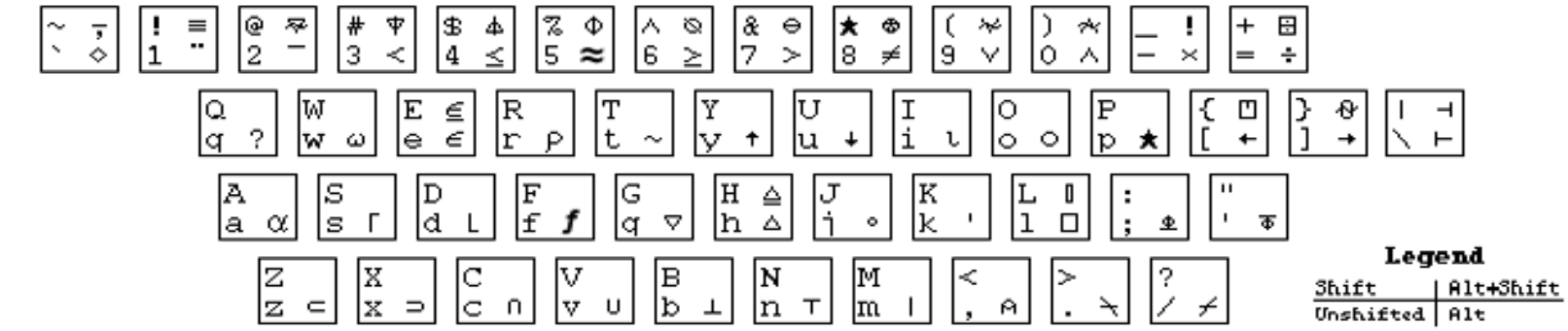
VisualAPL Keyboard – US English

The VisualAPL keyboard is illustrated below. VisualAPL uses some special symbols which are accessed using the “Alt+” and “Alt+Shift+” keystroke prefixes. These special symbols provide unique features and benefits for VisualAPL programmers who take the time to learn about their use. Reference documentation installed along with VisualAPL provides complete details of the use of these special symbols.



Unless special keyboard keytops or ‘stickers’ are used, the glyphs associated with the VisualAPL “Alt+” and “Alt+Shift+” keystroke prefixes will not be visible on the keyboard.

The designation of these VisualAPL keyboard shortcuts might conflict with keyboard shortcuts selected by vendors of other Visual Studio-related software or keyboard layouts for other languages. The “Visual Studio > Tools > Customize > Keyboard” dialog my be used by the programmer to resolve these issues according to their own needs.

Programmers familiar with legacy APL implementations should note:

- The "**Alt+f**" keystroke displays the VisualAPL ***function*** glyph "***ƒ***" (Unicode 402). This glyph is equivalent to the Visual Studio "function" keyword and used to define a VisualAPL function (with the "public" attribute) which can inherently inter-operate with any .Net language.

- The "**Shift+-**" (i.e. Shift+minus) keystroke displays the VisualAPL **underscore** glyph "**_**" (Unicode 95). In some legacy implementations of APL, this glyph was duplicatively accessible by the "Alt+f" keystroke.

- The "**Alt+g**" keystroke displays the VisualAPL ***operator*** glyph "**⍙**" (Unicode 8711). This glyph is used to define a VisualAPL operator, which is equivalent to the proprietary "function" format of legacy APL implementations. Since a VisualAPL function can call a VisualAPL operator, it is also possible to expose a VisualAPL operator to any .Net language.

- The "**Alt+5**" keystroke displays the VisualAPL ***approximately equal*** glyph "**≈**" (Unicode 8776). This glyph is used to designate comparison of two VisualAPL objects. It accurately describes the APL (both VisualAPL and legacy APL) implementation of the operator whereby two APL objects are compared and considered equal in value if they differ by not more than the programmer-controlled, comparison tolerance (⍙ct or 'fuzz'). In legacy APL implementations, the equal "**=**" operator was used instead of the approximately equal "**≈**" operator.

- The "**=**" keystroke displays the VisualAPL (and Visual Studio) ***assignment by reference*** glyph "**=**" (Unicode 61). This glyph indicates that an object is assigned another object's value by reference, rather than by value, so that subsequent modification of the source object will also cause the assigned object to be analogously modified.

- The "**Alt+[**" keystroke displays the VisualAPL ***assignment by value*** glyph "**←**" (Unicode 8592). This glyph indicates that an object is assigned another object's value and subsequent modification of the source object will not cause the assigned object to be modifed.

- The "**Alt+,**" keystroke traditionally displays the APL ***comment*** (or lamp) glyph "**⍝**" (Unicode 9053 – may not be visible in this pdf-format document). This glyph indicates that what follows on the current line is a non-executable comment for program clarification or notation. By default this keystroke is not assigned in VisualAPL because it conflicts with a pre-existing Visual Studio keystroke assignment. If desired, the VisualAPL programmer may utilize the Visual Studio standard "//" keystrokes to indicate a comment, or may use the "Visual Studio > Tools > Customize > Keyboard" dialog to establish the "Alt+," keyboard shortcut for the APL comment glyph.