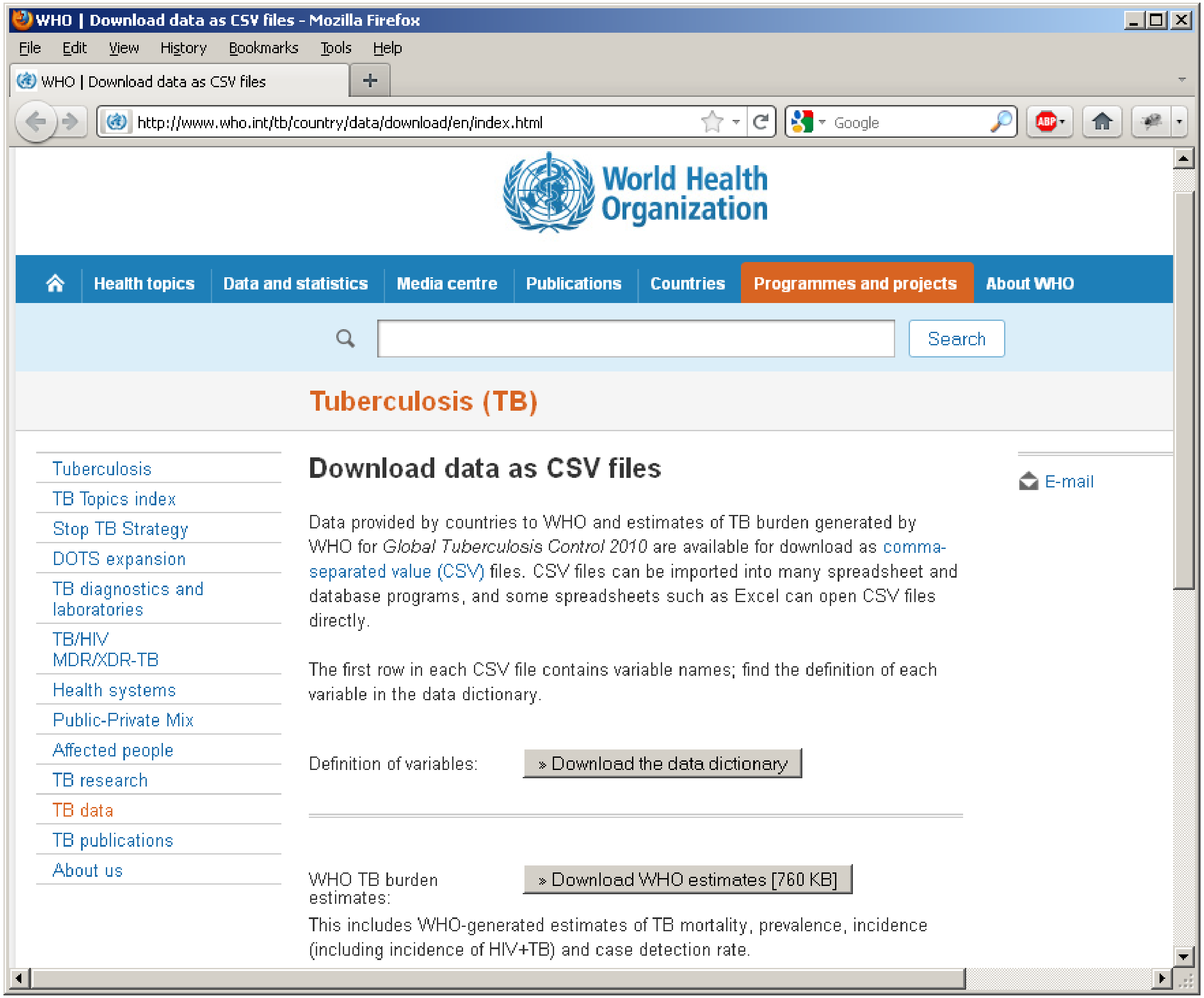
WHO Tuberculosis Data



WHO | Download data as CSV files - Mozilla Firefox
File Edit View History Bookmarks Tools Help
WHO | Download data as CSV files
http://www.who.int/tb/country/data/download/en/index.html
Google
World Health Organization
Health topics
Data and statistics
Media centre
Publications
Countries
Programmes and projects
About WHO
Search

Tuberculosis (TB)

- Tuberculosis
- TB Topics index
- Stop TB Strategy
- DOTS expansion
- TB diagnostics and laboratories
- TB/HIV
- MDR/XDR-TB
- Health systems
- Public-Private Mix
- Affected people
- TB research
- TB data
- TB publications
- About us

E-mail

Download data as CSV files

Data provided by countries to WHO and estimates of TB burden generated by WHO for *Global Tuberculosis Control 2010* are available for download as comma-separated value (CSV) files. CSV files can be imported into many spreadsheet and database programs, and some spreadsheets such as Excel can open CSV files directly.

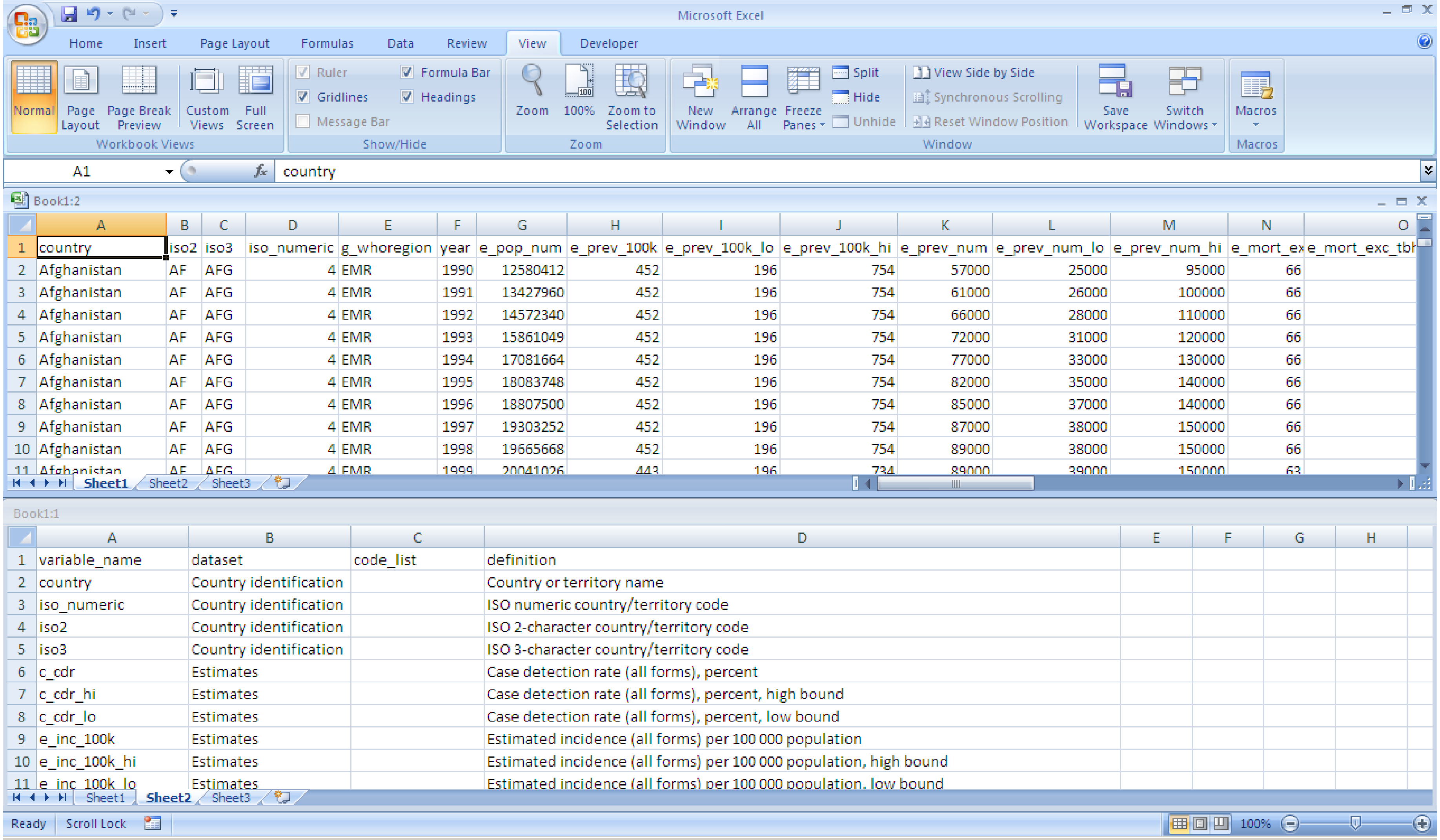
The first row in each CSV file contains variable names; find the definition of each variable in the data dictionary.

Definition of variables: » Download the data dictionary

WHO TB burden estimates: » Download WHO estimates [760 KB]

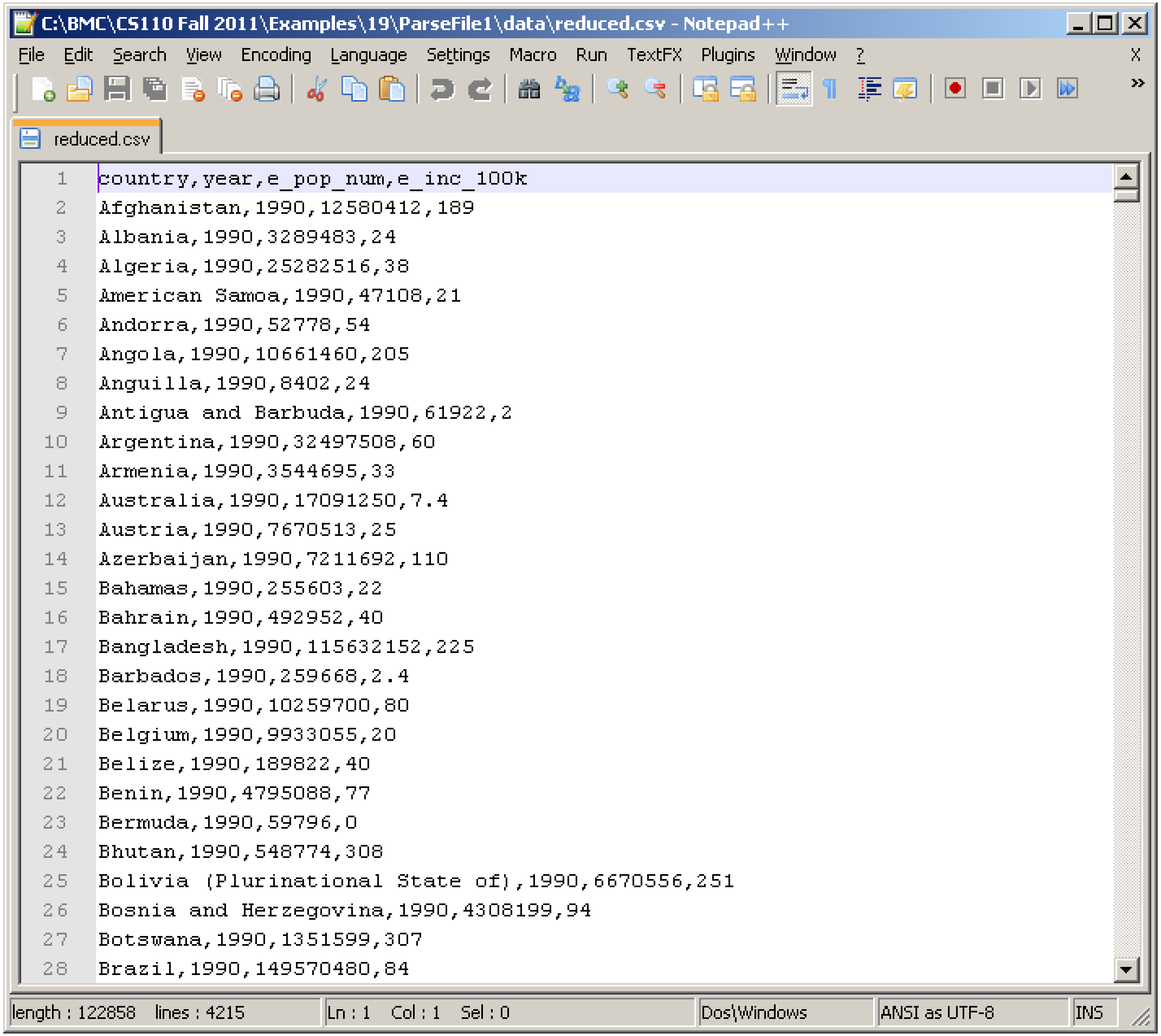
This includes WHO-generated estimates of TB mortality, prevalence, incidence (including incidence of HIV+TB) and case detection rate.

http://www.who.int/tb/country/data/download/en/index.html



	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
1	country	iso2	iso3	iso_numeric	g_whoregion	year	e_pop_num	e_prev_100k	e_prev_100k_lo	e_prev_100k_hi	e_prev_num	e_prev_num_lo	e_prev_num_hi	e_mort_ex	e_mort_exc_tb
2	Afghanistan	AF	AFG	4	EMR	1990	12580412	452	196	754	57000	25000	95000	66	
3	Afghanistan	AF	AFG	4	EMR	1991	13427960	452	196	754	61000	26000	100000	66	
4	Afghanistan	AF	AFG	4	EMR	1992	14572340	452	196	754	66000	28000	110000	66	
5	Afghanistan	AF	AFG	4	EMR	1993	15861049	452	196	754	72000	31000	120000	66	
6	Afghanistan	AF	AFG	4	EMR	1994	17081664	452	196	754	77000	33000	130000	66	
7	Afghanistan	AF	AFG	4	EMR	1995	18083748	452	196	754	82000	35000	140000	66	
8	Afghanistan	AF	AFG	4	EMR	1996	18807500	452	196	754	85000	37000	140000	66	
9	Afghanistan	AF	AFG	4	EMR	1997	19303252	452	196	754	87000	38000	150000	66	
10	Afghanistan	AF	AFG	4	EMR	1998	19665668	452	196	754	89000	38000	150000	66	
11	Afghanistan	AF	AFG	4	EMR	1999	20041026	443	196	734	89000	39000	150000	63	

	A	B	C	D	E	F	G	H
1	variable_name	dataset	code_list	definition				
2	country	Country identification		Country or territory name				
3	iso_numeric	Country identification		ISO numeric country/territory code				
4	iso2	Country identification		ISO 2-character country/territory code				
5	iso3	Country identification		ISO 3-character country/territory code				
6	c_cdr	Estimates		Case detection rate (all forms), percent				
7	c_cdr_hi	Estimates		Case detection rate (all forms), percent, high bound				
8	c_cdr_lo	Estimates		Case detection rate (all forms), percent, low bound				
9	e_inc_100k	Estimates		Estimated incidence (all forms) per 100 000 population				
10	e_inc_100k_hi	Estimates		Estimated incidence (all forms) per 100 000 population, high bound				
11	e_inc_100k_lo	Estimates		Estimated incidence (all forms) per 100 000 population, low bound				



C:\BMC\CS110 Fall 2011\Examples\19\ParseFile1\data\reduced.csv - Notepad++
File Edit Search View Encoding Language Settings Macro Run TextFX Plugins Window ?
reduced.csv
length : 122858 lines : 4215
Ln : 1 Col : 1 Sel : 0
Dos\Windows
ANSI as UTF-8
INS

```
country,year,e_pop_num,e_inc_100k
Afghanistan,1990,12580412,189
Albania,1990,3289483,24
Algeria,1990,25282516,38
American Samoa,1990,47108,21
Andorra,1990,52778,54
Angola,1990,10661460,205
Anguilla,1990,8402,24
Antigua and Barbuda,1990,61922,2
Argentina,1990,32497508,60
Armenia,1990,3544695,33
Australia,1990,17091250,7.4
Austria,1990,7670513,25
Azerbaijan,1990,7211692,110
Bahamas,1990,255603,22
Bahrain,1990,492952,40
Bangladesh,1990,115632152,225
Barbados,1990,259668,2.4
Belarus,1990,10259700,80
Belgium,1990,9933055,20
Belize,1990,189822,40
Benin,1990,4795088,77
Bermuda,1990,59796,0
Bhutan,1990,548774,308
Bolivia (Plurinational State of),1990,6670556,251
Bosnia and Herzegovina,1990,4308199,94
Botswana,1990,1351599,307
Brazil,1990,149570480,84
```

Loading Data From a File

- loadStrings() function
  - Reads all data and returns an array of Strings
  - Each String in the array is a separate line from the file

```
// ParseFile1

String[] data;
int count = 0;

void setup() {
  // Load data from a file as array of strings
  data = loadStrings("reduced.csv");
}

void draw() {
  // Continue printing data until run out
  if (count >= data.length) return;
  println(data[count]);
  count++;
}
```

Split a String based on a single or multiple separator chars

```
String s1 = "12, 34, 56";
String[] as;

void setup() {
  as = split(s1, ",");
  //as = trim(as);
  println( as );
}
```

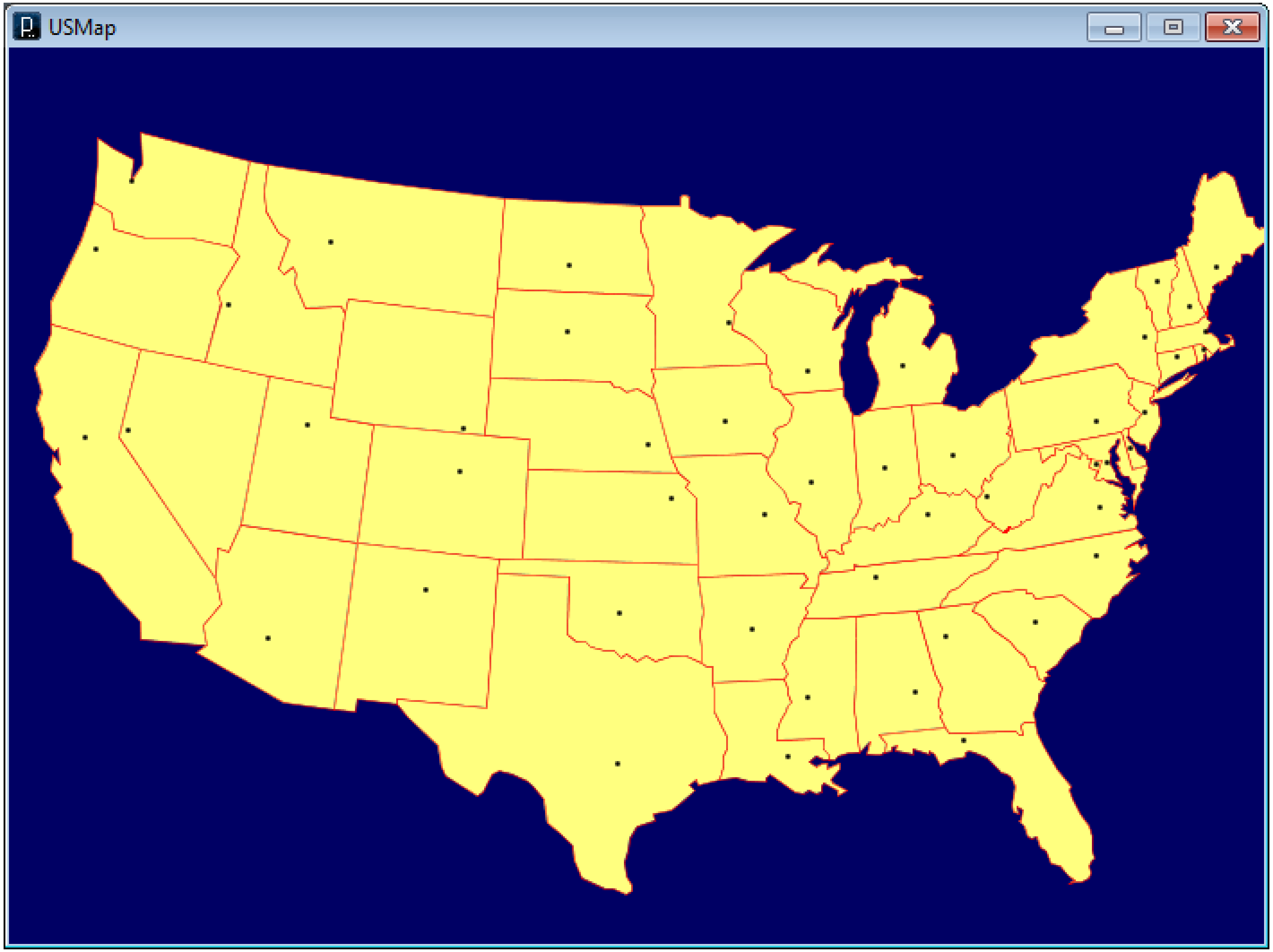
```
[0] "12"
[1] " 34"
[2] " 56"
```

```
String s1 = "Data: 12, 34, 56";
String[] as;

void setup() {
  as = splitTokens(s1, ":,");
  //as = trim(as);
  println( as );
}
```

```
[0] "Data"
[1] " 12"
[2] " 34"
[3] " 56"
```

```
// ParseFile2

String[] data;
Item[] items;
int count = 0;

void setup() {
  // Load data as array of strings
  data = loadStrings("reduced.csv");

  // Build object array
  items = new Item[data.length];
  for (int i=0; i<data.length; i++) {
    items[i] = new Item(data[i]);
  }
}

void draw() {
  // Continue printing data until run out
  if (count >= items.length) return;
  items[count].pr();
  count++;
}
```

```
class Item {

  String country;  // Country name
  int year;        // Year
  int pop;         // Population
  int inc;         // Incidences of TB
                   // per 100,000

  Item(String line) {
    String[] data = split(line, ",");
    country = data[0];
    year = int(data[1]);
    pop = int(data[2]);
    inc = int(data[3]);
  }

 void pr() {
    String msg = "In " + year + ", " + country;
    msg += " had population " + pop;
    msg += " and TB incidences per 100k of " + inc;
    println(msg);
  }  }
}
```



USMap.pde, USMap2.pde

File Selection

- selectInput() function
  - Displays a file chooser allowing user to select a file
  - Returns a complete path to selected file as a String

```
void draw() { }

void mousePressed()
{
  String filepath = selectInput("Please select a data file");
  println(filepath);
}
```

Also see selectOutput()

filepath.pde

```
// stripSpaces

void setup() {
  String s = "abc def vghi";
  String s3 = stripSpaces(s);
  println(s3);
}

// Remove spaces from a string
String stripSpaces( String s ) {

  for (int i=s.length()-1; i>=0; i--) {
    char c = s.charAt(i);
    if (c == ' ') {
      String s1 = s.substring(0, i);
      String s2 = s.substring(i+1);
      s = s1 + s2;
    }
  }

  return s;
}
```

stripSpaces.pde