

Aquisição de Arquivos e Datasets

Lendo arquivos MySQL

Delermundo Branquinho Filho

mySQL

- Livre e amplamente utilizado software de banco de dados de código aberto
 - Amplamente utilizado em aplicações baseadas na Internet
 - Os dados estão estruturados em
 - Bancos de dados
 - Tabelas em bases de dados
 - Campos dentro de tabelas
 - Cada linha é chamada de registro
-

Instalando RMySQL

```
# install.packages("RMySQL")  
library("RMySQL")
```

```
## Warning: package 'RMySQL' was built under R version 3.3.3
```

```
## Loading required package: DBI
```

```
## Warning: package 'DBI' was built under R version 3.3.3
```

Connectando e listando databases

```
ucscDb <- dbConnect(MySQL(),user="genome",  
                    host="genome-mysql.cse.ucsc.edu")  
result <- dbGetQuery(ucscDb,"show databases;"); dbDisconnect(ucscDb);
```

```
## [1] TRUE
```

```
result
```

```
##           Database  
## 1 information_schema  
## 2      ailMel1  
## 3      allMis1  
## 4      anoCar1  
## 5      anoCar2  
## 6      anoGam1  
## 7      apiMel1  
## 8      apiMel2  
## 9      aplCal1  
## 10     aptMan1  
## 11     balAcu1
```

## 12	bosTau2
## 13	bosTau3
## 14	bosTau4
## 15	bosTau5
## 16	bosTau6
## 17	bosTau7
## 18	bosTau8
## 19	bosTauMd3
## 20	braFlo1
## 21	caeJap1
## 22	caePb1
## 23	caePb2
## 24	caeRem2
## 25	caeRem3
## 26	calJac1
## 27	calJac3
## 28	calMil1
## 29	canFam1
## 30	canFam2
## 31	canFam3
## 32	cavPor3
## 33	cb1
## 34	cb3
## 35	ce10
## 36	ce11
## 37	ce2
## 38	ce4
## 39	ce6
## 40	cerSim1
## 41	chlSab2
## 42	choHof1
## 43	chrPic1
## 44	ci1
## 45	ci2
## 46	criGri1
## 47	danRer1
## 48	danRer10
## 49	danRer2
## 50	danRer3
## 51	danRer4
## 52	danRer5
## 53	danRer6
## 54	danRer7
## 55	dasNov3
## 56	dipOrd1
## 57	dm1
## 58	dm2
## 59	dm3
## 60	dm6
## 61	dp2
## 62	dp3
## 63	droAna1
## 64	droAna2
## 65	droEre1

## 66	droGri1
## 67	droMoj1
## 68	droMoj2
## 69	droPer1
## 70	droSec1
## 71	droSim1
## 72	droVir1
## 73	droVir2
## 74	droYak1
## 75	droYak2
## 76	eboVir3
## 77	echTel1
## 78	echTel2
## 79	equCab1
## 80	equCab2
## 81	eriEur1
## 82	eriEur2
## 83	felCat3
## 84	felCat4
## 85	felCat5
## 86	felCat8
## 87	fr1
## 88	fr2
## 89	fr3
## 90	gadMor1
## 91	galGal2
## 92	galGal3
## 93	galGal4
## 94	galGal5
## 95	galVar1
## 96	gasAcu1
## 97	gbMeta
## 98	geoFor1
## 99	go
## 100	go080130
## 101	go140213
## 102	go150121
## 103	gorGor3
## 104	gorGor4
## 105	gorGor5
## 106	hetGla1
## 107	hetGla2
## 108	hg16
## 109	hg17
## 110	hg18
## 111	hg19
## 112	hg19Patch10
## 113	hg19Patch2
## 114	hg19Patch5
## 115	hg19Patch9
## 116	hg38
## 117	hg38Patch2
## 118	hg38Patch3
## 119	hg38Patch6

## 120	hg38Patch7
## 121	hg38Patch9
## 122	hgFixed
## 123	hgTemp
## 124	hgcentral
## 125	latCha1
## 126	loxAfr3
## 127	macEug1
## 128	macEug2
## 129	macFas5
## 130	melGal1
## 131	melGal5
## 132	melUnd1
## 133	micMur1
## 134	micMur2
## 135	mm10
## 136	mm10Patch1
## 137	mm10Patch4
## 138	mm5
## 139	mm6
## 140	mm7
## 141	mm8
## 142	mm9
## 143	monDom1
## 144	monDom4
## 145	monDom5
## 146	musFur1
## 147	myoLuc2
## 148	nasLar1
## 149	nomLeu1
## 150	nomLeu2
## 151	nomLeu3
## 152	ochPri2
## 153	ochPri3
## 154	oreNil1
## 155	oreNil2
## 156	ornAna1
## 157	ornAna2
## 158	oryCun2
## 159	oryLat2
## 160	otoGar3
## 161	oviAri1
## 162	oviAri3
## 163	panPan1
## 164	panTro1
## 165	panTro2
## 166	panTro3
## 167	panTro4
## 168	panTro5
## 169	papAnu2
## 170	papHam1
## 171	performance_schema
## 172	petMar1
## 173	petMar2

## 174	ponAbe2
## 175	priPac1
## 176	proCap1
## 177	proteins120806
## 178	proteins121210
## 179	proteins140122
## 180	proteins150225
## 181	proteins160229
## 182	proteome
## 183	pteVam1
## 184	rheMac1
## 185	rheMac2
## 186	rheMac3
## 187	rheMac8
## 188	rhiRox1
## 189	rn3
## 190	rn4
## 191	rn5
## 192	rn6
## 193	sacCer1
## 194	sacCer2
## 195	sacCer3
## 196	saiBol1
## 197	sarHar1
## 198	sorAra1
## 199	sorAra2
## 200	sp120323
## 201	sp121210
## 202	sp140122
## 203	sp150225
## 204	sp160229
## 205	speTri2
## 206	strPur1
## 207	strPur2
## 208	susScr2
## 209	susScr3
## 210	taeGut1
## 211	taeGut2
## 212	tarSyr1
## 213	tarSyr2
## 214	test
## 215	tetNig1
## 216	tetNig2
## 217	triMan1
## 218	tupBel1
## 219	turTru2
## 220	uniProt
## 221	vicPac1
## 222	vicPac2
## 223	visiGene
## 224	xenTro1
## 225	xenTro2
## 226	xenTro3
## 227	xenTro7

Conectando em ailMel1 e listando as tabelas

```
hg19 <- dbConnect(MySQL(),user="genome", db="ailMel1",
                  host="genome-mysql.cse.ucsc.edu")
allTables <- dbListTables(hg19)
length(allTables)

## [1] 53

allTables[1:5]

## [1] "all_est"      "all_mrna"     "augustusGene" "bigFiles"
## [5] "blastHg18KG"
```

Obter dimensões de uma tabela específica

```
dbListFields(hg19, "augustusGene")

## [1] "bin"          "name"         "chrom"        "strand"
## [5] "txStart"     "txEnd"        "cdsStart"     "cdsEnd"
## [9] "exonCount"   "exonStarts"   "exonEnds"     "score"
## [13] "name2"       "cdsStartStat" "cdsEndStat"   "exonFrames"

dbGetQuery(hg19, "select count(*) from augustusGene")

##      count(*)
## 1      29317
```

Ler da tabela

```
affyData <- dbReadTable(hg19, "augustusGene")

## Warning in .local(conn, statement, ...): Unsigned INTEGER in col 0 imported
## as numeric
## Warning in .local(conn, statement, ...): Unsigned INTEGER in col 4 imported
## as numeric
## Warning in .local(conn, statement, ...): Unsigned INTEGER in col 5 imported
## as numeric
## Warning in .local(conn, statement, ...): Unsigned INTEGER in col 6 imported
## as numeric
## Warning in .local(conn, statement, ...): Unsigned INTEGER in col 7 imported
## as numeric
## Warning in .local(conn, statement, ...): Unsigned INTEGER in col 8 imported
## as numeric
```

```
head(affyData)
```

```
##   bin  name      chrom strand txStart  txEnd  cdsStart  cdsEnd  exonCount
## 1  73  g1.t1  GL192338.1    +    705 213405    760 213168        40
## 2  73  g2.t1  GL192338.1    + 239585 290632   239641 289331        23
## 3  73  g2.t2  GL192338.1    + 239585 290632   239641 289331        23
## 4 587  g3.t1  GL192338.1    - 290667 313965   291366 313726         8
## 5  73  g4.t1  GL192338.1    - 374062 464225   376895 463296        24
## 6  73  g4.t2  GL192338.1    - 374062 445055   376895 444225        19
##
## 1 705,20458,74837,118198,129380,133173,134214,134368,135861,138896,139187,143451,147070,151156,152459
## 2
## 3
## 4
## 5
## 6
##
## 1 816,20570,76295,118530,129492,133320,134283,134424,135916,138983,139325,143573,147314,151333,152599
## 2
## 3
## 4
## 5
## 6
##   score name2 cdsStartStat cdsEndStat
## 1     0    g1      compl      compl
## 2     0    g2      compl      compl
## 3     0    g2      compl      compl
## 4     0    g3      compl      compl
## 5     0    g4      compl      compl
## 6     0    g4      compl      compl
##
##                                     exonFrames
## 1 0,2,0,0,2,0,0,0,2,0,0,0,2,0,0,0,1,2,0,0,0,2,1,2,2,1,2,0,1,2,1,1,0,0,1,0,0,0,0,
## 2                                     0,0,2,2,1,1,0,2,0,2,2,1,0,0,1,1,0,0,0,1,0,0,-1,
## 3                                     0,0,2,2,2,1,1,0,2,0,2,2,1,0,0,1,1,0,0,0,1,0,0,
## 4                                     0,2,2,2,0,1,2,0,
## 5                                     1,2,1,0,1,0,0,0,1,1,0,0,0,1,2,0,0,2,2,1,2,1,0,-1,
## 6                                     1,2,1,0,1,0,0,0,1,1,0,0,0,1,2,0,0,0,-1,
```

Selecionar um subconjunto específico

```
query <- dbSendQuery(hg19, "select * from augustusGene where txStart between 700 and 239585")
affyMis <- fetch(query); quantile(affyMis$misMatches)
```

```
##   0%  25%  50%  75% 100%
##   NA   NA   NA   NA   NA
```

```
affyMisSmall <- fetch(query,n=10); dbClearResult(query);
```

```
## [1] TRUE
```

```
dim(affyMisSmall)
```

```
## [1] 10 16
```

Don't forget to close the connection!

```
dbDisconnect(hg19)
```

```
## [1] TRUE
```
