

# Raccomandazioni sui formati per i dati aperti

Giuseppe Ascone Modica

*Ufficio delle Pubblicazioni dell'Unione Europea*

# Che cos'è data.europa.eu?

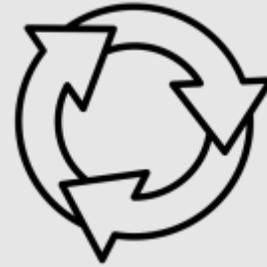
data.europa.eu è un catalogo di metadati che fornisce un unico punto di accesso ai dati aperti dei paesi europei e delle istituzioni dell'UE per il loro riutilizzo.



# Perché la qualità dei dati è importante

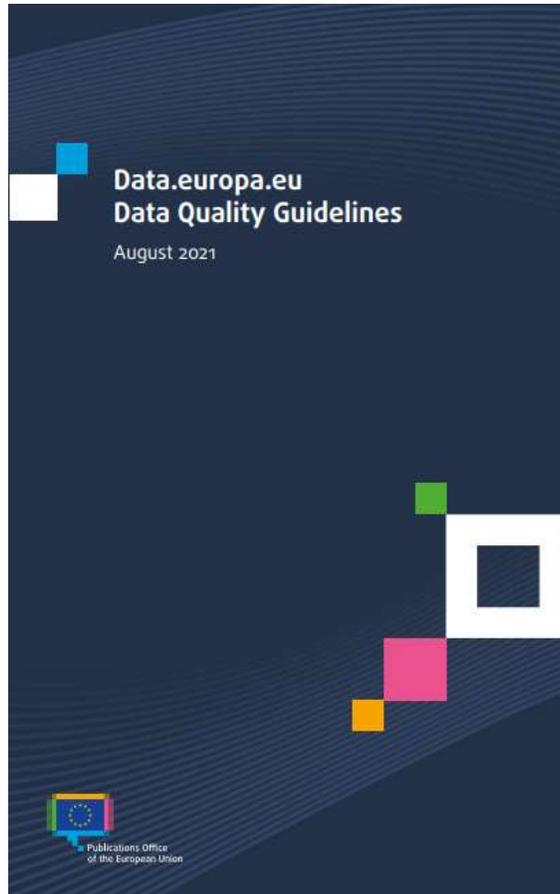


**Processo decisionale  
informato**



**Riuso dei dati**

# Le linee guida per la qualità dei dati



Progetto lanciato dall'Ufficio delle pubblicazioni dell'Unione europea nel 2019.

**Obiettivo:** analizzare i principali problemi di qualità e fornire una serie di raccomandazioni per i fornitori di dati dell'UE e dei suoi Stati membri.

Composto da 3 parti:

1. Analisi dei dati per identificare i problemi di qualità più comuni
2. Identificazione delle dimensioni, degli indicatori e delle metriche della qualità dei dati.
3. **Raccomandazioni per fornire dati di alta qualità**

<https://op.europa.eu/it/publication-detail/-/publication/b601d9cc-b3c0-11ec-9d96-01aa75ed71a1>

# Le linee guida per la qualità dei dati



1.2. Format-specific recommendations .....	29
1.2.1. CSV .....	29
1.2.1.1. Use a semicolon as a delimiter .....	29
1.2.1.2. Use one file per table .....	30
1.2.1.3. Avoid white space and additional information in the file .....	31
1.2.1.4. Insert column headers .....	34
1.2.1.5. Ensure that all rows have the same number of columns .....	36
1.2.1.6. Indicate units in an easily processable way .....	37
1.2.2. XML .....	38
1.2.2.1. Provide an XML declaration .....	38
1.2.2.2. Escape special characters .....	38
1.2.2.3. Use meaningful names for identifiers .....	40
1.2.2.4. Use attributes and elements correctly .....	41
1.2.2.5. Remove program-specific data .....	42
1.2.3. RDF .....	42
1.2.3.1. Use HTTP URIs to denote resources .....	42
1.2.3.2. Use namespaces when possible .....	43
1.2.3.3. Use existing vocabularies when possible .....	44
1.2.4. JSON .....	45
1.2.4.1. Use suitable data types .....	45
1.2.4.2. Use hierarchies for grouping data .....	46
1.2.4.3. Only use arrays when required .....	47

Questa presentazione analizzerà le raccomandazioni specifiche per ciascun formato di dati aperti identificati dalle linee guida sulla qualità dei dati.

I formati sono i seguenti:

- CSV
- XML
- RDF
- JSON

# Raccomandazioni specifiche per i formati – CSV

1. Usare punto e virgola al posto delle virgole.
2. Assicurarsi che non ci siano spazi o tabulazioni su entrambi i lati dei delimitatori nella riga.
3. Ogni file CSV deve contenere solo una tabella. Se la tabella da pubblicare è composta da più fogli, è necessario creare un file CSV per ogni foglio.
4. Evitare spazi bianchi e informazioni aggiuntive nel file.



Bad example



Good example

Year; Visitors; Viewing time;

Year; Visitors; Viewing time

2013; 822101;00:02:59;

2013;822101;00:02:59

2012;792967;00:02:52;

2012;792967;00:02:52

2011; 721519;00:03:44;

2011;721519;00:03:44

2010;707402;00:03:50;

2010;707402;00:03:50

2009;429430;00:03:16;

2009;429430;00:03:16



Table: Smallpox, surveillance systems overview, 2014

Colouring		Title				Data reported by				Case definition used
Country	Data source	L	P	H	O	reported by laboratories	reported by physicians	reported by hospitals	reported by others	Case definition used
Austria	AT-Epidemegesetz	Y	Y	Y	Y	Y	Y	Y	Y	Y
Belgium	BE-FLA_FRA	Y	Y	Y	Y	Y	Y	Y	Y	Y
Bulgaria	BG-NATIONAL_SURVEILLANCE	Y	Y	Y	Y	Y	Y	Y	Y	Y
Cyprus	CY-NOTIFIED_DISEASES	N	Y	N	N	Y	Y	Y	Y	Y

Data reported by: laboratories (L), physicians (P), hospitals (H), other (O)

Suggested citation: European Centre for Disease Prevention and Control. Annual epidemiological report 2016. Smallpox. Stockholm: ECDC; 2016. Reproduction is authorised, provided the source is acknowledged



Country	Data source	reported by laboratories	reported by physicians	reported by hospitals	reported by others	Case definition used
Austria	AT-Epidemegesetz	Y	Y	Y	Y	Y
Belgium	BE-FLA_FRA	Y	Y	Y	Y	Y
Bulgaria	BG-NATIONAL_SURVEILLANCE	Y	Y	Y	Y	Y
Cyprus	CY-NOTIFIED_DISEASES	N	Y	N	N	Y

# Raccomandazioni specifiche per i formati – CSV

5. Inserire intestazioni di colonna.
6. Assicurarsi che tutte le righe abbiano lo stesso numero di colonne.
7. Indicare le unità in modo facilmente elaborabile.

## Link e strumenti utili

<https://csvlint.io/>

CSVLint è uno strumento online che identifica spazi vuoti, numero di colonne per riga, righe di titolo

<https://frictionlessdata.io/tooling/goodtables/#a-simple-example>

GoodTables è uno strumento per la convalida dei dati e controlla, ad esempio, se tutte le righe hanno lo stesso numero di colonne

 <b>Bad example</b>		 <b>Good example</b>		
Ingredient	Amount	Ingredient	Amount	Unit
Carbohydrates	16g	Carbohydrates	16	g
Magnesium	2mg	Magnesium	20	mg

 <b>Better example</b>		
Ingredient	Amount	Unit
Carbohydrates	16	<http://publications.europa.eu/resource/authority/measurement-unit/GRM>
Magnesium	20	<http://publications.europa.eu/resource/authority/measurement-unit/MGM>

# Raccomandazioni specifiche per i formati– XML

1. Fornire una dichiarazione di tipo XML.
2. Fare l'escape di caratteri speciali.
3. Utilizzare nomi significativi per gli identificatori.
  - camelCase o PascalCase

Formato escaped	Sostituito da
&amp;	&
&lt;	<
&gt;	>
&quot;	"
&apos;	'

## ✗ Bad example

This screenshot shows an XML without a declaration.

```
<fruits>
  <fruit>
    <type>Apple</type>
    <origin>Germany</origin>
    <drupe>true</drupe>
  </fruit>
</fruits>
```

## ✓ Good example

This screenshot shows the same XML with a properly formatted declaration.

```
<?xml version="1.0" encoding="UTF-8"?>
<fruits>
  <fruit>
    <type>Apple</type>
    <origin>Germany</origin>
    <drupe>true</drupe>
  </fruit>
  <fruit>
    <type>Grape</type>
    <origin>Italy</origin>
    <drupe>>false</drupe>
  </fruit>
</fruits>
```

## ✗ Bad example

This example shows XML with the 'fairtrade' identifier (i.e. the element's name) not being written using PascalCase or camelCase, making it harder to read by humans and thus prone to processing errors.

```
<fruits>
  <type>Apple</type>
  <origin>Germany</origin>
  <drupe>true</drupe>
  <fairtrade>true</fairtrade>
</fruits>
```

## ✓ Good example

This screenshot shows XML with an identifier which consists of two words being concatenated via camelCase.

```
<fruits>
  <type>Apple</type>
  <origin>Germany</origin>
  <drupe>true</drupe>
  <fairTrade>true</fairTrade>
</fruits>
```

## ✗ Bad example

This screenshot shows an XML without escaping.

```
<fruit id="&1">
  <type>Apple <</type>
  <origin>> Germany</origin>
  <description>"Very tasty!"</description>
</fruit>
```

## ✓ Good example

This screenshot shows the same XML with properly escaped characters.

```
<fruit id="& amp;1">
  <type>Apple &lt;</type>
  <origin>&gt; Germany</origin>
  <description>&quot;Very tasty!&quot;</description>
</fruit>
```

# Raccomandazioni specifiche per i formati– XML

- Utilizzare correttamente gli attributi e gli elementi.
- Rimuovere i dati specifici del programma.

## Bad example

This screenshot shows XML in which data has been encoded using attributes where elements would have been more suitable.

```
<fruit type="apple" drupe="true" id="1">
  <origin>Germany</origin>
</fruit>
```

## Good example

This screenshot shows XML in which data and metadata have been encoded using elements and attributes correctly.

```
<fruit id="1">
  <type>Apple</type>
  <origin>Germany</origin>
  <drupe>true</drupe>
</fruit>
```

## Link e strumenti utili

<https://www.freeformatter.com/xml-escape.html>

Strumento online che consente di sfuggire ai caratteri speciali del testo in modo da poterli utilizzare in XML

<https://titlecase.com/>

Questo strumento converte le frasi composte da più parole in vari formati di caso

## Bad example

This screenshot shows XML which contains a version number of a hypothetical program that has been used for the creation or processing of the file. This information does not add anything to the data and should thus be removed.

```
<fruits>
  <fruit id="1">
    <type>Apple</type>
    <description>Very tasty</description>
  </fruit>
</fruits>
<createdWith version="1.0">myXmlTool</createdWith>
```

# Raccomandazioni specifiche per i formati – RDF

1. Utilizzare gli URI HTTP per indicare le risorse.
2. Utilizza 'namespace' (spazi dei nomi) quando possibile
  - Identificatori: PascalCase
  - Proprietà: camelCase

 **Bad example** RDF without namespaces and identifier conventions applied can be harder to read.

```
<rdf:rdf>
  <rdf:description rdf:about="http://myresource">
    <http://mynamespace#myproperty>Sample
    </http://mynamespace#myproperty>
  </rdf:description>
</rdf:rdf>
```

 **Good example** This screenshot shows the use of namespaces as well as conventions for class and property identifiers, which improves readability.

```
<rdf:RDF xmlns:myNamespace="http://myNamespace#">
  <rdf:Description rdf:about="http://myResource">

    <myNamespace:myProperty>Sample</myNamespace:myProperty>
  </rdf:Description>
</rdf:RDF>
```

 **Bad example** This screenshot shows a resource in RDF/XML which is not denoted via HTTP URI.

```
<vcard:hasAddress rdf:resource="myAddress">
```

 **Good example** This screenshot shows a resource in RDF/XML which is denoted via HTTP URI.

```
<vcard:hasAddress
  rdf:resource="http://www.w3.org/2006/vcard/ns#myAddress">
```

# Raccomandazioni specifiche per i formati – RDF

3. Utilizzare i vocabolari esistenti quando possibile (OP utilizza DCAT-AP).

## Link e strumenti utili

<https://www.ontotext.com/products/ontotext-platform/>  
Strumento che consente l'importazione di dati strutturati e la conversione in dati RDF. Durante l'importazione è possibile definire gli *namespaces*.

<https://www.cambridgesemantics.com/product/>  
Piattaforma che consente la trasformazione di dati strutturati e semistrutturati in grafici RDF.

<https://openrefine.org/>  
OpenRefine è uno strumento di raffinamento per la pulizia dei dati. Dispone di un esportatore integrato per generare file RDF.

<https://www.trifacta.com/products/wrangler-editions/#wrangler>  
Trifacta Wrangler è una suite di strumenti di preparazione dei dati. Consente la trasformazione di formati diversi, pulendo e unendo e unire i dati

## Link e strumenti utili

<https://op.europa.eu/en/web/eu-vocabularies>  
EU Vocabularies consentono di accedere ai vocabolari gestiti dalle istituzioni e dagli organi dell'UE

<https://www.cognitum.eu/semantics/Tools/SparqlExcelTools.aspx>  
Questo strumento è un plugin per Microsoft Excel 2010 e 2013 che può essere utilizzato per importare dati RDF in Excel da un endpoint SPARQL convertendo così RDF in XLS.

### ✘ Bad example

```
<dcterms:license rdf:resource="http://CC_BY_4_o"/>
```

This screenshot shows the licence of a data set referenced without using the controlled vocabulary. This makes further processing much harder and is error prone with regard to spelling.

### ✔ Good example

```
<dcterms:license rdf:resource=  
http://publications.europa.eu/resource/authority/licence/CC\_  
BY\_4\_o>
```

This screenshot shows the same licence being referenced using the controlled vocabulary published by the European Commission.

# Raccomandazioni specifiche formati – JSON

## 1. Utilizza tipo di dati adeguati:

- Null value
- Valori booleani
- Stringhe
- Numeri e sequenze semplici delle cifre (0-9)
- Elenchi (arrays)
- Oggetti



### Bad example

This screenshot shows a JSON file with various data types. All information has been encoded using strings, regardless of the underlying data type.

```
{  
  "type": "apple",  
  "fairTrade": "true",  
  "amount": "5"  
}
```



### Good example

This screenshot shows the same JSON file, this time with dedicated data types where applicable.

```
{  
  "type": "apple",  
  "fairTrade": true,  
  "amount": 5  
}
```

# Raccomandazioni specifiche formati – JSON

2. Utilizzare le gerarchie per raggruppare i dati.
3. Usare gli array solo se necessario.

## ✘ Bad example

This screenshot shows a JSON file with array usage, but it is unclear what type of nutrients the values are referring to. Dedicated fields would have been more useful in this scenario.

```
{
  "type": "apple",
  "nutrients": [6.0, 5.0, 0.0]
}
```

## ✔ Good example

This screenshot shows a JSON file in which array usage is useful.

```
{
  "type": "apple",
  "nutrients": {
    "calcium": 6.0,
    "magnesium": 5.0,
    "zinc": 0.0
  }
}
```

## ✘ Bad example

This screenshot shows a JSON file with grouped data. All information has been attached to the root object. For objects with a larger number of fields, this can quickly reduce readability.

```
{
  "type": "apple",
  "calcium": 6.0,
  "magnesium": 5.0,
  "zinc": 0.0
}
```

## ✔ Good example

The screenshot shows the same JSON file with semantically grouped data.

```
{
  "type": "apple",
  "nutrients": {
    "calcium": 6.0,
    "magnesium": 5.0,
    "zinc": 0.0
  }
}
```

## Link e strumenti utili

<https://jsonlint.com/>

Questo strumento online verifica se l'input è un JSON valido

# Grazie

Giuseppe Ascone Modica

*Ufficio delle Pubblicazioni dell'Unione Europea*

*[giuseppe.ascone-modica@publications.europa.eu](mailto:giuseppe.ascone-modica@publications.europa.eu)*