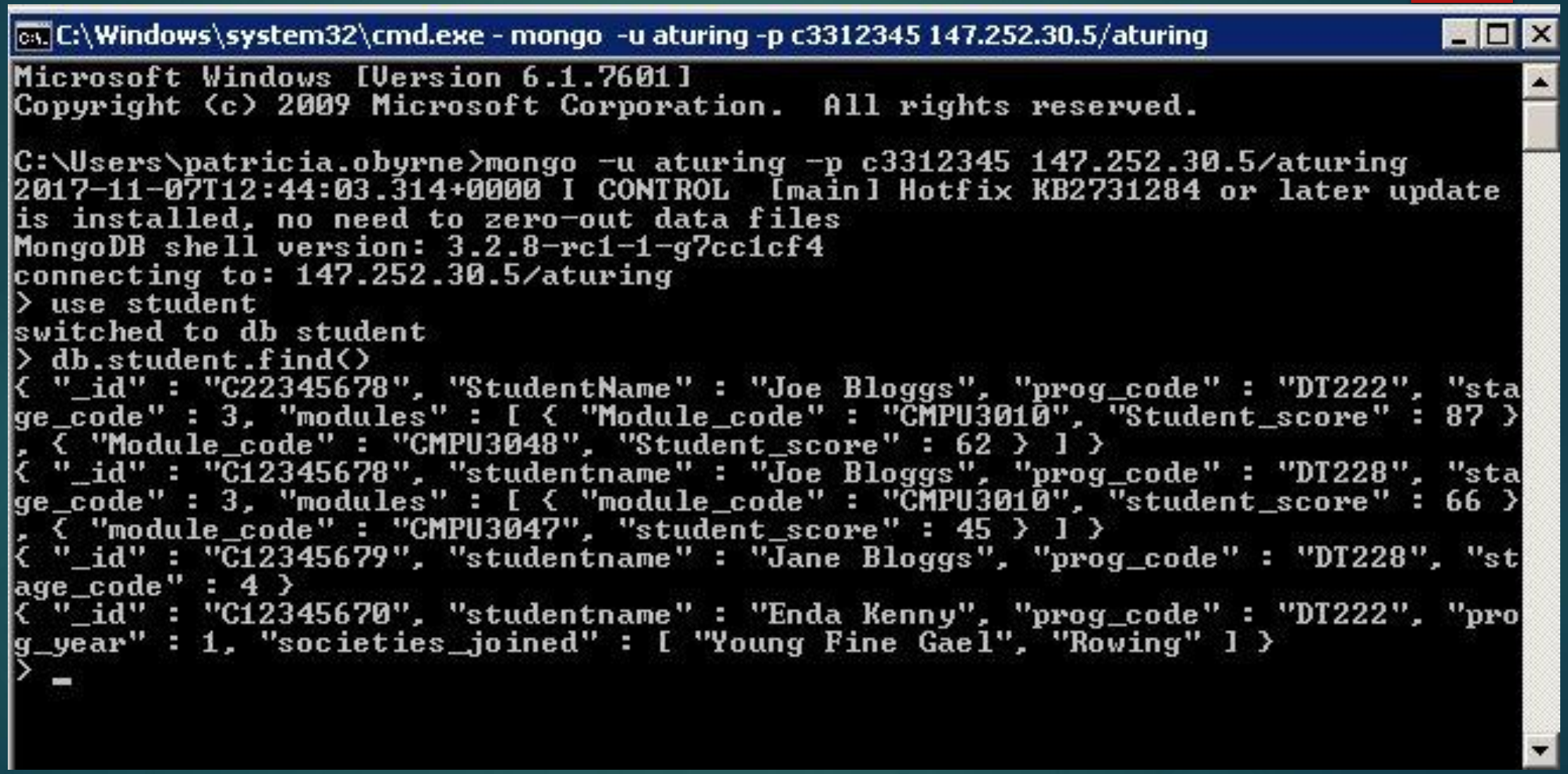
Socappsrv2



- Find the remote desktop icon on the lab computer.
- Open Socappsrv2 and login using your ictdomain username and password.
- Open the command prompt and access:
- D:/program files/mongoDB/bin/
- FROM THE COMMAND PROMPT
  - Login to the running MongoDB server

Accessing MongoDB on socappsrv2

- From the command line on path D:/program files/mongoDB/bin/ enter:
  - mongo –u ***uname***–p ***pwd*** 147.252.30.5/***databasename***
  - Where uname is the username
  - c12345 is the password and
  - dbname is the database you're accessing
  - E.g. Alan Turing's would be
    - mongo –u ***aturing*** –p ***c3312345*** 147.252.30.5/***aturing***
- Your account also has read access to the student database. Everyone can use that for the moment.
- When connect you should see something similar to the following slide.

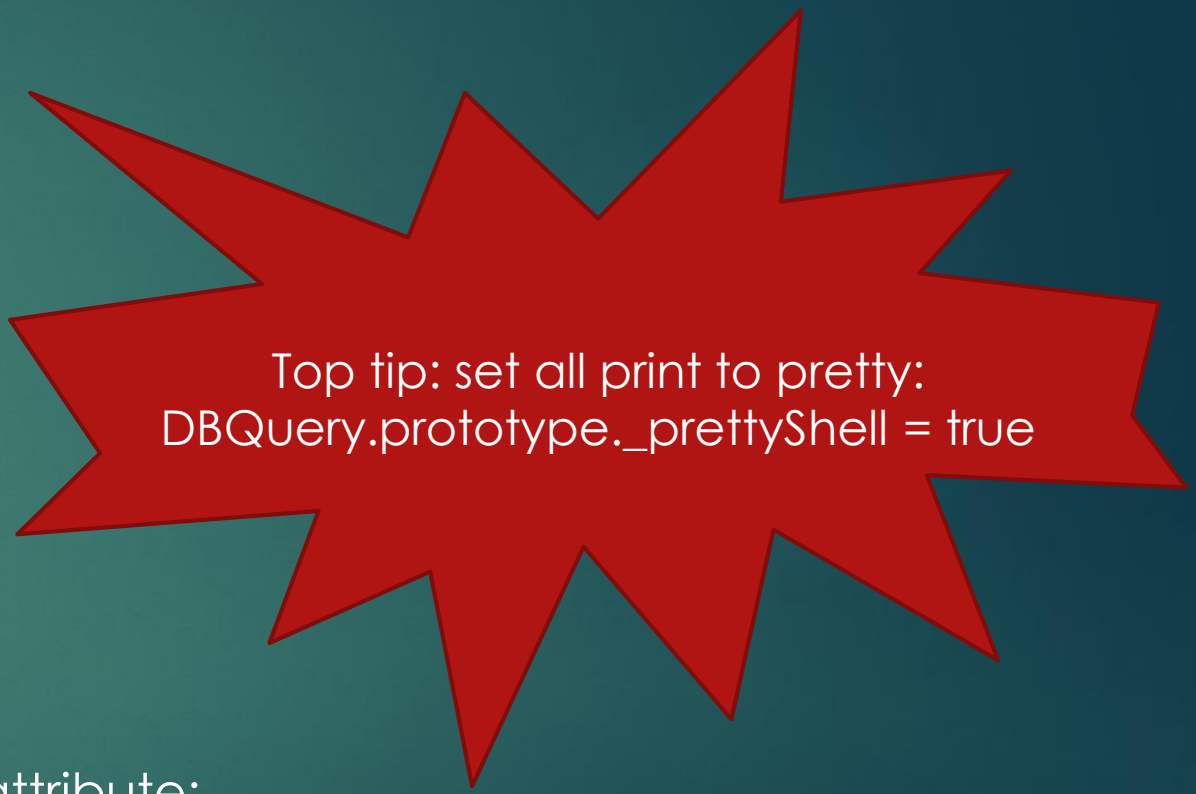

C:\Windows\system32\cmd.exe - mongo -u aturing -p c3312345 147.252.30.5/aturing
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation. All rights reserved.
C:\Users\patricia.obyrne>mongo -u aturing -p c3312345 147.252.30.5/aturing
2017-11-07T12:44:03.314+0000 I CONTROL [main] Hotfix KB2731284 or later update is installed, no need to zero-out data files
MongoDB shell version: 3.2.8-rc1-1-g7cc1cf4
connecting to: 147.252.30.5/aturing
> use student
switched to db student
> db.student.find()
{ "_id" : "C22345678", "StudentName" : "Joe Bloggs", "prog_code" : "DT222", "stage_code" : 3, "modules" : [{ "Module_code" : "CMPU3010", "Student_score" : 87 }, { "Module_code" : "CMPU3048", "Student_score" : 62 }] }
{ "_id" : "C12345678", "studentname" : "Joe Bloggs", "prog_code" : "DT228", "stage_code" : 3, "modules" : [{ "module_code" : "CMPU3010", "student_score" : 66 }, { "module_code" : "CMPU3047", "student_score" : 45 }] }
{ "_id" : "C12345679", "studentname" : "Jane Bloggs", "prog_code" : "DT228", "stage_code" : 4 }
{ "_id" : "C12345670", "studentname" : "Enda Kenny", "prog_code" : "DT222", "prog_year" : 1, "societies_joined" : ["Young Fine Gael", "Rowing"] }
>

Practical example - transcript

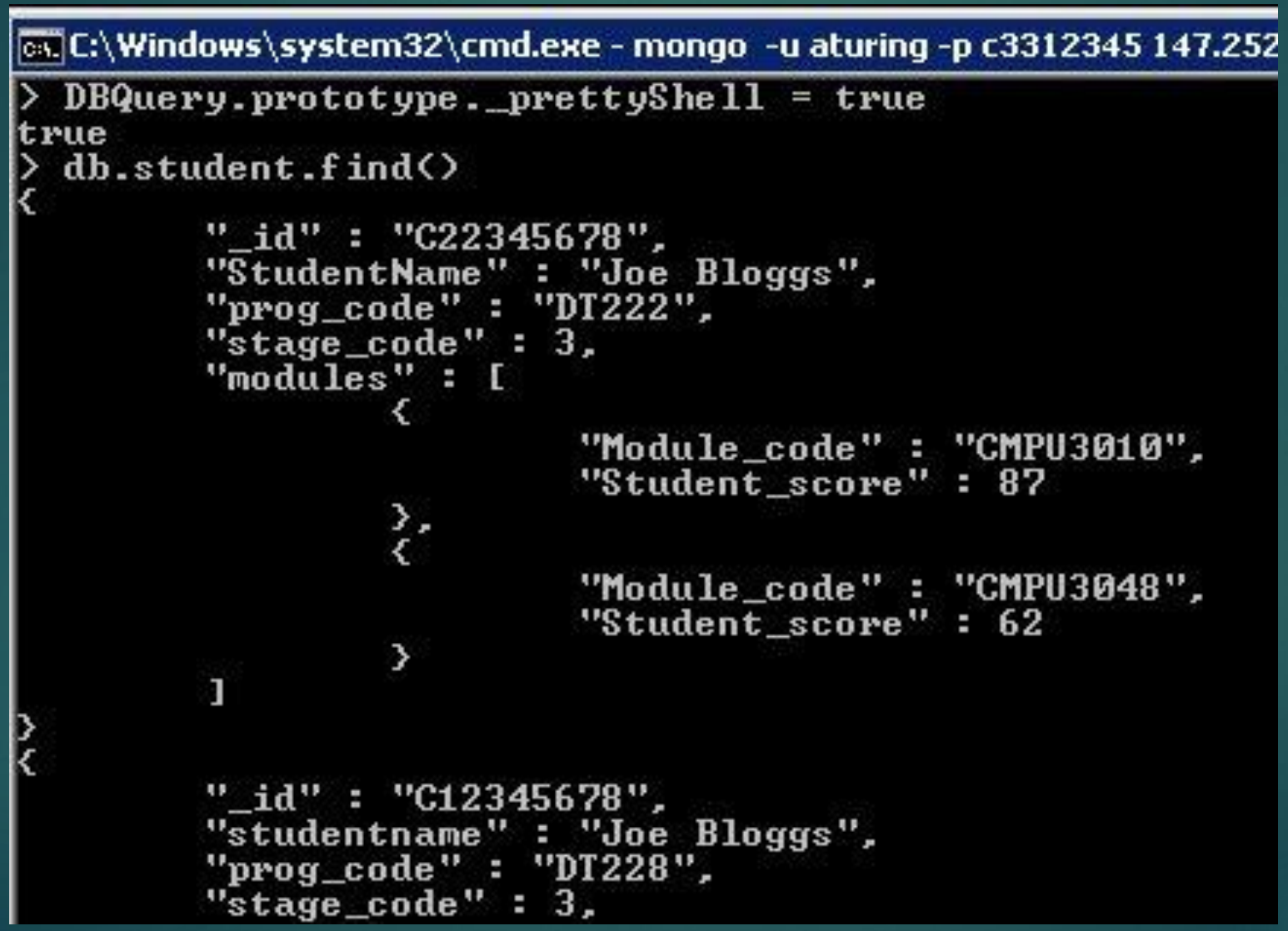
- A database is similar to a schema in Oracle.
- A Collection is similar to a table.
- A document is similar to a row, but it is schema-less.

Adding documents to a collection

- Connect to the database (e.g. student)
  - use student
- See what collections are there:
  - show collections
- List the documents that are already there:
  - db.transcript.find({})
- Formatted listing
  - db.transcript.find().pretty()
- List the documents, filtering on a specific attribute:
  - db.transcript.find**({ "Results.Mark": 83 })**
  - db.transcript.find**({ "Results.Mark": {$gt: 40} })**



With prettyShell



A few more examples

- Selection - only find transcript for Nealon, etc.
  - db.transcript.find({"Surname":"Nealon"})
- Selection from embedded documents
  - db.transcript.find({"Results.Grade":"P"})
  - db.transcript.find({"Results.Mark":40.0})
  - db.transcript.find({"Results.Mark":"EX"})

Selection with comparison operator

- db.transcript.find({"Results.Mark":{$gt:60.0}})

A few more examples

- // Selection with 'or'.
  - db.transcript.find({"ProgDecision":{$in: ["Progress","Pass by Compensation"]}})
- // Checking existence of a field
  - db.transcript.find({"Results.GradeComment":{$exists: false}})
- //Projection - show just the Surname and Progression Decision. Suppress the_id.
  - db.transcript.find({},{"Surname":1,_id:0,"ProgDecision":1})
- //Sorting the output
  - db.transcript.find({},{"Surname":1,_id:0,"ProgDecision":1}).sort({"Surname":1})
- //Sorting the output descending
  - db.transcript.find({},{"Surname":1,_id:0,"ProgDecision":1}).sort({"Surname":-1})
- //Counting the number of students in each category:
- db.transcript.aggregate**([{$group:{_id:"$ProgDecision","NumberinCategory":{$sum:1}}}])**

Your own copy

- You can download the server from http://www.mongodb.org/downloads
- There's a reasonably comprehensive tutorial on
  - https://www.tutorialspoint.com/mongodb/mongodb_tutorial.pdf
- To start *your own* server run mongod.exe
- The standard terminal client is called mongo.exe but there is a graphical user interface called MongoChef.