

Creating Compatible Electronic Exam Questions -

Ensure that what you see is what you get.

Giovanni D. Lorusso, MD

**Department of Pathology, Louisiana State University Health Sciences Center, New Orleans, LA
and Veterans' Affairs Medical Center, New Orleans, LA**

Abstract:**Purpose:**

The purpose of this paper is to create guidelines to help examination authors generate test questions in a format compatible with a wide range of software with minimal deviation from the appearance of the original document and minimal reformatting required on the part of the person compiling the exam.

Procedures:

The author draws upon eight years of experience directing a sophomore medical pathology course to a class averaging 175 students with approximately 20 lecturers and 15 additional laboratory instructors. The author has compiled a summary of pitfalls encountered when pasting electronically submitted questions from different authors into a single document for an exam.

Findings:

When compiling electronic files from multiple authors into a single examination document, many stylistic specifications of the original document are lost and often extraneous characters can be inserted. These problems can occur even if all authors use the same program to write their questions if certain guidelines are not followed.

Conclusions:

The most widely compatible format for electronic documents is a pure text file. When questions are created and submitted, they should have no special style, characters, or margin designations. These guidelines will help create "ASCII compliant" documents. The goal is to create a document that will be compatible with virtually any program that may be used to generate a test and have the end product look as similar to the original document as possible.

Key Words:

ASCII, cut and paste, electronic exam question, export, formatting, plain text, style guide

INTRODUCTION:

Many courses involve teaching by multiple faculty members. It is preferable that each lecturer submit questions relating to the material they presented. Prior to the desktop computer, most lecturers would handwrite, type, or dictate questions and a secretary would type the questions into the final paper exam format.

Many course directors now accept questions submitted to them electronically via computer disk or email. Theoretically these electronic questions can be “cut and pasted” together to create the final version of the exam. If, however, the lecturers writing the questions do not all use the same computer hardware and software, the questions submitted may have to be converted or manipulated after being pasted into the exam document. Manipulation of the electronic questions may be needed even if all authors use identical software. Cutting and pasting from one file to another is essentially exporting data from the original file and importing it into a second file. Whenever data is moved from one file to another, unexpected results may occur. This is especially true when different programs, different operating systems, or different computer platforms are involved. Even if all authors use the same program to write test questions, problems may still occur when pasting these questions into one exam document. If the individual authors use fonts or margin settings that are not used in the compiled exam, the pasted questions will look very different from their intended appearance. For example, the responses may be misaligned and need to be fixed manually. The author has encountered numerous problems converting questions between a wide range of documents. Some require laborious manipulation to correct. This article presents guidelines for lecturers (exam item creators) that will minimize the likelihood of problems when creating and transporting electronic documents.

DISCUSSION:

When an author creates a composition, he has a definite concept of how the final product should look. Many exam authors go to great pains to tediously format their questions so that they are visually appealing or have a specific look. Unfortunately when items are pasted from one program to another much of the formatting is lost; indeed some well intentioned formatting can actually be transformed into unintelligible gibberish that requires laborious manual removal. Whenever an electronic document is transposed from one computer

program to another, the omission of formatting elements or introduction of extraneous characters is possible. Many of us have opened emails to find each line preceded by strange characters such as ">" or "□." This is an example of one program (the email viewer) misinterpreting characters from the original document.

A detailed technical description is beyond the scope of this article, but a brief explanation of how computers deal with text and formatting elements is required to understand why these conversion problems occur and how they can be eliminated or at least minimized. A basic principal to keep in mind is that computers only understand numbers. Computers use a numeric code to represent keyboard characters. The American Standard Code for Information Interchange (ASCII - pronounced "ask-key") specifies the computer language codes for the characters. An "ASCII-text" file contains only the actual text of a document without formatting elements such as underline, bold, italic, or tabs. The terms "ASCII" file and "plain text" file are essentially synonymous and are the most compatible format of a word processing document. Virtually all commercial available applications will recognize an ASCII format file. More importantly, because it is a standard, the ASCII character will mean the same thing to any program that follows the ASCII standard. ASCII was developed in the mid 1960's, and does not include specifications for special formatting characteristics. As software developers created programs with elaborate formatting options, they developed their own codes for these special functions. Because these codes were developed independently by software vendors and are specific to each companies' software, the formatting elements do not mean the same thing to all software programs. When a formatting code meaning one thing to one program is pasted into a second program which uses the same code to mean something else, a substitution occurs. When a code not recognized by a program is encountered, a default character is inserted and appears to be nonsensical.

Pure text characters such as letters and numbers usually translate from one file or program to another with high reliability, but formatting elements such as margin settings, tabs, blank spaces, font designations, bold, italic, and other styles can be lost or transformed into extraneous elements. In general, a text file is the most compatible format for any word processor, database, or commercial exam package to import. The author recommends writing exam questions using only those elements conforming to the ASCII style. In general, if a document can be converted to a text only file with no visual changes, it will faithfully transfer into most applications.

The most common problem encountered when dealing with exam questions involves the alignment of the questions and responses. Some authors will align their responses by using tabs; some align via the space bar, and some by using hard returns or setting margins or justifications. Any of these methods is fine if it is the only method used throughout the exam and the exam is created using the same program and typestyle the author used to create the original question. The problem comes when several questions using these different techniques are submitted for use on the same exam. When the exam is assembled and a single font and margin designation are applied to the entire document, the questions formatted by the various different methods will be affected differently. The questions will become misaligned, and the results will be visually unappealing and possibly quite confusing. The problems are compounded when several different authors attempt to format using different techniques. When the different questions are pasted into one document, they may contain many extraneous elements that have to be tediously removed. A standardized format to ensure compatibility is needed when receiving material from multiple authors created on different computer systems. The goal is for the question to look the same in the final version of the exam as it does when it is written. This is the concept of "What You See Is What You Get" (WYSIWYG - pronounced wizzie-wig).

The following "question" and responses are formatted using techniques that might be used to create a real test item. Obviously an author would not use all techniques in one question, but these are examples of different methods that might be found in different questions by different authors submitted for use on the same exam.

Question formatted by author as it is intended to look:

This “*question*” was created in Arial font, 12pt. Notice that the right margin is justified and that there is a “hanging indent” on the left margin. Special formatting codes including “*italic*,” “**bold**,” and “underline” are used. Special characters like the \neq , \geq , \leq , \uparrow , Δ , \leftarrow , $\sqrt{\quad}$, $\text{\textcircled{R}}$, and $\frac{3}{4}$ are included just for illustration.

- A. This response was positioned by using the “Tab” key.
- B. This response was positioned by hitting the space bar.
- C. This response was positioned by setting the margins.
- D. This response contains “run-over” text (text on more than one line) At the end of each line a “Hard Return” was created to force the run-over text onto the next line; then the “Tab” key was pressed to align each line up with the one above it.
- E. This response also contains “run-over” text, but this statement was formatted by setting the margins to create a “hanging indent.” No hard return was created at the end of the line.

This is the identical question as above. Here’s how it looks after conversion to a text file:

This “question” was created in Arial font, 12pt. Notice that the right margin is justified and that there is a “hanging indent” on the left margin. Special formatting codes including “*italic*,” “**bold**,” and “underline” are used. Special characters like the \neq , \geq , \leq , \uparrow , Δ , \leftarrow , $\sqrt{\quad}$, $\text{\textcircled{R}}$, and $\frac{3}{4}$ are included just for illustration.

- A. This response was positioned by using the “Tab” key.
- B. This response was positioned by hitting the space bar.
- C. This response was positioned by setting the margins.
- D. This response contains “run-over” text (text on more than one line) At the end of each line a “Hard Return” was created to force the run-over text onto the next line; then the “Tab” key was pressed to align each line up with the one above it.
- E. This response also contains “run-over” text, but this statement was formatted by setting the margins to create a “hanging indent.” No hard return was created at the end of the line.

Notice the striking difference in how the conversion to text was handled in each line; try to imagine manually reformatting an entire exam of these questions!

The following guide helps exam creators prepare questions in the most compatible format possible and with a uniform style throughout the exam. If authors follow these guidelines, there should be minimal clerical work required to assemble the final version of the exam.

GUIDELINES FOR CREATING ELECTRONIC QUESTIONS:

IN GENERAL:

Turn OFF all auto formatting, automatic numbering, and automatic bullets.

Do not use indents or any manual or automatic margin features.

Do not justify any margins.

Don't use fancy fonts: Times or Times New Roman are preferred.

FOR QUESTION STEM*:

Do not number your questions: The numbers will have to be manually removed when the exam is compiled.

Do not use any spaces before the stem.

Do not use a "tab" before the stem.

Do not indent run over lines.

FOR RESPONSES (Choices):**

Do not use a "tab" before the response labels***.

Do not use any spaces before the response labels.

Use capital letters followed by a period and one space for the response labels.

Don't capitalize the response statements unless they are a proper noun or a complete sentence.

Do not press "return" (hard return[†]) at the end of a line unless it is the end of a response.

Do not indent run over lines.

Use only soft returns[§] within any individual response (Do not press return key)

Press return 1 time (hard return) at the end of each response.

After the last response, hard return 1 time (do not use an extra hard return between questions).

Do not use any page breaks in the document.

Sample Question:

This is what a submitted question should look like. Notice the font is basic and there are no special features. If this question is copied and pasted into virtually any program, it should look identical to this original.

- A. There is one hard return after the question stem.
- B. There is one hard return after each response.
- C. The responses are labeled by capital letters followed by a period then one space.
- D. The responses are capitalized because they are sentences.
- E. No page breaks would be used between questions.

Definitions:

* Stem = the question itself

** Response = possible answers or choices to the question

*** Response label = the "A.", "B.", "C.", etc preceding the text of the response

‡ Hard return = pressing the "return" key

§ Soft return = letting the text automatically wrap from one line to the next without pressing the return key.

Learning Points:

When creating electronic questions that will be pasted into an exam document:

- Do not use special formatting elements such as bold, italic or underline.
- Turn OFF all auto formatting, automatic numbering, and automatic bullets.
- Do not use “tabs”, indents or any manual or automatic margin features. Do not justify any margins.
- Don't use fancy fonts: Times or Times New Roman are preferred.

Giovanni D. Lorusso, MD

Department of Pathology, Box P5-1

LSU Health Sciences Center

1901 Perdido Street

New Orleans, Louisiana 70112

Phone: (504) 568-6070

FAX: 504.568.6037

email: glorus@lsuhsc.edu

12-22-2003