

**Tutorial:**

**Building up and querying databases**

# Objectives

- During this tutorial you will build a custom database on data retrieved from the *BioMart* portal.
- *BioMart* provides a data acquisition and mining platform, suited especially to biological datasets. The *BioMart* homepage links to several data clusters, like *Ensembl Bacteria* and *Ensembl Plant*, which are dedicated to genomic, proteomic and other datasets of the respective organism clades.

# Step 1: Gathering data

- We will be focussing on the human genome and choose the **BioMart central portal** as our starting point (<http://www.biomart.org/>).
- Our first step will be to retrieve datasets for our self-made database. Let's assume we are especially interested in building a database on:
  - 1. The genomic context of all *Ensembl* genes on Chromosome X (ChrX)
  - 2. The Gene Ontology (GO) of all Ensembl genes on ChrX
  - 3. The splice variants of all ChrX transcripts.

# Step 2: Create / fill database

- The second step will take us to a database administration platform, where we will create a database to store, manage and query our retrieved data.
- Since we will use the *BioMart* retrieved data to build a database, we will need to come up with a suitable database scheme.

# The ER-scheme

So, before going to the *BioMart* site, take a minute to think about what kind of information you want to store in the database

*Hint: You will want to characterize your database entities sufficiently but exclude unnecessary or redundant information. For example, to characterize a gene you absolutely need to store location information, but GC content is not an essential information you will need to store.*

Next think about which entities will carry this information, and how you will be able to relate between these entities.

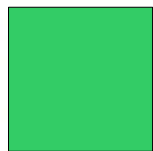
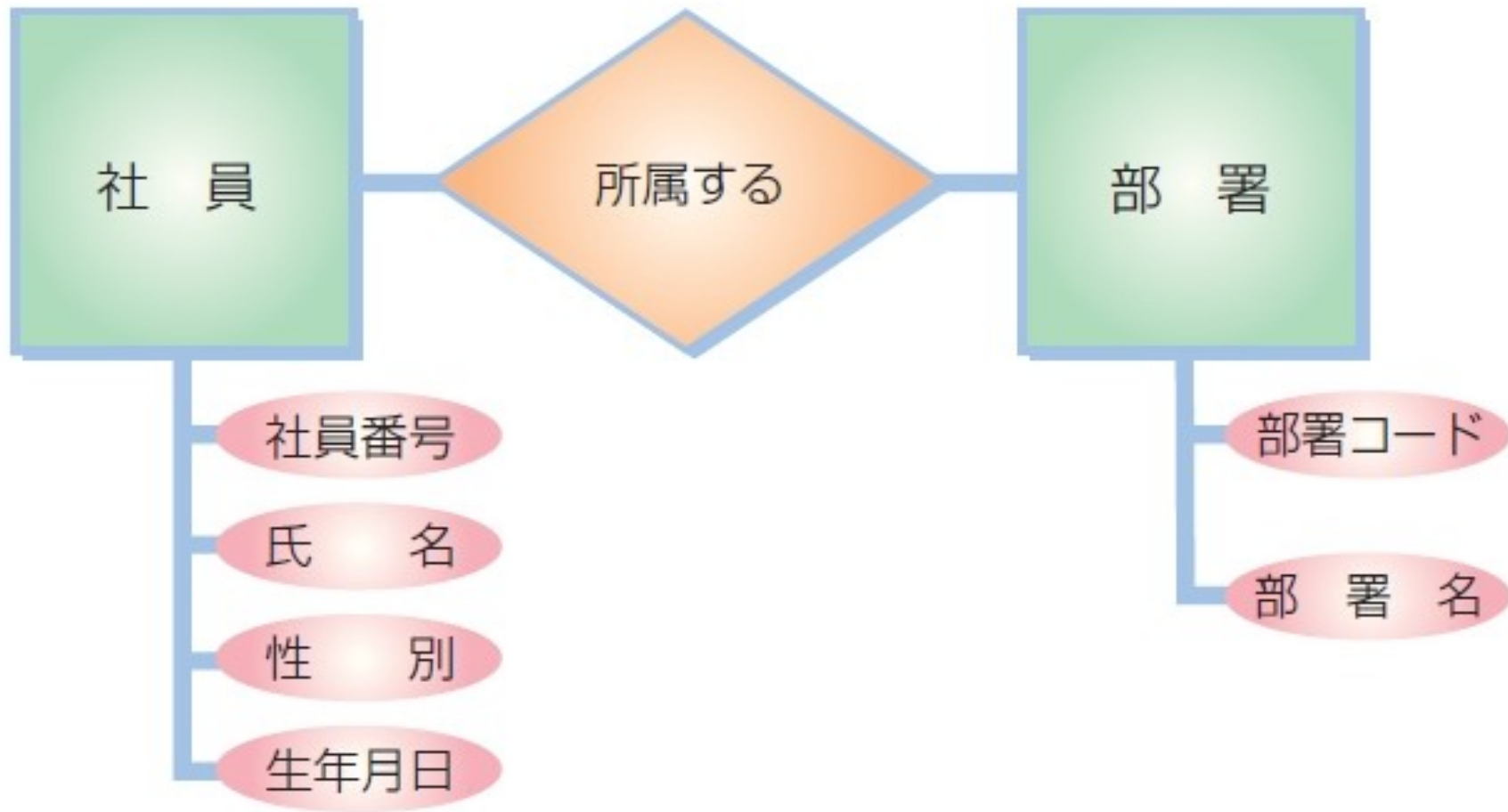
*Hint: Without some kind of ID to characterize entries in your database tables, the entity will be insular, or at least it will be difficult to refer to it from other entities.*

**Draw on paper an ER (Entity Relation) scheme to visualize the database you are going to build.**

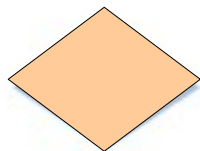
This is somehow the main point of database building and for this reason crucial for this tutorial.

# Realtional database scheme

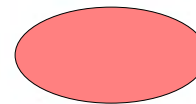
- Think of the different relation-types between the entities in your scheme.
- How can these be modelled in your database, by using tables?
- Surf the [www](http://www) for help. (hint: *cardinality*)



エンティティ



リレーションシップ



アトリビュート

Entity

Relation

Attribute

# The data retrieval:

## Browse the „BioMart Central Portal“

### BioMart Project

BioMart is a query-oriented data management system developed jointly by the [Ontario Institute for Cancer Research \(OICR\)](#) and the [European Bioinformatics Institute \(EBI\)](#).

The system can be used with any type of data and is particularly suited for providing 'data mining' like searches of complex descriptive data. BioMart comes with an 'out of the box' website that can be installed, configured and customised according to user requirements. Further access is provided by graphical and text based applications or programmatically using web services or API written in Perl and Java. BioMart has built-in support for query optimisation and data federation and in addition can be configured to work as a DAS 1.5 Annotation server. The process of converting a data source into BioMart format is fully automated by the tools included in the package. Currently supported RDBMS platforms are MySQL, Oracle and Postgres.

BioMart is completely Open Source, licensed under the LGPL, and freely available to anyone without restrictions.

#### Powered by BioMart software:

- [BioMart Central Portal](#)
- [Ensembl](#)
- [Ensembl Bacteria](#)
- [Ensembl Metazoa](#)
- [Ensembl Protists](#)
- [Ensembl Plants](#)
- [Ensembl Fungi](#)
- [Gramene](#)
- [Europhenome](#)
- [UniProt](#)
- [InterPro](#)
- [HGNC](#)
- [Wormbase](#)
- [DroSpeGe](#)
- [ArrayExpress DW](#)
- [Eurexpress](#)
- [HapMap](#)
- [Dictybase](#)
- [Rat Genome Database](#)
- [GermOnLine](#)
- [PRIDE](#)
- [PepSeeker](#)
- [VectorBase](#)
- [HTGT](#)
- [Pancreatic Expression Database](#)
- [Reactome](#)
- [EU Rat Mart](#)
- [Paramecium DB](#)
- [International Potato Center \(CIP\)](#)



New
Count
Results
★ URL
XML
Perl
Help

**Dataset**  
 [None selected]



Choose the Ensembl 56 Dataset and the human genome.

New Count Results
★ URL XML Perl Help

<b>Dataset</b>	ENSEMBL 56 GENES (SANGER UK) ▼
Homo sapiens genes (GRCh37)	Homo sapiens genes (GRCh37) ▼
<b>Filters</b>	
[None selected]	
<b>Attributes</b>	
Ensembl Gene ID	
Ensembl Transcript ID	
<b>Dataset</b>	
[None Selected]	

Restrict the region to the X-Chromosome

New Count Results
★ URL XML Perl Help

**Please restrict your query using criteria below**

<b>Dataset</b>	Homo sapiens genes (GRCh37)	
<b>Filters</b>	<div style="background-color: yellow; border: 1px solid black; padding: 2px;">REGION:</div> <input checked="" type="checkbox"/> Chromosome <span style="margin-left: 100px;">X ▼</span>	
Chromosome: X		
<b>Attributes</b>	<input type="checkbox"/> Base pair	
Ensembl Gene ID	Gene Start (bp)	1
Ensembl Transcript ID	Gene End (bp)	10000000

Next select the „GENE“ pop down menu in the „Attributes“ section, under the attribute subcategory „Features“.

Here you will find all downloadable information related to the genes themselves. Select those that you determine necessary to characterize the genes.

Check out the other pop down menus as well and find what information is presented there.

**Please select columns to be included in the output and hit 'Results' when ready**

**Dataset** 2379 / 49506 Genes

Homo sapiens genes (GRCh37)

**Filters**

Chromosome: X

**Attributes**

[None selected]

---

**Dataset**

[None Selected]

**Features**
 **Transcript Event**

**Structures**
 **Homologs**

**Variations**
 **Sequences**

☐ GENE:

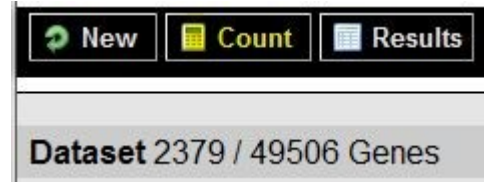
**Ensembl**

- Ensembl Gene ID
- Ensembl Transcript ID
- Ensembl Protein ID
- Canonical transcript stable ID(s)
- Description
- Chromosome Name
- Gene Start (bp)
- Gene End (bp)
- Strand
- Band
- Transcript Start (bp)
- Transcript End (bp)

?

- Associated Gene Name
- Associated Transcript Name
- Associated Gene DB
- Associated Transcript DB
- Transcript count
- % GC content
- Gene Biotype
- Transcript Biotype
- Source
- Status (gene)
- Status (transcript)

With the count tab at the top right you can get a rough estimate on how large your dataset will be.



The Results tab will retrieve the data according to the selected attributes and filters. It is important to mark the „Unique results only“ check box. Write down how many columns your table has, whether it contains numbers or letters(+numbers) and how many letters there are in every column.

New Count **Results**
★ URL XML Perl Help

**Dataset** 2379 / 49506 Genes  
 Homo sapiens genes (GRCh37)

**Filters**  
 Chromosome: X

**Attributes**  
 Ensembl Gene ID  
 Strand  
 Band  
 Gene Start (bp)  
 Gene End (bp)  
 Chromosome Name

**Dataset**  
 [None Selected]

Export all results to:  TSV  Unique results only

Email notification to:

View: 10 rows as HTML  Unique results only

Ensembl Gene ID	Strand	Band	Gene Start (bp)	Gene End (bp)	Chromosome Name
<a href="#">ENSG00000232765</a>	1	<a href="#">p11.1</a>	<a href="#">55306576</a>	<a href="#">55315219</a>	X
<a href="#">ENSG00000196433</a>	1	<a href="#">p22.33</a>	<a href="#">1733894</a>	<a href="#">1761974</a>	X
<a href="#">ENSG00000204131</a>	1	<a href="#">q13.1</a>	<a href="#">71353499</a>	<a href="#">71361212</a>	X
<a href="#">ENSG00000126945</a>	1	<a href="#">q22.1</a>	<a href="#">100663191</a>	<a href="#">100669121</a>	X
<a href="#">ENSG00000241465</a>	1	<a href="#">p11.23</a>	<a href="#">49178558</a>	<a href="#">49223943</a>	X
<a href="#">ENSG00000239469</a>	1	<a href="#">q22.3</a>	<a href="#">107020963</a>	<a href="#">107037689</a>	X
<a href="#">ENSG00000170935</a>	1	<a href="#">q22.3</a>	<a href="#">107037451</a>	<a href="#">107037912</a>	X
<a href="#">ENSG00000240894</a>	1	<a href="#">q22.2</a>	<a href="#">102840506</a>	<a href="#">102841753</a>	X
<a href="#">ENSG00000182484</a>	1	<a href="#">q28</a>	<a href="#">155249967</a>	<a href="#">155255375</a>	X
<a href="#">ENSG00000189108</a>	1	<a href="#">q22.3</a>	<a href="#">103810996</a>	<a href="#">105011822</a>	X

Export the data as a .csv (comma separated value) file – in the compressed form, via the menu at the top.

You might want to rename the file after downloading it.

---

Export all results to

Compressed file (.gz)

CSV

Unique results only

Email notification to

You can find GO data under the „External“ drop down menu under the Features Attributes subcategory. Transcript related data is available directly under *Features->Gene->Ensembl*, just like for the genes themselves.

As you assemble your data collection, reconsider which attributes and entities are necessary. Feel free to debate this with others and the tutor – there are many ways to build a database and finding the optimal attribute set/arrangement is not always easy. It also might depend on the focus of interest...

When you have downloaded all the data you want to integrate in your database you can use **phpmyadmin** to upload the data into the database server and build the database.

Go to <http://skinner/phpmyadmin> and log in as dbuser01 ... dbuser30.  
Please ask the tutor which username to take.

To create a new database enter the name of the database (no empty space or special characters) and click create.  
You can change the language of the interface at the bottom left panel

The screenshot displays the phpMyAdmin interface with a top navigation bar containing tabs for Databases, SQL, Status, Variables, Charsets, Engines, Privileges, Replication, Processes, Export, and Synchronize. The main content area is divided into several panels:

- Actions**: A section for performing database actions.
- MySQL localhost**: A panel for creating a new database. It features a "Create new database" button, a text input field (highlighted in yellow), a "Collation" dropdown menu, and a "Create" button. Below this, it shows the "MySQL connection collation" set to "utf8\_general\_ci".
- Interface**: A panel for customizing the user interface. It includes a "Language" dropdown set to "English", a "Theme / Style" dropdown set to "Original", a "Custom color" section with a "Reset" button, and a "Font size" dropdown set to "82%".
- MySQL**: A panel displaying server information, including "Server: Localhost via UNIX socket", "Server version: 5.1.40-1", "Protocol version: 10", "User: root@localhost", and "MySQL charset: UTF-8 Unicode (utf8)".
- Web server**: A panel showing web server details, such as "lighttpd/1.4.24", "MySQL client version: 5.1.40", and "PHP extension: mysqli". It also includes a link to "Show PHP information".
- phpMyAdmin**: A panel providing version information ("3.3.0-dev") and a link to "Documentation".





phpMyAdmin

Database

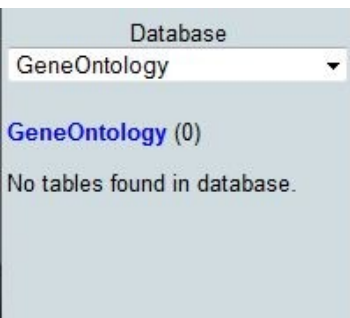
GeneOntology

GeneOntology (0)

No tables found in database.

After creating the database you will find it at the top left. The number behind the db name indicates the number of tables. So far the database is empty

To fill it click on the database and the „Structure“ tab. Enter the name of the new table and the number of columns it is going to have. This needs to be exact or the data import will fail.



Database

GeneOntology

GeneOntology (0)

No tables found in database.



Structure SQL Search Tracking Query Export Import Designer Operations Privileges Drop

No tables found in database.

Create new table on database GeneOntology

Name: Genes Number of fields: 6

Go



Phpmyadmin will ask you to specify the type of attribute in every column. If the column will contain letters only, or numbers as well as letters, choose „VARCHAR“, if it contains a number only, choose „INT“.

You need to enter a length that is at least as large as the maximum length of attributes in this column.

Field	Type ?	Length/Values <sup>1</sup>	Default <sup>2</sup>	Collation	
GeneID	VARCHAR	15	None		
...	INT		None		
...	INT		None		
	INT		None		
	INT		None		
	INT		None		

**Table comments:**

**Storage Engine: ?**

MyISAM

**Collation:**





































**PARTITION definition: ?**







Think about whether you can declare one of the columns to be a primary key. It is advisable that every table has a primary key, if possible. The primary key column can only contain unique values, or the import will fail.

Tracking of GeneOntology.Genes is activated.

Table 'GeneOntology'. 'Genes' has been created.

```
CREATE TABLE 'GeneOntology'. 'Genes' (
  'GeneID' VARCHAR( 15 ) NOT NULL ,
  'Strand' INT( 1 ) NOT NULL ,
  'Band' VARCHAR( 10 ) NOT NULL ,
  'Start' INT( 15 ) NOT NULL ,
  'Stop' INT( 15 ) NOT NULL ,
  'Chr' VARCHAR( 2 ) NOT NULL
) ENGINE = MYISAM ;
```

Field	Type	Collation	Attributes	Null	Default	Extra	Action
<input type="checkbox"/> GeneID	varchar(15)	latin1_swedish_ci		No	None		     
<input type="checkbox"/> Strand	int(1)			No	None		     
<input type="checkbox"/> Band	varchar(10)	latin1_swedish_ci		No	None		     
<input type="checkbox"/> Start	int(15)			No	None		     
<input type="checkbox"/> Stop	int(15)			No	None		     
<input type="checkbox"/> Chr	varchar(2)	latin1_swedish_ci		No	None		     

Check All / Uncheck All With selected:      

Print view Relation view Propose table structure Track table

Add 1 field(s) At End of Table At Beginning of Table After GeneID Go

No index defined!

*define index*

Now the table is ready to be filled with our data. Click on the „import“ tab and upload the first table data. You need to specify the format as csv and change the „Fields terminated by“ field to comma (",") - since this is what our csv is delimited by

**Browse** **Structure** **SQL** **Search** **Tracking** **Insert** **Export** **Import** **Operations** **Empty** **Drop**

**i** Tracking of GeneOntology.Genes is activated.

**File to import**

Location of the text file: Downloads\Genes.txt.gz **Browse...** (Max: 2,048 KiB)

Character set of the file: utf8

Imported file compression will be automatically detected from: None, gzip, bzip2, zip

**Partial import**

Allow the interruption of an import in case the script detects it is close to the PHP timeout limit. This might be good way to import large files, however it can break transactions.

Number of records (queries) to skip from start:

**Format of imported file**

**CSV**

CSV using LOAD DATA

Open Document Spreadsheet

SQL

Excel 97-2003 XLS Workbook

Excel 2007 XLSX Workbook

XML

**Options**

Replace table data with file

Ignore duplicate rows

**Fields terminated by**  *" , " comma*

Fields enclosed by

Fields escaped by

Lines terminated by

Column names

**Go**



Under „Browse“ you can check whether the import is successful. You may need to delete the first row if it does not contain actual attributes but the column names of BioMart. Alternatively u can skip the first line during import (see previous slide/ red circle-> change from '0' to '1').

[Browse](#)
[Structure](#)
[SQL](#)
[Search](#)
[Tracking](#)
[Insert](#)
[Export](#)
[Import](#)
[Operations](#)
[Empty](#)
[Drop](#)

Tracking of GeneOntology.Genes is activated.

Showing rows 0 - 29 (2,380 total, Query took 0.0004 sec)

```

SELECT *
FROM `Genes`
LIMIT 0, 30
    
```

Profiling [ [Edit](#) ] [ [Explain SQL](#) ] [ [Create PHP Code](#) ] [ [Refresh](#) ]

Show:  row(s) starting from record #

in  mode and repeat headers after  cells

Sort by key:

+ Options

	GeneID	Strand	Band	Start	Stop	Chr
<input checked="" type="checkbox"/>	Ensembl Gene ID	0	Band	0	0	Ch
<input type="checkbox"/>	ENSG00000232765	1	p11.1	55306576	55315219	X
<input type="checkbox"/>	ENSG00000196433	1	p22.33	1733894	1761974	X
<input type="checkbox"/>	ENSG00000204131	1	q13.1	71353499	71361212	X
<input type="checkbox"/>	ENSG00000126945	1	q22.1	100663191	100669121	X
<input type="checkbox"/>	ENSG00000241465	1	p11.23	49178558	49223943	X
<input type="checkbox"/>	ENSG00000239469	1	q22.3	107020963	107037689	X
<input type="checkbox"/>	ENSG00000170935	1	q22.3	107037451	107037912	X
<input type="checkbox"/>	ENSG00000240894	1	q22.2	102840506	102841753	X
<input type="checkbox"/>	ENSG00000182484	1	q28	155249967	155255375	X
<input type="checkbox"/>	ENSG00000189108	1	q22.3	103810996	105011822	X

Now click on your database (at the top) and add the next table, following the same steps as with the first.

The screenshot shows the phpMyAdmin interface for the 'GeneOntology' database. The 'Structure' tab is active, displaying a table with the following data:

Table	Action	Records <sup>1</sup>	Type	Collation	Size	Overhead
Genes		2,380	MyISAM	latin1_swedish_ci	128.4 KiB	-
<b>1 table(s)</b>	<b>Sum</b>	<b>2,380</b>	<b>MyISAM</b>	<b>latin1_swedish_ci</b>	<b>128.4 KiB</b>	<b>0 B</b>

Below the table, there is a 'Create new table on database GeneOntology' dialog box. The 'Name' field contains 'Gene2Trans' and the 'Number of fields' is set to '...'. A 'Go' button is located at the bottom right of the dialog box.

At the bottom of the interface, a yellow information bar states: <sup>1</sup> May be approximate. See [FAQ 3.11](#)

In the end you should have three tables imported

The screenshot shows the phpMyAdmin interface for the 'GeneOntology' database. The 'Structure' tab is active, displaying three tables with the following data:

Table	Action	Records <sup>1</sup>	Type	Collation	Size	Overhead
Gene2Trans		6,229	MyISAM	latin1_swedish_ci	309.0 KiB	-
Genes		2,380	MyISAM	latin1_swedish_ci	128.4 KiB	-
TransOnt		5,628	MyISAM	latin1_swedish_ci	275.4 KiB	-
<b>3 table(s)</b>	<b>Sum</b>	<b>14,237</b>	<b>MyISAM</b>	<b>latin1_swedish_ci</b>	<b>712.8 KiB</b>	<b>0 B</b>

The left sidebar shows the database structure with three tables listed: Gene2Trans, Genes, and TransOnt.

Last, click on the „SQL“ tab and try out some queries as showed below (next slide).

What do these queries achieve?  
At least try to understand, please.  
Discuss with others or the tutor.

```
SELECT * FROM gene WHERE gene_id = "ENSG00000232765";
```

```
SELECT * FROM gene LIMIT 3;
```

```
SELECT count(*) FROM gene;
```

```
SELECT gene.name FROM gene LIMIT 5;
```

```
SELECT count(*) FROM gene, transcript WHERE gene.gene_id =  
transcript.gene_id;
```

```
SELECT count(*) FROM gene JOIN transcript ON gene.gene_id  
=transcript.gene_id;
```

```
SELECT transcript.gene_id FROM ...;
```

```
SELECT transcript.gene_id, gene.band FROM ...;
```

```
SELECT transcript.gene_id, gene.band FROM ... ... WHERE gene.band  
="q28";
```

```
SELECT gene.name, gene.gene_id, go.go_id, go_term FROM gene, gene2go, go  
WHERE gene.gene_id = gene2go.gene_id AND gene2go.go_id = go.go_id;
```

```
SELECT g.name, go.go_term FROM gene g JOIN gene2go g2go ON g.gene_id =  
g2go.gene_id JOIN go go ON go.go_id = g2go.go_id WHERE go.go_term LIKE  
"%immune response%";
```

```
SELECT AVG((transcript.trans_end - transcript.trans_start)) FROM  
transcript;
```