NAME
gv_ruby - graph manipulation in ruby

SYNOPSIS
#!/usr/bin/ruby
require 'gv'

USAGE
INTRODUCTION
gv_ruby is a dynamically loaded extension for **ruby** that provides access to the graph facilities of **graphviz.**

COMMANDS
New graphs

New empty graph
> *graph_handle* **Gv.graph** *(name);*
> *graph_handle* **Gv.digraph** *(name);*
> *graph_handle* **Gv.strictgraph** *(name);*
> *graph_handle* **Gv.strictdigraph** *(name);*

New graph from a dot-syntax string or file
> *graph_handle* **Gv.readstring** *(string);*
> *graph_handle* **Gv.read** *(string filename);*
> *graph_handle* **Gv.read** *(channel);*

Add new subgraph to existing graph
> *graph_handle* **Gv.graph** *(graph_handle, name);*

New nodes

Add new node to existing graph
> *node_handle* **Gv.node** *(graph_handle, name);*

New edges

Add new edge between existing nodes
> *edge_handle* **Gv.edge** *(tail_node_handle, head_node_handle);*

Add a new edge between an existing tail node, and a named head node which will be induced in the graph if it doesn't already exist
> *edge_handle* **Gv.edge** *(tail_node_handle, head_name);*

Add a new edge between an existing head node, and a named tail node which will be induced in the graph if it doesn't already exist
> *edge_handle* **Gv.edge** *(tail_name, head_node_handle);*

Add a new edge between named tail and head nodes which will be induced in the graph if they don't already exist
> *edge_handle* **Gv.edge** *(graph_handle, tail_name, head_name);*

Setting attribute values

Set value of named attribute of graph/node/edge - creating attribute if necessary
> *string* **Gv.setv** *(graph_handle, attr_name, attr_value);*
> *string* **Gv.setv** *(node_handle, attr_name, attr_value);*
> *string* **Gv.setv** *(edge_handle, attr_name, attr_value);*

Set value of existing attribute of graph/node/edge (using attribute handle)
> *string* **Gv.setv** *(graph_handle, attr_handle, attr_value);*
> *string* **Gv.setv** *(node_handle, attr_handle, attr_value);*

> *string* **Gv.setv** *(edge_handle, attr_handle, attr_value);*

Getting attribute values

Get value of named attribute of graph/node/edge
> *string* **Gv.getv** *(graph_handle, attr_name);*
> *string* **Gv.getv** *(node_handle, attr_name);*
> *string* **Gv.getv** *(edge_handle, attr_name);*

Get value of attribute of graph/node/edge (using attribute handle)
> *string* **Gv.getv** *(graph_handle, attr_handle);*
> *string* **Gv.getv** *(node_handle, attr_handle);*
> *string* **Gv.getv** *(edge_handle, attr_handle);*

Obtain names from handles
> *string* **Gv.nameof** *(graph_handle);*
> *string* **Gv.nameof** *(node_handle);*
> *string* **Gv.nameof** *(attr_handle);*

Find handles from names
> *graph_handle* **Gv.findsubg** *(graph_handle, name);*
> *node_handle* **Gv.findnode** *(graph_handle, name);*
> *edge_handle* **Gv.findedge** *(tail_node_handle, head_node_handle);*
> *attribute_handle* **Gv.findattr** *(graph_handle, name);*
> *attribute_handle* **Gv.findattr** *(node_handle, name);*
> *attribute_handle* **Gv.findattr** *(edge_handle, name);*

Misc graph navigators returning handles
> *node_handle* **Gv.headof** *(edge_handle);*
> *node_handle* **Gv.tailof** *(edge_handle);*
> *graph_handle* **Gv.graphof** *(graph_handle);*
> *graph_handle* **Gv.graphof** *(edge_handle);*
> *graph_handle* **Gv.graphof** *(node_handle);*
> *graph_handle* **Gv.rootof**