



## TelFormFactory Datatypes

### Namespace

It may be noticed that some of the Datatype names are prefixed by the 'colon-ised' 'tag' - 'xsd' and others are prefixed by 'tel'. A tag prefix is an abbreviation for the namespace of the Datatype: in this case xsd indicates the namespace of the W3C Recommendation -<http://www.w3.org/2001/XMLSchema> and tel stands for the namespace of <http://telform.net/ns/2009/telform>. Namespace is an important feature of XML. It may be thought of as referring to a dictionary that the XML Document refers to when decoding the document. It is a powerful feature but can cause problems - it means an XML phrase may be connected with other XML documents the other side of the world but which may not understand other phrases in the same paragraph. In theory any prefix can be used to refer to a namespace providing it has been defined in the document, but this should not be relied upon.

Datatype name Description tel:TelFormRootType The basis on which the TelForm is formed. xsd:string The string datatype represents character strings in XML. tel:EmailFieldType EmailFieldType is checked for valid Email address format. tel:PhoneFieldType Telephone number field. xsd:integer integer is derived from decimal by fixing the value of fractionDigits to be 0 and disallowing the trailing decimal point. xsd:decimal decimal represents a subset of the real numbers, which can be represented by decimal numerals. tel:MultipleChoiceType Allows a number of options to be chosen from a range of values. tel:MultipleChoiceItemtype One of several MultipleChoiceType options. tel:ExclusiveChoiceType Allows a single option to be chosen from a range of values. tel:ExclusiveChoiceItemtype One of several ExclusiveChoiceType options. tel:GROUP A sub-group of fields may be attached to this node. tel:RegEX A holder for a datatype which contains a Regular Expression (RegEx). See link below. xsd:boolean boolean has the value space required to support the mathematical concept of binary-valued logic: {true, false}. xsd:float float is patterned after the IEEE single-precision 32-bit floating point type [IEEE 754-1985]. xsd:double The double datatype is patterned after the IEEE double-precision 64-bit floating point type [IEEE 754-1985]. xsd:duration duration represents a duration of time. xsd:dateTime dateTime values may be viewed as objects with integer-valued year, month, day, hour and minute properties, a decimal-valued second property, and a boolean timezoned property. xsd:time time represents an instant of time that recurs every day. xsd:date The value space of date consists of top-open intervals of exactly one day in length. xsd:gYearMonth gYearMonth represents a specific gregorian month in a specific gregorian year. xsd:gYear gYear represents a gregorian calendar year. xsd:gMonthDay gMonthDay is a gregorian date that recurs, specifically a day of the year such as the third of May. xsd:gDay gDay is a gregorian day that recurs, specifically a day of the month such as the 5th of the month. xsd:gMonth gMonth is a gregorian month that recurs every year. xsd:hexBinary hexBinary represents arbitrary hex-encoded binary data. xsd:base64Binary base64Binary represents Base64-encoded arbitrary binary data. xsd:anyURI anyURI represents a Uniform Resource Identifier Reference (URI). xsd:QName QName represents XML qualified names. xsd:NOTATION NOTATION represents the NOTATION attribute type from [XML 1.0 (Second Edition)]. xsd:normalizedString normalizedString represents

white space normalized strings. xsd:token token represents tokenized strings. xsd:language language represents natural language identifiers as defined by by [RFC 3066]. xsd:NMTOKEN NMTOKEN represents the NMTOKEN attribute type from [XML 1.0 (Second Edition)]. xsd:NMTOKENS NMTOKENS represents the NMTOKENS attribute type from [XML 1.0 (Second Edition)]. xsd:Name Name represents XML Names. xsd:NCName NCName represents XML 'non-colonized' Names. xsd:ID ID represents the ID attribute type from [XML 1.0 (Second Edition)]. xsd:IDREF IDREF represents the IDREF attribute type from [XML 1.0 (Second Edition)]. xsd:IDREFS IDREFS represents the IDREFS attribute type from [XML 1.0 (Second Edition)]. xsd:ENTITY ENTITY represents the ENTITY attribute type from [XML 1.0 (Second Edition)]. xsd:ENTITIES ENTITIES represents the ENTITIES attribute type from [XML 1.0 (Second Edition)]. xsd:nonPositiveInteger nonPositiveInteger is derived from integer by setting the value of maxInclusive to be 0. xsd:negativeInteger negativeInteger is derived from nonPositiveInteger by setting the value of maxInclusive to be -1. xsd:long long is derived from integer by setting the value of maxInclusive to be 9223372036854775807 and minInclusive to be -9223372036854775808. xsd:int int is derived from long by setting the value of maxInclusive to be 2147483647 and minInclusive to be -2147483648. xsd:short short is derived from int by setting the value of maxInclusive to be 32767 and minInclusive to be -32768. xsd:byte byte is derived from short by setting the value of maxInclusive to be 127 and minInclusive to be -128. xsd:nonNegativeInteger nonNegativeInteger is derived from integer by setting the value of minInclusive to be 0. xsd:unsignedLong unsignedLong is derived from nonNegativeInteger by setting the value of maxInclusive to be 18446744073709551615. xsd:unsignedInt unsignedInt is derived from unsignedLong by setting the value of maxInclusive to be 4294967295. xsd:unsignedShort unsignedShort is derived from unsignedInt by setting the value of maxInclusive to be 65535. xsd:unsignedByte unsignedByte is derived from unsignedShort by setting the value of maxInclusive to be 255. xsd:positiveInteger positiveInteger is derived from nonNegativeInteger by setting the value of minInclusive to be 1.

### **See XML Schema Regular Expressions (RegEx)**

Copyright © terry-comms 2003-2011 version-20111223 : 1437 |