Here is a synopsis of the tutorial examples presented in this set. They have been designed for learning the PG language and its MathObjects extension — they are not necessarily the best questions to use for mathematics instruction.

1. Standard example: One part asks for a numerical answer and the other requires a functional expression. Many problems for a precalculus or calculus course are constructed using only these simple tasks (see below for a variety of techniques to use graphs). *MathObjects* simplify this problem considerably, particularly the second question asking for a derivative.

2. Hello ... : Illustrates the basic structure of a PG problem with a text answer. Modifications using MathObjects don't simplify this problem much, but they provide for better student feedback when the word answer is incorrect.

3. Multiple Choice — Perl list: Uses lists (arrays) to implement a multiple choice question. This uses "barebones" Perl and a few basic macros to create the problem. It's the assembly language version of the question.

4. Multiple Choice — radio button: Uses the multiple choice object to implement a multiple choice question — answer specified by selecting a radio button.

5. Multiple Choice — checkbox: Allows more than one correct answer to be supplied — answer(s) specified with checkboxes, all-or-nothing grading.

6. True/False — text entry: Uses the TF context to allow text answer to be single letter (T or F) or full word (True or False).

7. True/False — Pop-up choice: Answers are chosen from pop-up lists.

8. Matching lists — text entry:

9. Matching lists — popup choice: Two column display on screen

10. Matching lists — static graphs: Function given graphically; match formulas for transformations with their graph. (Example 19 is similar, but with dynamic graphs.)

11. On-the-fly graphics — 1: Figure shows function and its derivatives. Task: identify each.

12. On-the-fly-graphics — 2: Figure shows function and its derivatives — colors and labels are scrambled; same task.

13. JavaScript — 1: Estimate value of a derivative using an "oracle" function. Code is written "in clear" into the webpage.

14. JavaScript — 2: A variant of the previous example which obfuscates the example function as a cubic spline so that students can't read the javaScript code to find out the answer.

15. Java applet — 1: Applet presents graph with sliders for 2 parameters. Students can explore a task; first part asks for 3 numerical answers, but last question asks for a functional relationship.

16. Java applet — 2: Use applet to find map coordinates, then do computation of distances. Note: examples of HTML links are included here.

17. Java applet — 3: mild variant of preceding — cities chosen randomly, help about the applet appears below the questions

18. Conditional Questions: Illustrates one method to create a problem which first asks an easy question, and once that has been answered correctly, follows up with a more involved question on the same material.

19. Slope Field: Match various differential equations with their slope fields — dynamically generatated graphs (for ODEs with mildly varying parameters).

20. Spline fits to data: Demonstration of some numerical and graphical macros.

Edited by Dick Lane 2011