Subject: Re: Some petitions for GridCtrl
Posted by koldo on Tue, 06 Jul 2010 18:37:45 GMT
View Forum Message <> Reply to Message

unodgs wrote on Thu, 01 July 2010 14:35koldo wrote on Tue, 29 June 2010 05:11Hello Uno

I have some petitions to be implemented in GridCtrl.
If they are already implemented please tell me how to do it:

1. A kind of FindColumn function
Giving a column name, it returns its column index.
It could be like this:

```
int GridCtrl_FindColumn(GridCtrl &grid, String &colName) {
 for (int i = 0; i < grid.GetColumnCount(); ++i) {
  if (grid.GetColumnName(i) == colName)
   return i;
 }
 return -1;
}
```

There is only FindCol(int id). I've never needed searching by column name that's why such method doesn't exist
Quote:
2. Copy to Clipboard including column names row
At least an option to let this copy to clipboard behavior.
This way when copying a grid to an spreadsheet the column names would appear.

I will modify SetClipboard method to do that
Quote:
3. A kind of ReadCol function
A function to read a column (or part of it)
It could be like this:

```
Vector<Value> ReadCol(GridCtrl& grid, int col, int begin, int end)
{
 if (begin < 0 || end >= grid.GetRowCount() || col < 0 || col >= grid.GetColumnCount())
  throw Exc(t_("Wrong param. in ReadCol"));

 Vector<Value> v;

 for(int i = begin; i <= end; i++)
  v.Add(grid(i, col));
 return v;
}
```

There's only ReadRow. I agree having ReadCol would be useful too
Quote:

4. A kind of GetGridData/SetGridData functions
Some functions to set GridCtrl data to a Vector<Vector<Value> > and the opposite.
They could be like this:

```
Vector<Vector<Value> > GetGridData(GridCtrl& grid) {
 Vector<Vector<Value> > data;

 for (int row = 0; row < grid.GetRowCount()+1; ++row)
  data.Add(grid.ReadRow(row));

 return data;
}
void SetGridData(GridCtrl& grid, Vector<Vector<Value> > &data) {
 grid.Clear(true);
 if (!data.IsEmpty()) {
  int nrow = data.GetCount();
  int ncol = data[0].GetCount();
  for (int col = 0; col < data[0].GetCount(); ++col)
   grid.AddColumn(data[0][col]);
  grid.SetRowCount(data.GetCount()-1);
  for (int row = 0; row < grid.GetRowCount(); ++row)
   for (int col = 0; col < grid.GetColumnCount(); ++col)
    grid(row, col) = data[row+1][col];
 }
}
```

I'm not sure if these methods should be part of grid api. I'll think about that.
Quote:
5. Xmlize
A simple implementation could be like this

```
template <> void Xmlize(XmlIO xml, GridCtrl& r) {
 Vector<Vector<Value> > data;

 if(xml.IsLoading()) {
  xml("data", data);
  SetGridData(r, data);
 } else {
  data = GetGridData(r);
  xml("data", data);
 }
}
```

Yes, easy loading/saving from/to xml would be nice.

I'll try to add most of requested functionality in a few days.

Thank you