**220CT – Data and Information Retrieval Coursework**

**Hannah Rice**

**SID: 6238519**

**Task 1, Part 1:**

First I had to normalise the table:

**First Normal Form:**

**Mission Table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Mission\_No | Agency\_No | Lead\_Agency | Country | Mission\_Date |
| ISS-2237 | 178 | JAXA | Japan | 14/12/2013 |
| ISS-3664 | 526 | ESA | EU | 16/01/2014 |
| ISS-2356 | 167 | NASA | USA | 16/02/2014 |
| ISS-1234 | 032 | Roskosmos | Russia | 16/04/2014 |

**Equipment Table:**

|  |  |  |
| --- | --- | --- |
| Equipment | Quantity | Weight |
| Portable water dispenser | 2 | 100kg |
| Flexible airduct | 6 | 0.5kg |
| Small storageRack | 4 | 2kg |
| Bio filter | 6 | 0.20kg |
| Small storageRack | 3 | 2kg |
| Battery pack | 2 | 5kg |
| Urine transfertubing | 2 | 1.5kg |
| O2 scrubber | 1 | 50kg |
| Small storageRack | 1 | 2kg |
| Flexible airduct | 2 | 0.5kg |

**Second Normal Form:**

**Mission Table:**

|  |  |  |
| --- | --- | --- |
| Mission\_No | Agency\_No(fk) | Mission\_Date |
| ISS-2237 | 178 | 14/12/2013 |
| ISS-3664 | 526 | 16/01/2014 |
| ISS-2356 | 167 | 16/02/2014 |
| ISS-1234 | 032 | 16/04/2014 |

**Agency Table:**

|  |  |  |
| --- | --- | --- |
| Agency\_No | Lead\_Agency | Country |
| 178 | JAXA | Japan |
| 526 | ESA | EU |
| 167 | NASA | USA |
| 032 | Roskosmos | Russia |

**Equipment Table:**

|  |  |
| --- | --- |
| Equipment | Weight |
| Portable water dispenser | 100kg |
| Flexible Airduct | 0.5kg |
| Small storageRack | 2kg |
| Bio filter | 0.20kg |
| Battery Pack | 5kg |
| Urine tansfertubing | 1.5kg |
| O2 scrubber | 50kg |

**Third Normal Form:**

**Mission Table:**

|  |  |  |  |
| --- | --- | --- | --- |
| Mission\_No | Agency\_No(fk) | Mission\_Date | Total\_Weight |
| ISS-2237 | 178 | 14/12/2013 | 211kg |
| ISS-3664 | 526 | 16/01/2014 | 1.20kg |
| ISS-2356 | 167 | 16/02/2014 | 69kg |
| ISS-1234 | 032 | 16/04/2014 | 2.5kg |

**Agency Table:**

|  |  |  |
| --- | --- | --- |
| Agency\_No | Lead\_Agency | Country |
| 178 | JAXA | Japan |
| 526 | ESA | EU |
| 167 | NASA | USA |
| 032 | Roskosmos | Russia |

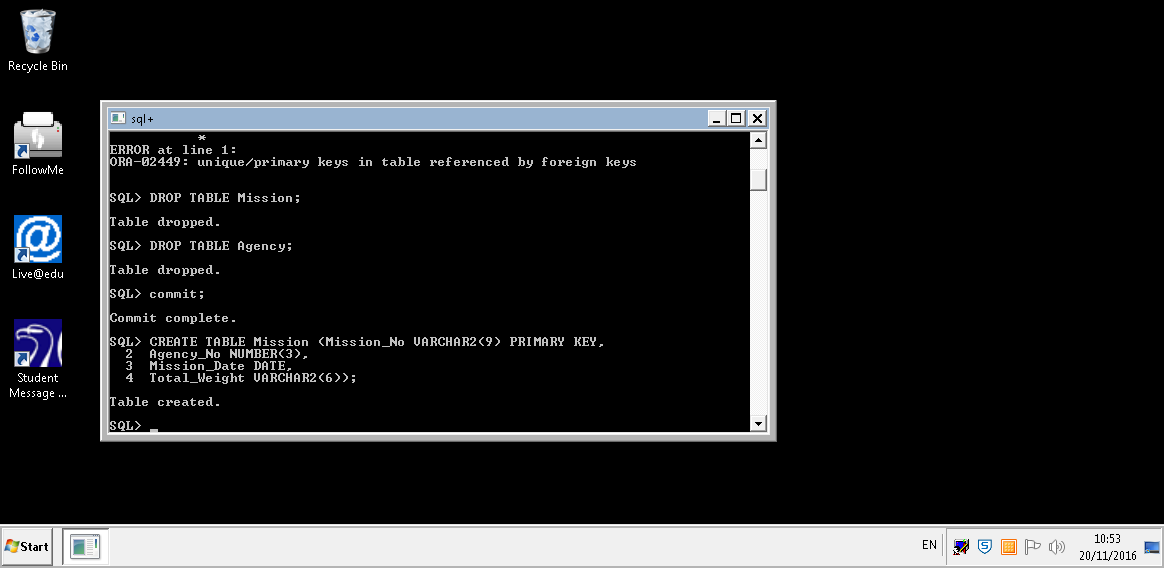
**Equipment Table:**

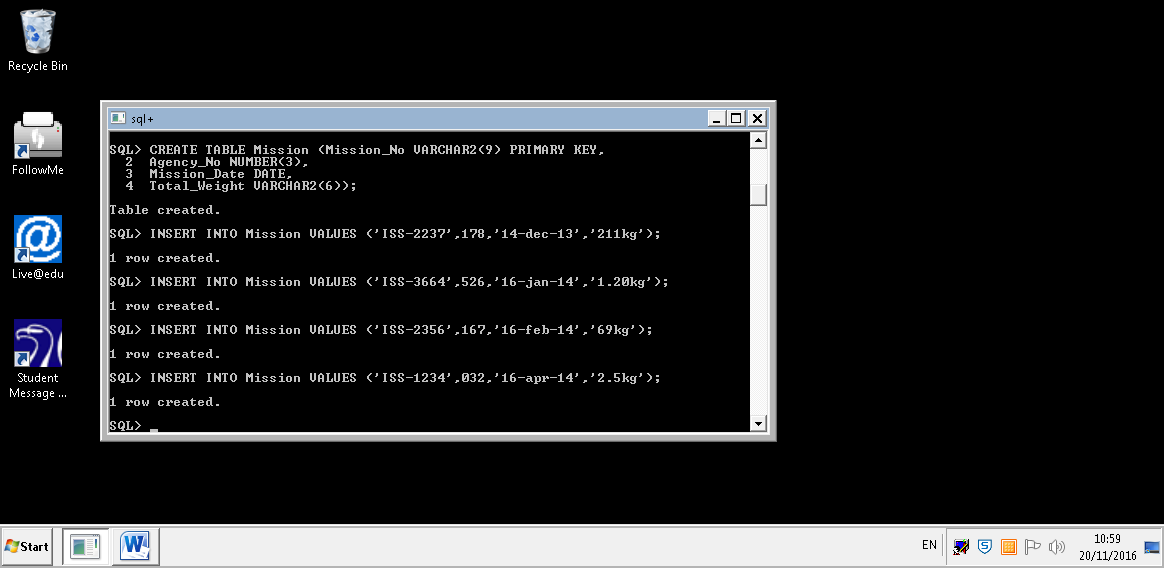
|  |  |  |
| --- | --- | --- |
| Equipment\_ID | Equipment NAME | Weight |
| PWD | Portable water dispenser | 100kg |
| FA | Flexible Airduct | 0.5kg |
| SSR | Small storageRack | 2kg |
| BF | Bio filter | 0.20kg |
| BP | Battery Pack | 5kg |
| UT | Urine tansfertubing | 1.5kg |
| O2S | O2 scrubber | 50kg |

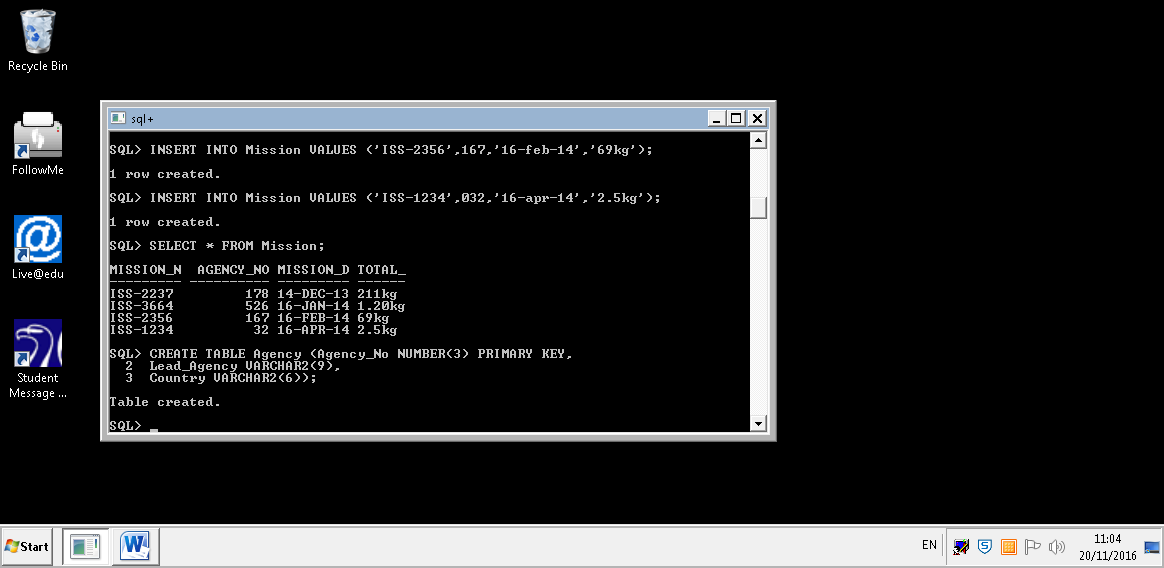
**Mission-Equipment TABLE:**

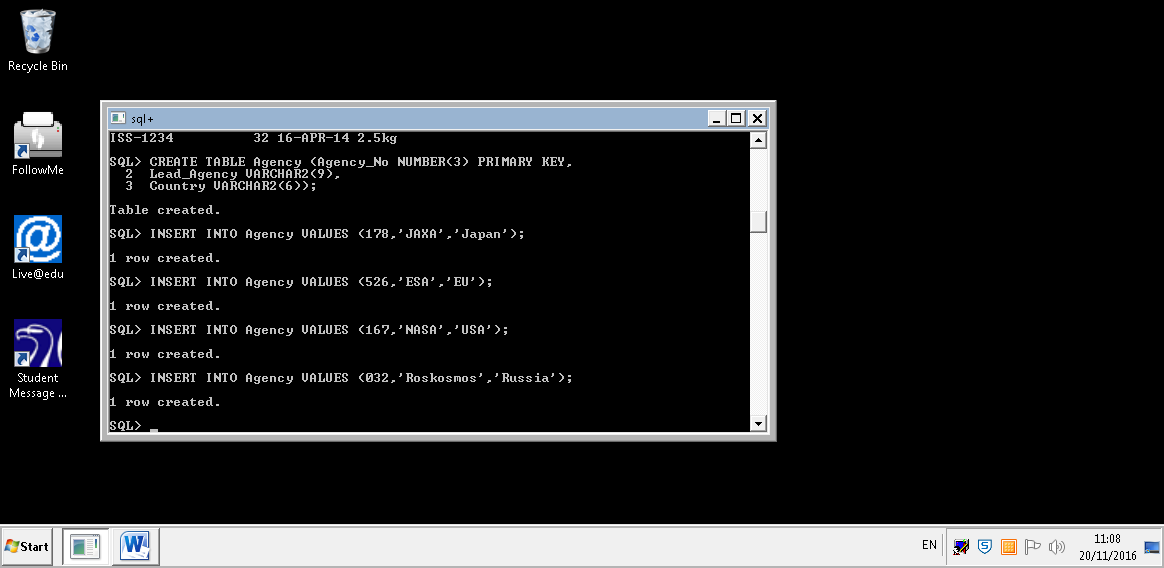
|  |  |  |  |
| --- | --- | --- | --- |
| Mission\_equip\_ID | Equipment\_ID(fk) | Mission\_No(fk) | Quantity |
| ID: 1 | PWD | ISS-2237 | 2 |
| ID: 1.1 | FA | ISS-2237 | 6 |
| ID:1.2 | SSR | ISS-2237 | 4 |
| ID: 2 | BF | ISS-3664 | 6 |
| ID: 3 | SSR | ISS-2356 | 3 |
| ID: 3.1 | BP | ISS-2356 | 2 |
| ID: 3.2 | UT | ISS-2356 | 2 |
| ID: 3.3 | O2S | ISS-2356 | 1 |
| ID: 4 | SSR | ISS-1234 | 1 |
| ID: 4.1 | FA | ISS-1234 | 2 |

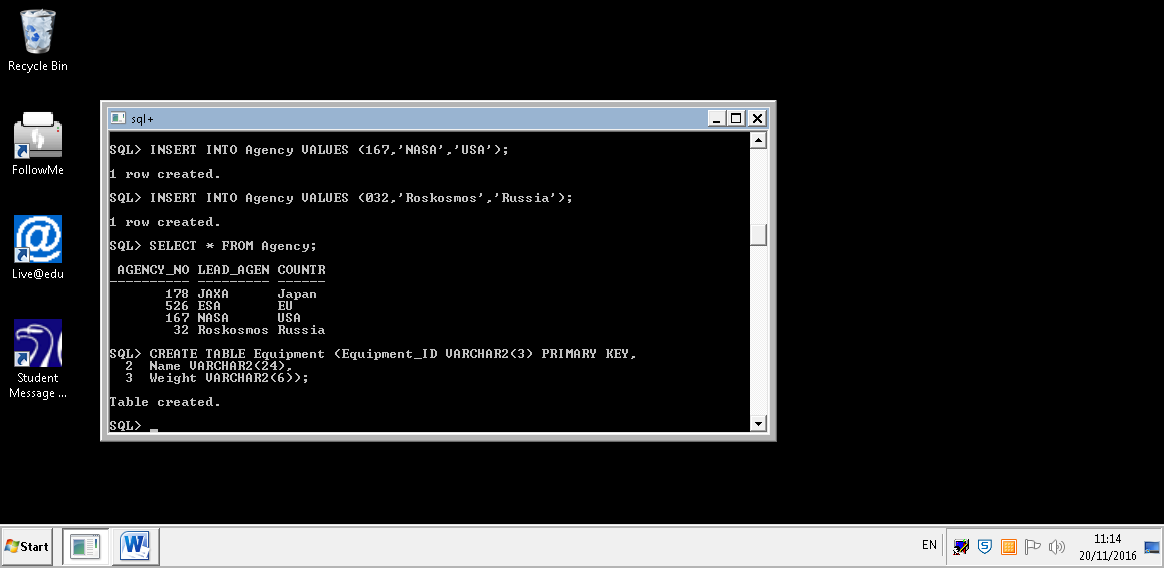
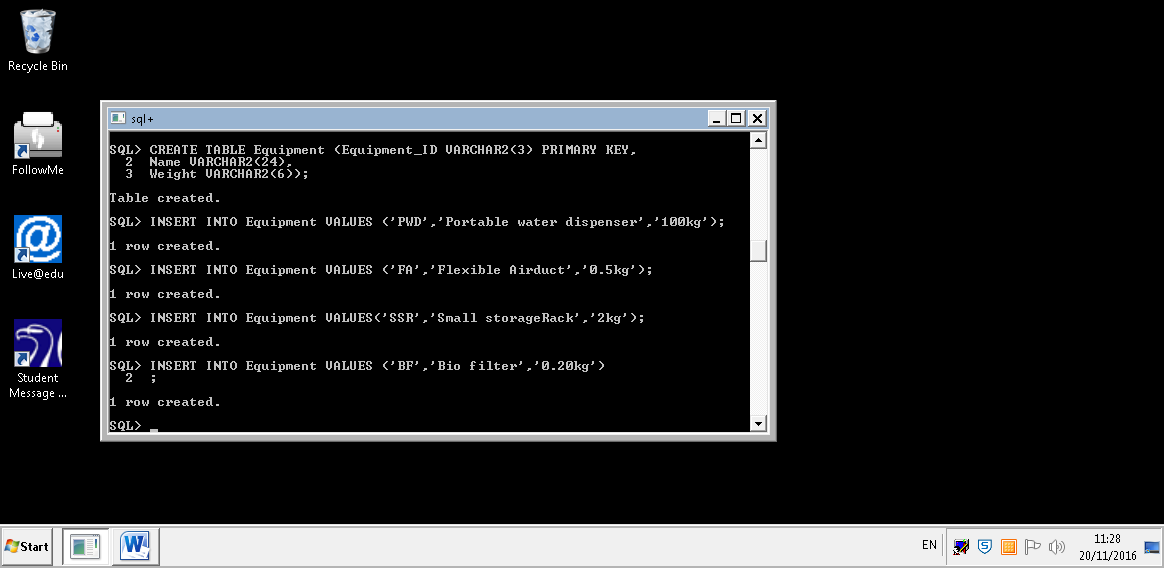
**SQL Commands and Screenshots**

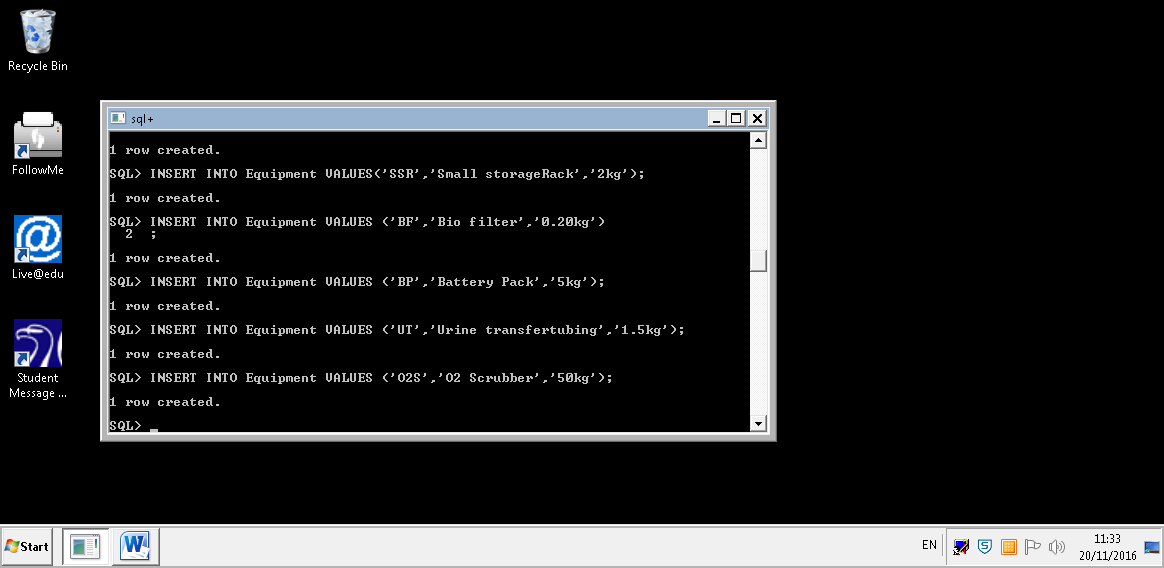
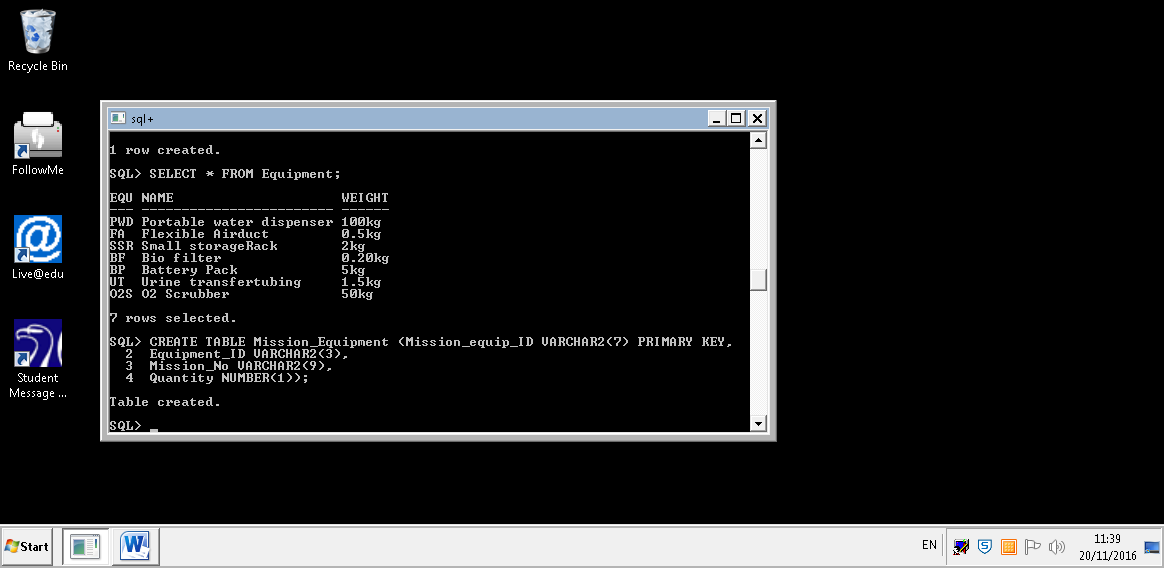


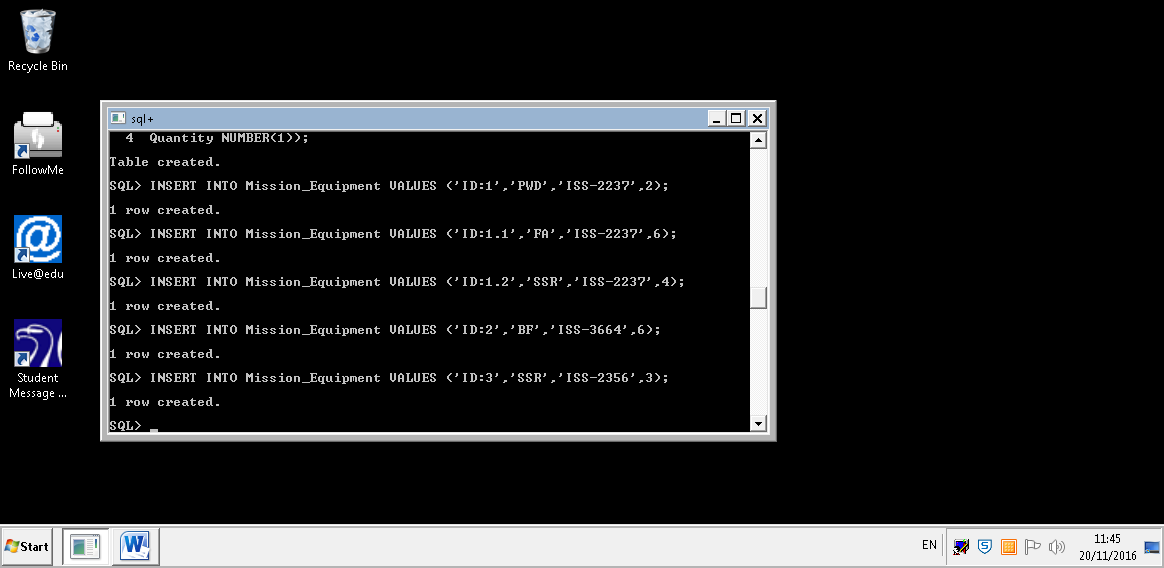
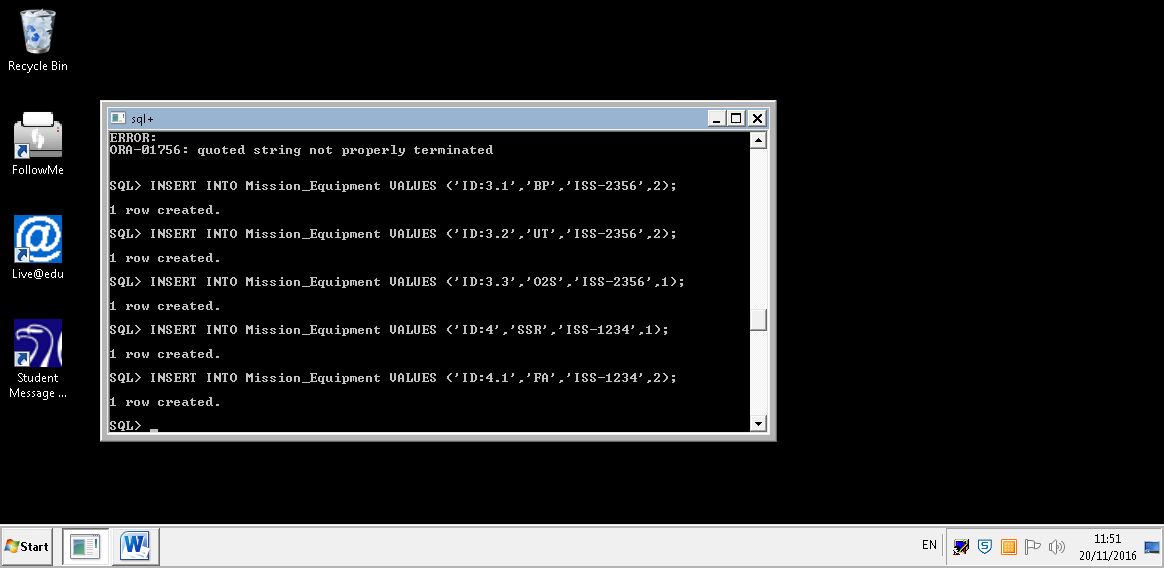
****

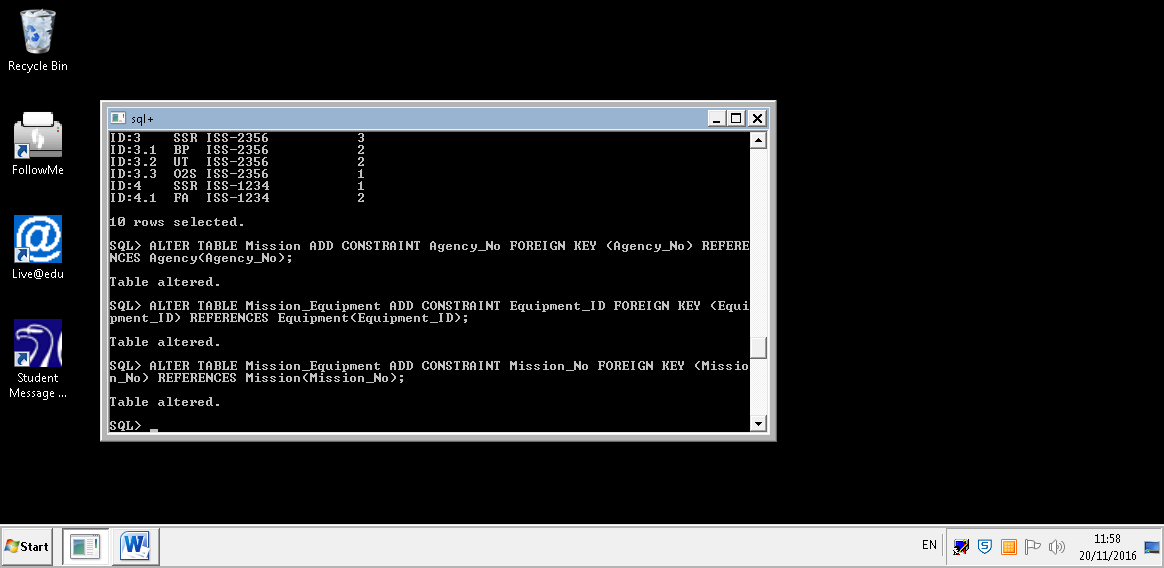
****

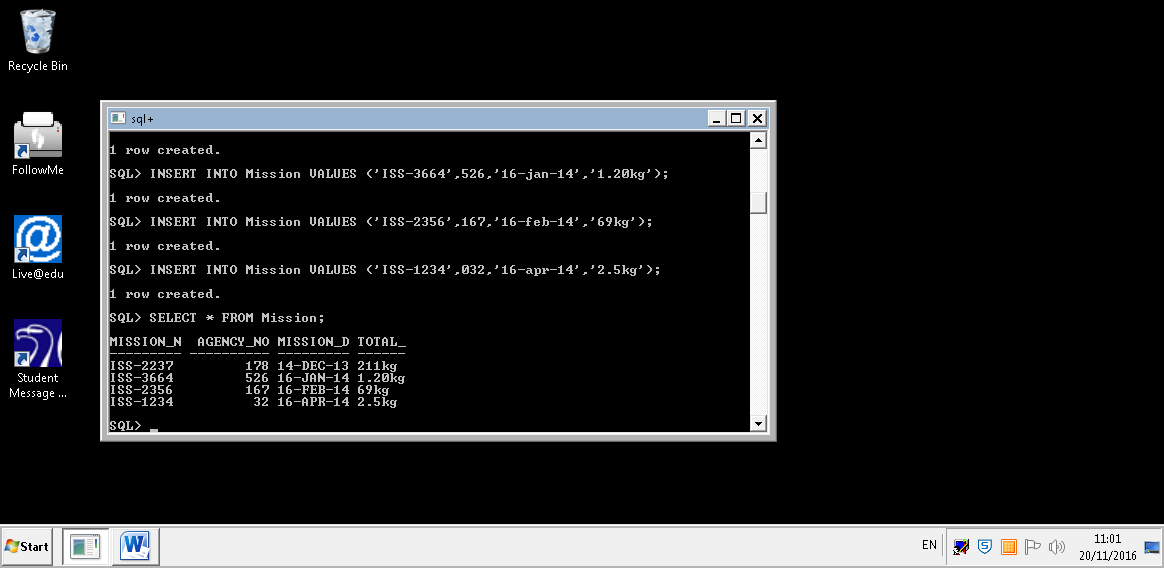
****

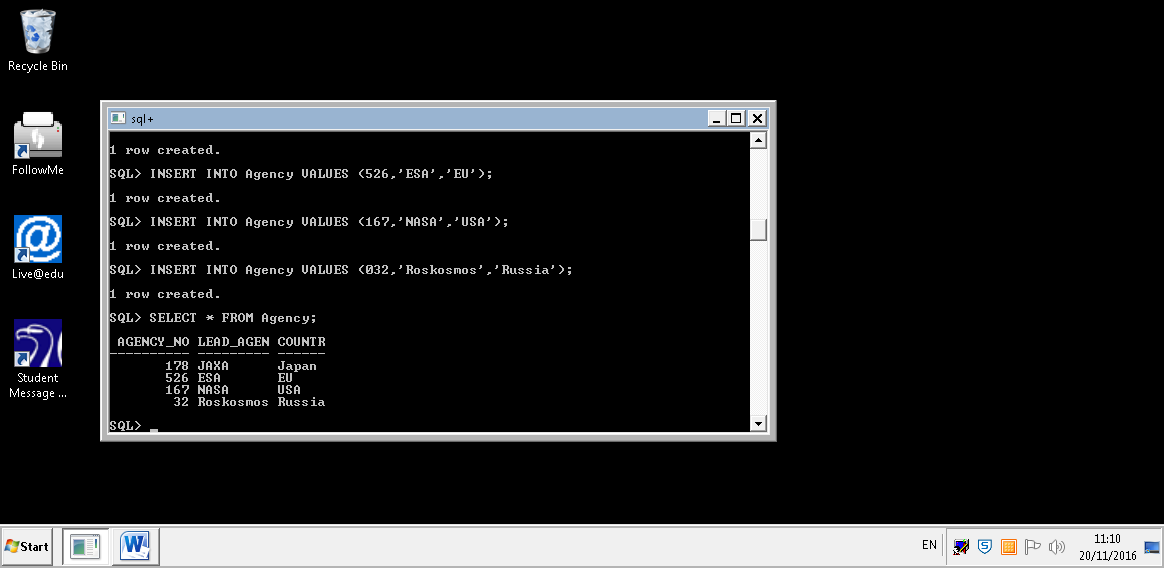
****

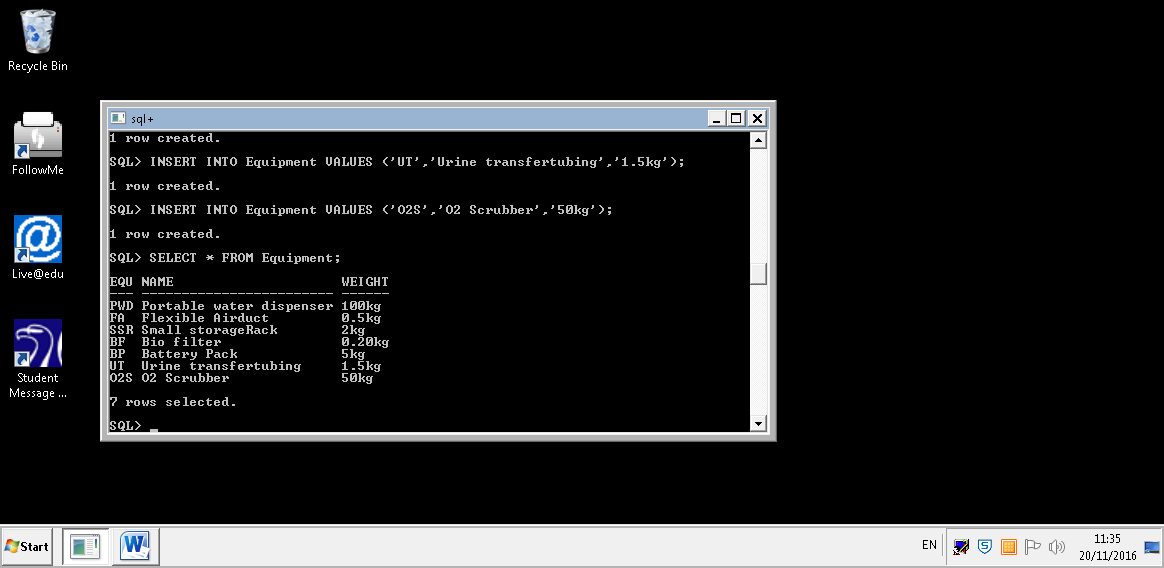
****

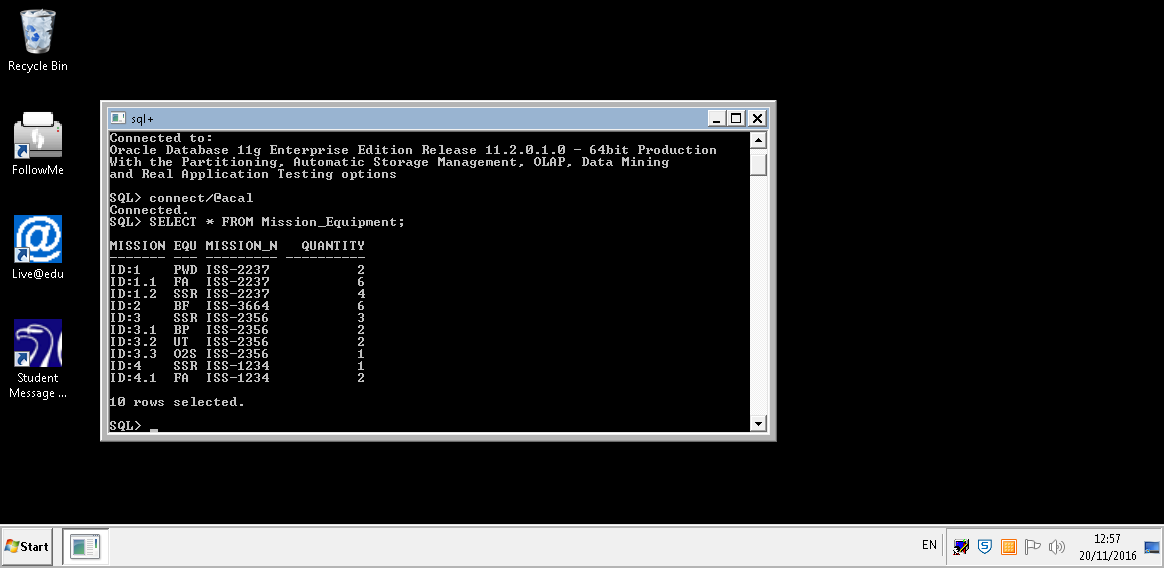
****

****

****

****

****

****

**Entity Relationship Diagram:**

**AGENCY**

Agcy\_No:

178, 526, 167, 032

Lead Agcy:

JAXA, ESA, NASA, Roskosmos

Country:

Japan, EU, USA, Russia

**MISSION**

Mission\_No:

ISS-2237, ISS-3664, ISS-2356, ISS-1234

Mission Date:

14/12/2013, 16/01/2014, 12/02/2014, 16/04/2014

**EQUIPMENT**

Name:

Portable water dispenser, Flexible Airduct, Small storageRack, Bio filter, Battery Pack, Urine transfertubing, O2 Scrubber

Weight:

100kg, 0.5kg, 2kg, 0.20kg, 2kg, 5kg, 1.5kg, 50kg

**Task 1, Part 2:**

To choose the correct database for the Nasa Exoplanet Archive data set I had to consider; what was the purpose of the dataset, the structure of the database, data retrieval patterns and the performance of the database engine.

Considering all the above, the database solution of my choice is Graph DB. A graph is a collection of nodes with relationships between them, from one edge of a node to another edge of a node, it is also a part of the NO SQL family. A node is an instance of an entity and, a graph is considered a key-value store. There will be a node for each exoplanet and edges that connect pairs of nodes together. For each row in the Nasa exoplanet dataset there will be a node(vertex), and for each column there will be key-value pairs. The properties will be attached in key, values pairs on both nodes and relationships(edges).

**For Example:**

PLANET

Host Name: 11 Com

Properties …

DISCOVERED USING

LETTER OF

DISCOVERY METHOD

Type: Radial Velocity

PLANET LETTER

ID: b

DISCOVERED USING

LETTER OF

PLANET

Host Name: 55 Cnc

Properties …

**Node descriptions would look like:**

Node 1: Planet 1, instance of Entity PLANET

Node 1 properties: HostName = “11 Com”, DefaultAlias = “11 Com b”, No\_Planets\_System = “1”

Node 2: Discovery Method 1, instance of Entity DISCOVERY METHOD

Node 2 properties: Type = “Radial Velocity”

**I would use the query language Cypher and Neoj4 to:**

**Create the Nodes:**

CREATE (n: PLANET LETTER { ID: ‘b’ }) RETURN n

CREATE (n: PLANET { Host\_Name: ’11 Com’ }) RETURN n

CREATE (n: PLANET {Host\_Name: ’55 Cnc’ }) RETURN n

CREATE (n: DISCOVERY METHOD { Type: ‘Radial Velocity’ }) RETURN n

**Create the Relationships:**

MATCH (a: PLANET LETTER { ID: ‘b’ }), (b:PLANET { Host\_Name: ’11 Com’ }) CREATE (a) – [:LETTER\_OF] -> (b)

MATCH (a: PLANET LETTER {ID: ‘b’}), (b:PLANET { Host\_Name: ’11 Com’ }) MERGE (a) – [:LETTER\_OF] -> (b)

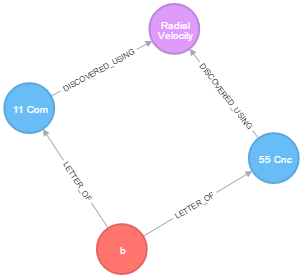
MATCH (a: PLANET LETTER { ID: ‘b’ }), (b:PLANET { Host\_Name: ’55 Cnc’ }) CREATE (a) –[:LETTER\_OF] -> (b)

MATCH (a: PLANET { Host\_Name: ’11 Com’ }), (b: DISCOVERY METHOD { Type: ‘Radial Velocity’ }) CREATE (a) –[:DISCOVERED\_USING] -> (b)

MATCH (a: PLANET { Host\_Name: ’55 Cnc’ }), (b: DISCOVERY METHOD { Type: ‘Radial Velocity’ }) CREATE (a) – [:DOSCOVERED\_USING] ->(b)

**Display the Graph:**

MATCH (n) RETURN n LIMIT 25



Due to the Nasa exoplanet archive dataset being so large the database solution needs to be adaptable and dynamic. I chose to use Graph DB because, it is expressive and unstructured making it perfect for such a large dataset as the Nasa exoplanet data set. You can Create, Read, Update, and Delete for the database allowing it to be flexible and very adaptable for the use of relationships. Graph DB is unstructured meaning that it is also very expandable and scalable, making it perfect to suit the needs and purpose of this data set because whenever new information or planets are found the database is unstructured for the addition of the new information. Nasa will also have no idea how large their data set could potentially grow to be over time, which is also why I chose Graph DB. Due to the DB being so unstructured relationships are very important to provide context for nodes and structure.

The properties of Graph DB are:

* Agility: e.g. allowing for more complex relationships between nodes.
* Flexibility: e.g. can add more relationships, nodes, labels, and types.
* Performance: e.g. we can access data very quickly and using traversal methods.

However, with all database solutions Graph Db has some disadvantages that need to be considered. The main disadvantage is that any user that needs to access the database whether to add new data or simply view existing data must have a knowledge of Cypher. This restricts the number of users that could use the database and would require a business to provide training.

Overall I chose Graph DB because it’s flexible, scalable, and expressive. We can customize the database to meet our needs, and its schema free. The use of Cypher is a restriction but once training has been given, using cypher allows the database to be very accessible and easy to query.

**Task 2**