

Pattern Factory e Singleton

- Si vuole sviluppare un sistema di esportazione dati
- Al momento sono previste 2 tipologie di esportazione.
- L'interfaccia comune è **DataExport** e fornirà i metodi:
 - `void export(List<String> data, String fileName)`
 - `String getMime()`
- Dettaglio tipologie e funzionamento :
 - **CsvExporter**: unisce le stringhe in una sola stringa, separate da virgole e chiude con il ritorno a capo, infine salva il contenuto nel file indicato
 - Il MIME relativo sarà **text/cvs**
 - **JsonExporter**: crea la struttura a blocchi con parentesi graffe, invio a capo, virgolette per le proprietà chiave, infine salva il contenuto nel file indicato
 - Il MIME relativo sarà **application/json**
- **Nota:** le stringhe della lista di input devono avere un formato chiave valore: `"ID:101"`, `"Prodotto:Caffè"`, `"Prezzo:5.50"`

Scrittura nel file

- Supponiamo che `nomeFile` è la stringa col nome file dove scrivere e `testo` è la stringa che sarà il contenuto, allora devi usare il seguente blocco:

```
try {  
    BufferedWriter bf =  
        new BufferedWriter(new FileWriter(nomeFile, true));  
    bf.write(testo);  
    bf.close();  
} catch (IOException e) {  
    e.printStackTrace();  
}
```



Factory e Consumer

- Realizzare una classe **Statistiche** che recupera dei valori letti dal database e li vuole esportare in file diversi (con formati diversi) senza avere dipendenze dalle modalità di esportazione.
- Le tabelle operative su cui agisce questa classe sono 2:
 - **Category** (contiene **idCategory PK** e **description**)
 - **City** (contiene **cityName PK**, **region**)
- Una classe di **setting** (col main), propone inizialmente una lista di opzioni di export, ottenuta tramite una terza tabella **ExportType** sullo stesso db, configura la classe **Statistiche** ed esegue i suoi metodi di business
- Una classe **FactoryDataExport** infine dovrà fornire, via reflection, gli oggetti DataExport necessari alla classe Statistiche per operare.
 - Gli oggetti generati devono essere dei singleton