Exploiting Virtual Observatory and Information Technology: Techniques for Astronomy

Lecture #4: Mining the Sloan Digital Sky Survey

- Standard Query Language(SQL)
- Data mining with the SDSS DR3
- Open sky query and Sloan Web Services (not being covered)

Richard McMahon
Institute of Astronomy

Sloan Digital Sky Survey Survey Goals

- Make definitive map of local universe
- Image π steradians(10,000deg^2) of high galactic latitutude sky(|b|>30), in five colours *ugriz*
  - *whole sky is 40,000deg^2*
  - Measure 50 million galaxy images to r~22
- Obtain spectra for 1 million galaxies and 100,000 quasars
- Survey currently funded until June 2005:
  - expect to have completed ~ 80% of above goals by then(8,000deg^2)
- Galactic survey planned; IOA is a consortium member for this.

Status (Nov 2004)

(from Jon Loveday, Sussex)

- Current survey ends June 2005
- Imaging: 8216 out of 8452 sq deg (97%)
- Spectroscopy: 1060 out of 1688 plates (62%) including spectra of
  - 534,255 galaxies
  - 69,500 QSOs
  - 150,000 stars

Data Release 3(DR3)
www.sdss.org/dr3/

- Published(released to public) October 2004
- Imaging area 5282 deg^2
- Photometry of 141 million unique objects
- Spectra of 528,640 objects over 4188 deg^2:

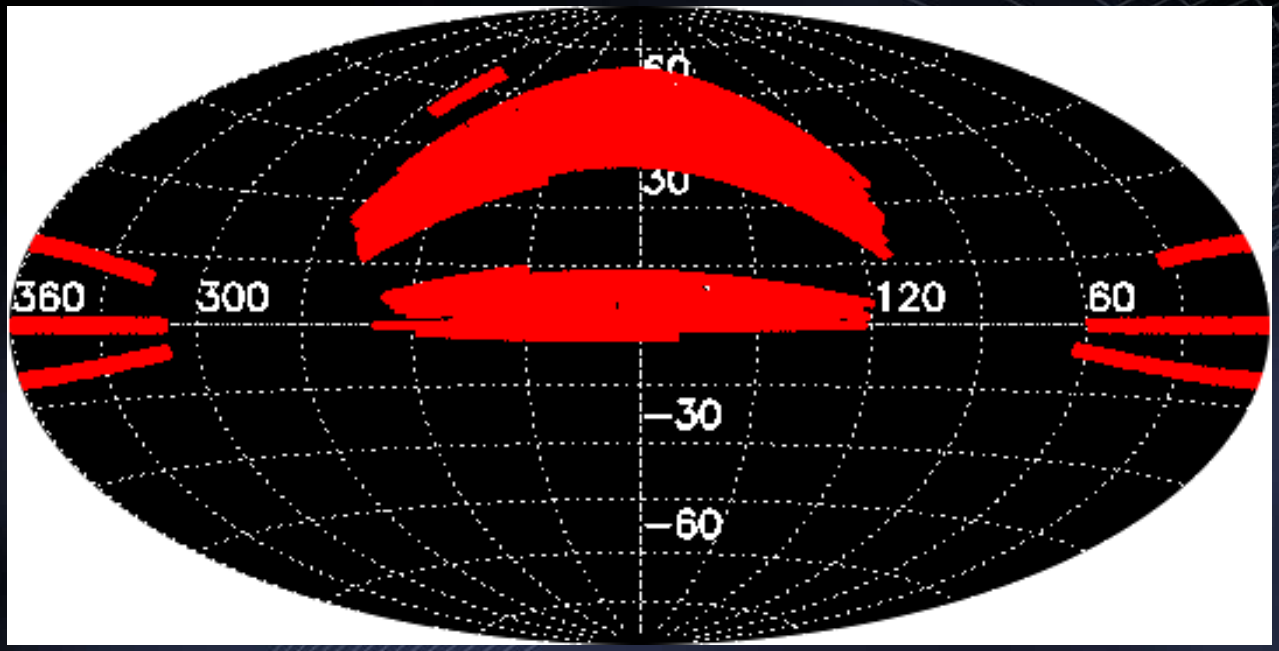
  374,767 galaxies
  51,027 quasars
  102,846 stars/other

- DR4 expected in July 2005
  - Imaging: 6670deg^2 ; Spectra
- DR5(final) July 2006
  - Imaging: 7700deg^2; Spectra: 7000deg^2

DR3 Data Volume

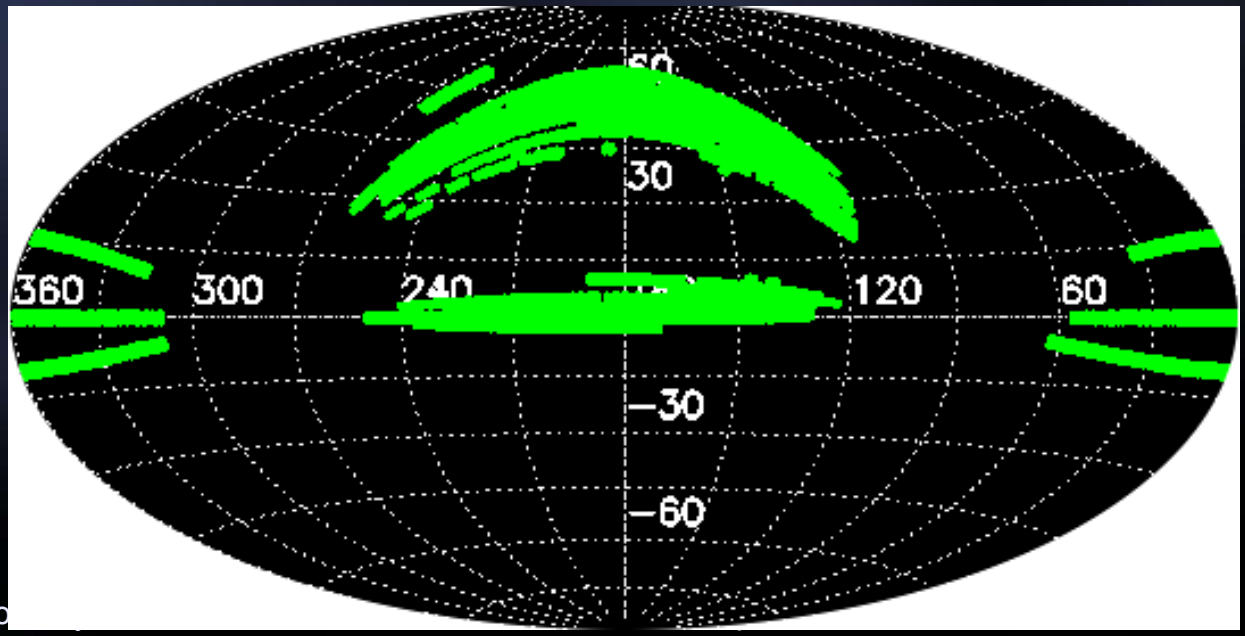
- Images 6.0 TB
- Catalogs (FITs format)1.2 TB
- Catalogs (SQL database) 2.3 TB

- Calibrated 2D spectra as images 41 GB
- 1D spectra with derived parameters; redshifts, line measurements ("1d") 110 GB
  - Copy on disk at IOA



Imaging

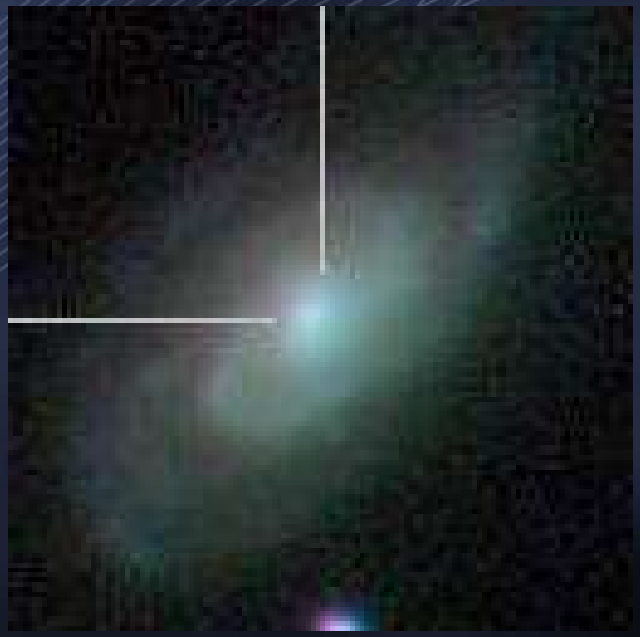
Spectroscopy



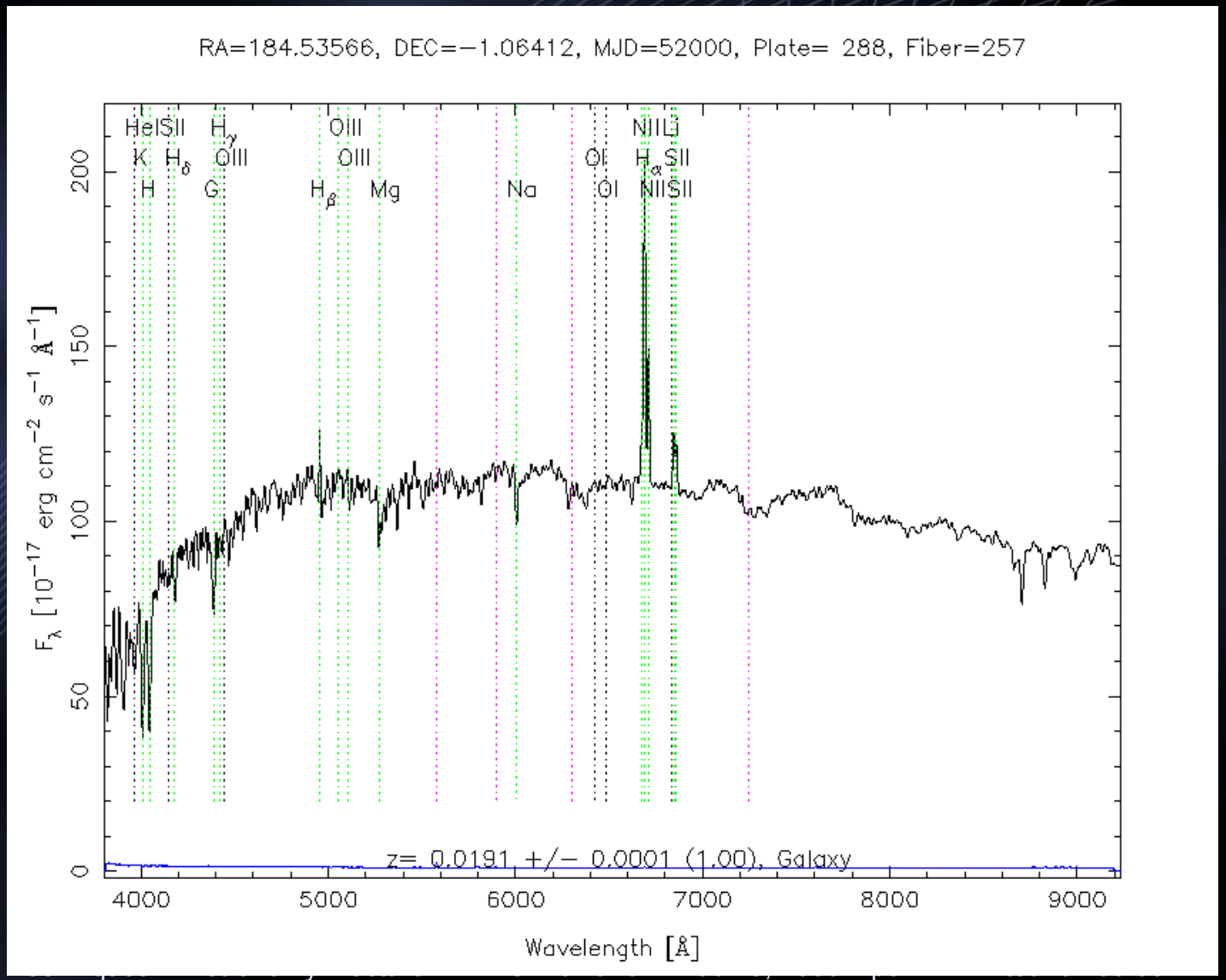
Data Products

- Object lists (position, magnitudes, …)
- Images (FITS files or jpeg images for supplied coordinates)
- Calibrated Spectra (FITS files or gif images)
- Finding charts

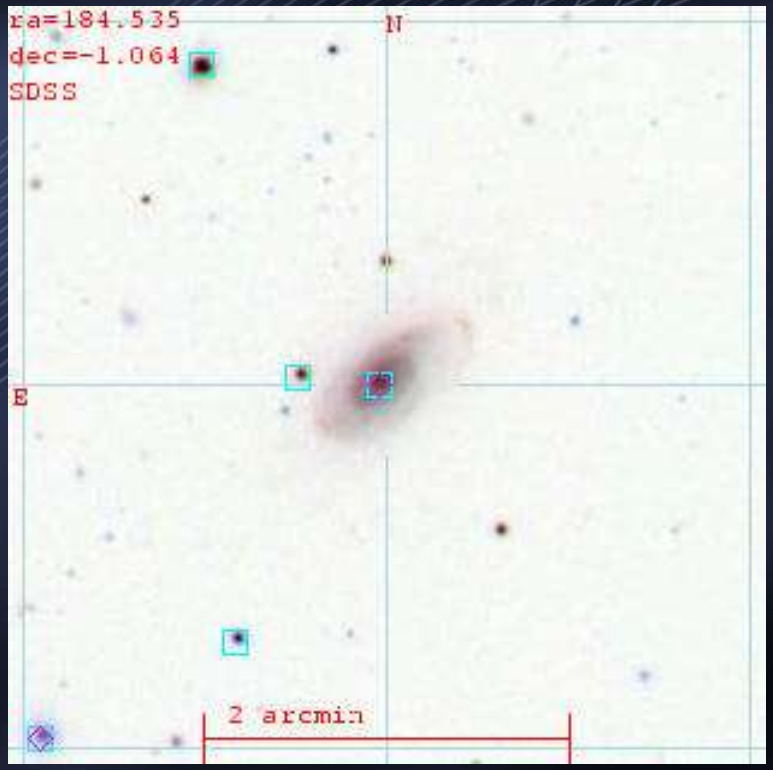
Atlas Image



Spectrum


RA=184.53566, DEC=−1.06412, MJD=52000, Plate= 288, Fiber=257
HeI
SII
K
H
H_δ
G
H_γ
OIII
H_β
Mg
Na
OI
NII
H_α
z= 0.0191 +/− 0.0001 (1.00), Galaxy
F_λ [10^{-17} erg cm^{-2} s^{-1} Å^{-1}]
Wavelength [Å]
0
50
100
150
200
4000
5000
6000
7000
8000
9000

Finding Chart



Data Access
SkyServer http://cas.sdss.org

- Download 5-colour image or spectrum of an object given its:
  - SDSS ID e.g. 587726015611928832
  - Name (eg. NGC4747)
  - Coordinates (RA, Dec)

- "Browse" the sky

- Query the object catalogue

- Key Documents are
  - Glossary
  - Schema Browser
  - Early Data Release Data Description Paper(Stoughton etal, 2002)

SQL queries

- Queries take the form

  SELECT parameters
  FROM table
  WHERE condition

Tables

- Because SDSS measures so many parameters (photometric, astrometric, spectroscopic) they are split up into *tables* or *views* (subsets of tables)
- For help, select Schema Browser from the SkyServer Help menu
- The SpecPhoto view contains most useful photometric and spectroscopic parameters

Quasar Example Query

- SELECT top 1000
  objid, modelmag_u, modelmag_g, modelmag_r,
  modelmag_i, modelmag_z, z
  FROM SpecPhoto
  WHERE specclass=3 and zconf > .95

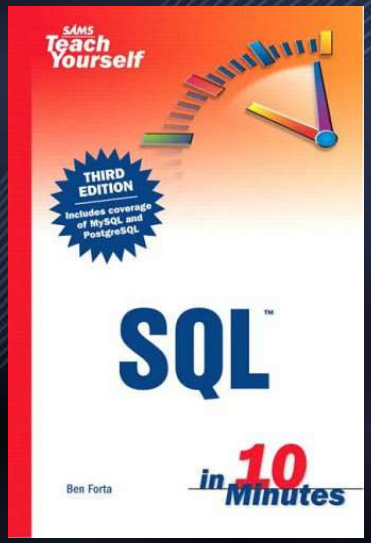
Magnitudes

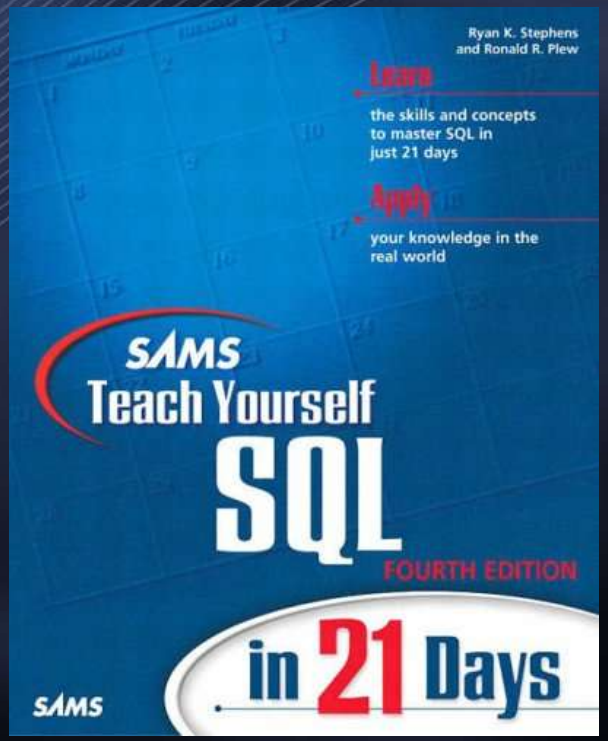
- For stars and QSOs use Point Spread Function(PSF) magnitudes
- For bright galaxies (eg. spectroscopic samples) Petrosian mags probably best
- For fainter galaxies, model magnitudes recommended (combined de Vauc + exp profiles)
- I currently recommend model magnitudes if your are unsure.

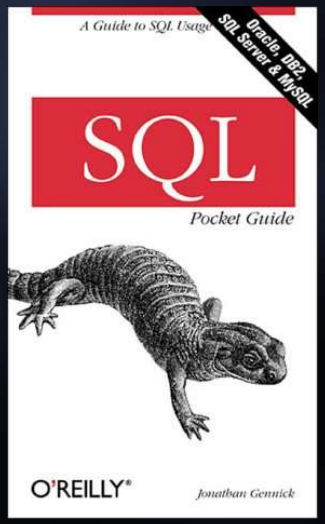
Accessing SDSS Data

- The old fashioned way via a Web Page and http and/or ftp. Try also wget.
  - www.sdss.org
  - www.sdss.org/dr3/access/
- The modern way
  - http://skyserver.sdss.org/
    - casjobs.sdss.org
- In the future using Web Services
  - e.g. skyquery.net as a sampler

SQL
Structured **Q**uery **L**anguage







SQL is a Standard - BUT....

SQL is an ANSI (American National Standards Institute) standard computer language for accessing and manipulating database systems. SQL statements are used to retrieve and update data in a database. SQL works with database programs like:
Microsoft Access, IBM DB2, Informix, Microsoft SQL Server, Oracle, Sybase, Postgresql, MySql

Unfortunately, there are many different versions of the SQL language, but to be in compliance with the ANSI standard, they must support the same major keywords in a similar manner (such as SELECT, UPDATE, DELETE, INSERT, WHERE, and others).

Note: Most of the SQL database programs also have their own proprietary extensions in addition to the SQL standard!

SQL Database Tables

A database most often contains one or more tables. Each table is identified by a name (e.g. "Customers" or "Orders"). Tables contain records (rows) with data.
Below is an example of a table called "Persons":

LastName	FirstName	Address	City
Hansen	Ola	Timoteivn 10	Sandnes
Svendson	Tove	Borgvn 23	Sandnes
Pettersen	Kari	Storgt 20	Stavanger

SQL Queries

With SQL, we can query a database and have a result set returned.
A query like this:

```
SELECT LastName FROM Persons
```

Gives a result set like this:

LastName
Hansen
Svendson
Pettersen

SQL Data Manipulation Language (DML)

•SQL (Structured Query Language) is a syntax for executing queries. But the SQL language also includes a syntax to update, insert, and delete records.
•These query and update commands together form the Data Manipulation Language (DML) part of SQL:
•

- **SELECT** - extracts data from a database table
- **UPDATE** - updates data in a database table
- **DELETE** - deletes data from a database table
- **INSERT INTO** - inserts new data into a database table

SQL Data Definition Language (DDL)

- The Data Definition Language (DDL) part of SQL permits database tables to be created or deleted. We can also define indexes (keys), specify links between tables, and impose constraints between database tables.

- The most important DDL statements in SQL are:

- **CREATE TABLE** - creates a new database table
- **ALTER TABLE** - alters (changes) a database table
- **DROP TABLE** - deletes a database table
- **CREATE INDEX** - creates an index (search key)
- **DROP INDEX** - deletes an index

SQL The SELECT Statement

The SELECT statement is used to select data from a table. The tabular result is stored in a result table (called the result-set).

Syntax

SELECT column_name(s)
FROM table_name

To select the columns named "LastName" and "FirstName", use a SELECT statement like this:

SELECT LastName, FirstName FROM Persons

Persons

LastName	FirstName	Address	City
Hansen	Ola	Timoteivn 10	Sandnes
Svendson	Tove	Borgvn 23	Sandnes
Pettersen	Kari	Storgt 20	Stavanger

LastName	FirstName
Hansen	Ola
Svendson	Tove
Pettersen	Kari

Select All Columns

To select all columns from the "Persons" table, use a * symbol instead of column names, like this:

SELECT * FROM Persons

LastName	FirstName	Address	City
Hansen	Ola	Timoteivn 10	Sandnes
Svendson	Tove	Borgvn 23	Sandnes
Pettersen	Kari	Storgt 20	Stavanger

Semicolon after SQL Statements?

Semicolon is the standard way to separate each SQL statement in database systems that allow more than one SQL statement to be executed in the same call to the server.

Some SQL tutorials end each SQL statement with a semicolon. Is this necessary? We are using MS Access and SQL Server 2000 and we do not have to put a semicolon after each SQL statement, but some database programs force you to use it.

Using the DISTINCT keyword

To select ALL values from the column named "Company" we use a SELECT statement like this:

SELECT Company FROM Orders

Orders

Company	OrderNumber
Sega	3412
W3Schools	2312
Trio	4678
W3Schools	6798

Company
Sega
W3Schools
Trio
W3Schools

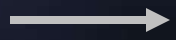
Note that "W3Schools" is listed twice in the result-set.

To select only DIFFERENT values from the column named "Company" we use a SELECT DISTINCT statement like this:

SELECT **DISTINCT** Company FROM Orders

Orders

Company	OrderNumber
Sega	3412
W3Schools	2312
Trio	4678
W3Schools	6798



Company
Sega
W3Schools
Trio

Select All Columns

The WHERE clause is used to specify a selection criterion.

The WHERE Clause

To conditionally select data from a table, a WHERE clause can be added to the SELECT statement.

Syntax

SELECT column FROM table

WHERE column operator value

With the WHERE clause, the following operators can be used:

Operator	Description
=	Equal
<>	Not equal
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
BETWEEN	Between an inclusive range
LIKE	Search for a pattern

Note: In some versions of SQL the <> operator may be written as !=

Using the WHERE Clause

To select only the persons living in the city "Sandnes", we add a WHERE clause to the SELECT statement:

```
SELECT * FROM Persons
WHERE City='Sandnes'
```

LastName	FirstName	Address	City	Year
Hansen	Ola	Timoteivn 10	Sandnes	1951
Svendson	Tove	Borgvn 23	Sandnes	1978
Svendson	Stale	Kaivn 18	Sandnes	1980
Pettersen	Kari	Storgt 20	Stavanger	1960

LastName	FirstName	Address	City	Year
Hansen	Ola	Timoteivn 10	Sandnes	1951
Svendson	Tove	Borgvn 23	Sandnes	1978
Svendson	Stale	Kaivn 18	Sandnes	1980

Using Quotes

Note that we have used single quotes around the conditional values in the examples.

SQL uses single quotes around text values (most database systems will also accept double quotes). Numeric values should not be enclosed in quotes.

For text values:

This is correct:

```
SELECT * FROM Persons WHERE FirstName='Tove'
```

This is wrong:

```
SELECT * FROM Persons WHERE FirstName=Tove
```

Example Skyquery

Transatlantic access to two databases simultaneously using skyquery.net

```
SELECT
  s.objId, s.g, s.i,
  i.objId, i.g, i.i
FROM SDSS:PhotoPrimary s, INTWFS:obj i
WHERE XMATCH(s,i)<1.0
AND AREA(355.5,0.0,2.0)
```


Retrieve data from the Virtual Observatory - Mozilla
File Edit View Go Bookmarks Tools Window Help
http://www.euro-vo.org/astrogrid-portal/main/mount/datacenter/variablesFromMB.html?ac
Search
Retrieve data from the Virtual ...
Home MySpace Resources Queries Workflows Jobs Help Logout
Retrieve data from the Virtual Observatory
Data Query Builder (in (s)ADQL)
User friendly Table-Query Form
Select * FROM J/MNRAS/351/1290/catalog AS T1 where
t1.recno > 0
5
10
20
30
40
50
Clear
Load from MySpace
Select a Table
Save to MySpace
Execute Query
Examples: Cone Search (ivoa) Cone Search (roe) Example 3 Example 4
ADQL Helpers
from ... top ... (... / ... > ... sin ... asin ... abs ...
as ... table ...) ... = ... >= ... cos ... acos ... ceiling ...
where ... name ... + ... <> ... and ... tan ... atan ... floor ...
select ... alias ... - ... < ... or ... cot ... atan2 ... exp ...
region ... circle ... * ... <= ... not ... log ... log10 ... power ...
square ... sqrt ... min ... avg ... max ... sigma ... sum ...
order ... orderby ... direction ... asc ... desc ... distinct ...
pi ... degrees ... radians ... xmatch ... like ... notlike ...
Table:
J/MNRAS/351/1
FROM: J/MNRAS/351/1290/
AS: T1
Click & Paste
recno ELAIS
RAJ2000 DEJ2000
S20cm e_S20cm
S175um e_S175u
S/N175 Off175
S90um e_S90um
S/N90 Off90
S15um S/N15
S6.7um e_S6.7un
q_S6.7un Flag1
Flag2 Jmag
e_Jmag Hmag
e_Hmag Kmag
e_Kmag r_Jmag
RAo DEo
Umag g'mag
r'mag i'mag
Zmag e_Umag
e_g'mag e_r'mag
e_i'mag e_Zmag
S/GU S/Gg'
S/Gr' S/Gi'
S/GZ r'magS
Astrogrid Portal - Mozilla
VizieR/J/MNRAS/325/1173/catalog
VizieR/J/A+A/379/798/table2
VizieR/J/MNRAS/351/1290/catalog
VizieR/J/MNRAS/351/1290/unassoc
VizieR/II/255/iraccat
VizieR/II/255/mips70
VizieR/II/255/mips160
Select... Restart Cancel Help
IVOA

Slide Credits

- Nicholas Walton
- Jon Loveday from Sussex
- Introduction to SQL from not sure(via Google)