

[Galaxy_TOC](#)

Programmer Friendly text Exchange (ProgFTE)

The Programmer Friendly text Exchange (hereafter: ProgFTE) is a [structured text format](#) for serialization and deserialization of hashtables that contain only strings as keys and values.

Its main difference from many other structured text formats, like XML, JSON, etc., is that the libraries of the other formats are laborious to implement, if implemented correctly, but the ProgFTE support can be added to many experimental or non-mainstream, programming languages with relatively small effort.

Similar Solutions

People, who are interested in the ProgFTE, might find the [ASN.1](#) interesting. The ASN.1 is part of the European Space Agency [TASTE](#) project.

ProgFTE Specification Version 0 (ProgFTE_v0)

ProgFTE_v0 strings are encoded in UTF-8.

Version 0 of the ProgFTE specification contains a flaw and resides at <http://mv-veebilog.blogspot.com/2009/10/programmer-friendly-text-exchange.html>

The ProgFTE_v0 strings can be distinguished from all strings that conform to some later ProgFTE specification by a fact that ProgFTE_v0 strings start with a digit, but strings that conform to later ProgFTE specifications start with a character "v".

ProgFTE Specification Version 1 (ProgFTE_v1)

ProgFTE_v1 strings are encoded in UTF-8.

If [non-terminal](#) symbols are written by enclosing text in [chevrons](#), then a regular expression of ProgFTE_v1 string looks like:

```
v<ProgFTE_format_version>[<ProgFTE_format_mode>[<number_of_key-value_pairs>[<key-value_pair>]+
```

where:

```
<si_positive_integer> ::= 0|[123456789][\d]*
<ProgFTE_format_version> ::= 1
<ProgFTE_format_mode> ::= <si_positive_integer>
<number_of_key-value_pairs> ::= <si_positive_integer>
<key-value_pair> ::= <key_stringrecord><value_stringrecord>
<key_stringrecord> ::= <stringrecord>
```

```
<value_stringrecord> ::= <stringrecord>
Galaxy\_TOC
<stringrecord> ::= <si_string_length>[[]].*[[[]]
<si_string_length> ::= <si_positive_integer>
```

The very first key-value pair is reserved for ProgFTE string related meta-data and it belongs to an automatically created hashtable that always has exactly one key-value pair.

The format of the metadata is determined by the ProgFTE_format_mode and the purpose of the metadata is to allow more advanced ProgFTE implementations to decode ProgFTE strings faster than the default decoding algorithm allows them to do. If the speed is not an issue, then all ProgFTE libraries that support the ProgFTE_v1, are able to decode all ProgFTE_v1 strings, regardless of the value of the ProgFTE_format_mode.

The key of the metadata key-value pair is an empty string, regardless of the value of the ProgFTE_format_mode. If the value of the meta-data key-value pair is an empty string, then the ProgFTE_format_mode==0. Metadata formats for cases, where the ProgFTE_format_mode!=0, will be described as part of this specification.