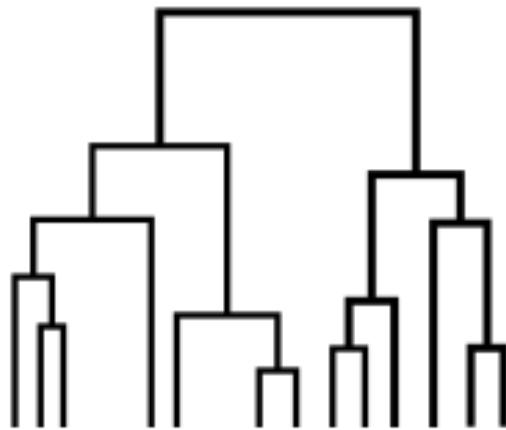


H-CLUS



Analisi strutturata dei dati con il
clustering gerarchico

Nazim Elmadhi
Giovanna Antida De Pascale

Indice

1. INTRODUZIONE

2. HIERARCHICAL CLUSTERING: INTRODUZIONE AL PROGETTO

2.1 ESTENSIONE SVILUPPATA

2.2 PROGETTAZIONE

3. DIAGRAMMI UML

3.1 SERVER VERSIONE BASE

3.2 CLIENT CONSOLE

3.3 SERVER VERSIONE ESTENSIONE

3.4 CLIENT ESTENSIONE

4 MANUALE UTENTE

4.1 REQUISITI

4.2 VERSIONE CONSOLE

4.3 VERSIONE INTERFACCIA WEB

1. INTRODUZIONE

La tecnica del **Hierarchical Clustering** è una delle tecniche di clustering più utilizzate nel data mining, in grado di creare una rappresentazione gerarchica dei cluster in un set di dati. L'algoritmo utilizzato nel progetto è il clustering gerarchico di tipo agglomerativo in cui ogni punto dati viene considerato come un cluster individuale. A ogni iterazione, vengono combinati i cluster più vicini fino a raggiungere un criterio di arresto.

Il risultato del clustering agglomerativo dipenderà dalle tecniche utilizzate per calcolare la distanza tra i cluster. I metodi impiegati per tale calcolo sono la distanza Single-Link e la distanza Average-Link. Nel metodo del Single-Link la distanza tra due cluster è definita come la distanza minima tra i punti di due cluster diversi, invece nel metodo dell'Average-Link la distanza tra due cluster è definita come la media delle distanze tra tutti i punti di un cluster e tutti i punti dell'altro cluster.

Il risultato del clustering gerarchico è una struttura ad albero, chiamata dendrogramma, che rappresenta le relazioni gerarchiche tra i cluster.

2. HIERARCHICAL CLUSTERING: INTRODUZIONE AL PROGETTO

L'algoritmo di clustering agglomerativo è stato implementato in un software Java progettato per eseguire la clusterizzazione di esempi di dati estratti da una tabella di un database. Il software è composto da due applicativi: un client e un server. In particolare, il client funge da interfaccia per l'utente, permettendo di interagire con il server tramite terminale. Inoltre, il client permette di avviare operazioni di clustering e gestire il caricamento o la lettura dei file generati. Dunque, offre funzionalità intuitive per selezionare le impostazioni desiderate, come la profondità del dendrogramma da definire, il tipo di distanza tra i cluster da utilizzare e la tabella da cui estrarre i dati. Il server è il motore vero e proprio del software in quanto può eseguire diverse richieste per i diversi client connessi. In particolare, può estrapolare dati da una tabella di un database, eseguire il

clustering su quei dati, salvarli e di conseguenza leggerli da un file. Inoltre, lo stato del server è consultabile da terminale.

2.1 ESTENSIONE SVILUPPATA

Nonostante la versione base sia completa e funzionante, per rendere il progetto accessibile a più utenti è stata sviluppata, come estensione, un'interfaccia web per il client. Questa ha reso il progetto più intuitivo e user-friendly rispetto a un sistema che richiede l'interazione attraverso un terminale. L'aspetto dell'interfaccia è semplice e pulito garantendo all'utente facilità d'uso e maggiore chiarezza e libertà di navigazione tra le varie finestre senza ambiguità e possibilità di cadere in errore. Inoltre, la comunicazione client-server è stata modificata, si è passato infatti dalla comunicazione tramite i Socket a quello tramite il protocollo Http in quanto più efficace e affidabile.

2.2 PROGETTAZIONE

L'interfaccia web è stata costruita utilizzando **Vaadin Flow**, la versione moderna di Vaadin, un framework Java per la creazione di applicazioni web con interfacce utente interattive e responsive, che genera automaticamente il codice HTML e JavaScript per il browser. Il programma utilizza anche il framework open source **Spring Boot** come base per configurare e avviare il server, integrandosi perfettamente con Vaadin. Per personalizzare l'aspetto grafico dell'interfaccia sono stati usati file .css per definire e applicare stili ai singoli elementi, e file .svg dal momento che l'interfaccia include delle icone.

Dopo aver definito con chiarezza i requisiti e valutato possibili fonti di ispirazione per la struttura dell'interfaccia grafica da sviluppare, si è proceduto alla realizzazione degli elementi costitutivi del progetto. Il lavoro è stato organizzato in modo collaborativo, con una suddivisione delle attività tra i due membri del gruppo: ciascuno si è concentrato su aspetti differenti dell'interfaccia, lavorando rispettivamente sul funzionamento e sull'aspetto grafico. Le parti sviluppate individualmente sono state integrate iterativamente, consentendo un continuo allineamento e un miglioramento progressivo dell'intero sistema.

Il progetto dell'interfaccia ha coinvolto la realizzazione di tre schermate principali. La prima schermata raffigura la home del progetto in cui è possibile scegliere il tema della pagina e include una descrizione del progetto e due opzioni per l'utente. A seconda della scelta, la pagina può indirizzarlo verso l'individuazione dei cluster da un database oppure verso la lettura di dati da un file.

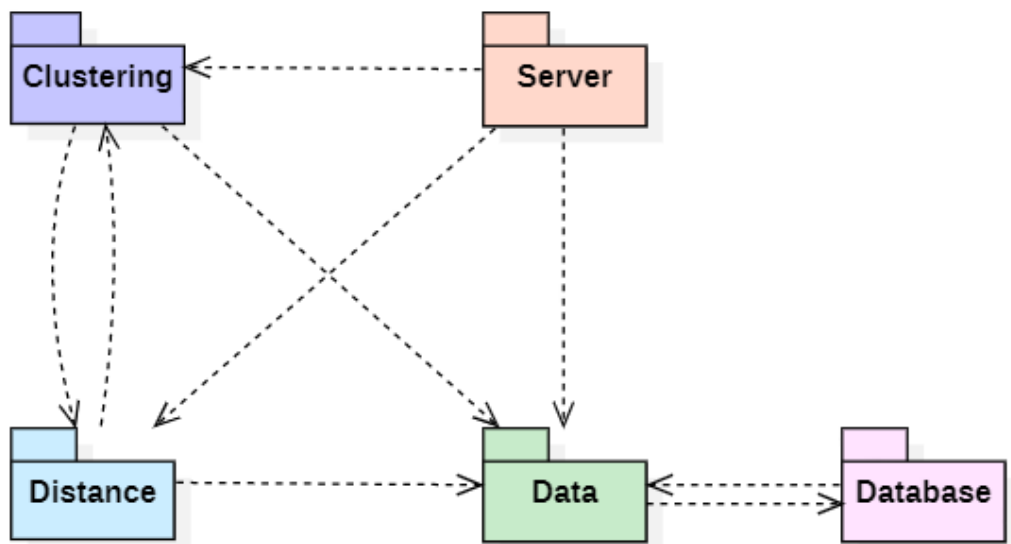
La realizzazione del progetto è visibile nella guida utente di questo documento.

3. DIAGRAMMI UML

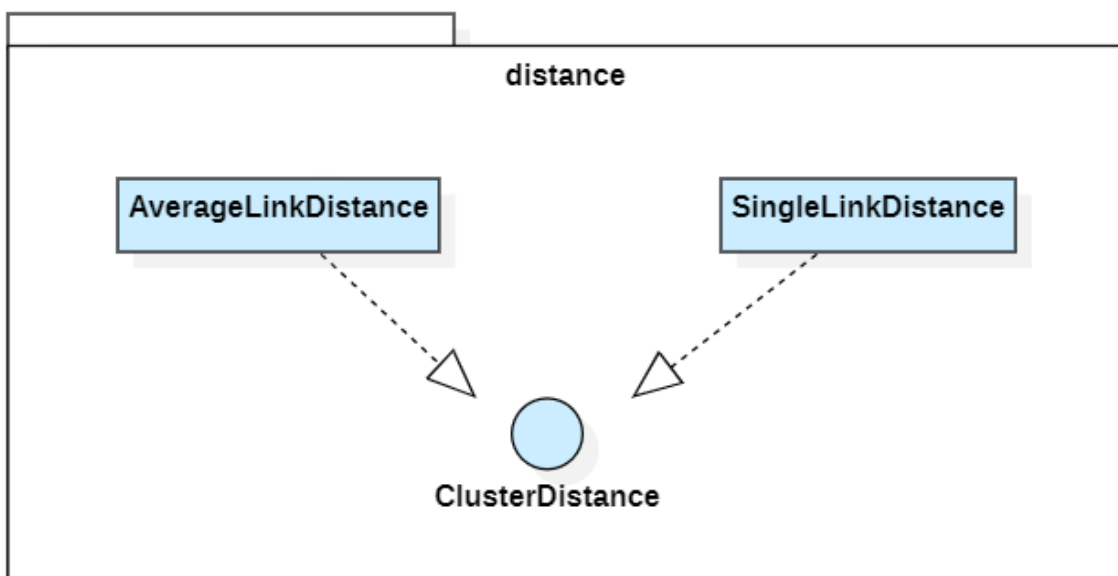
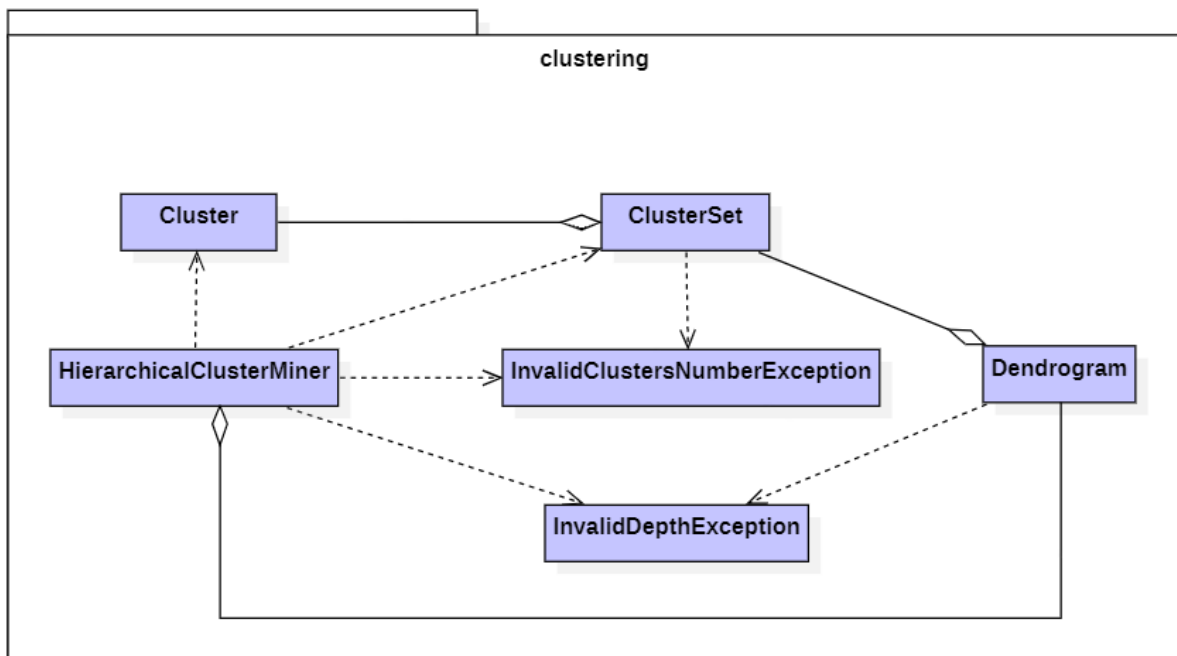
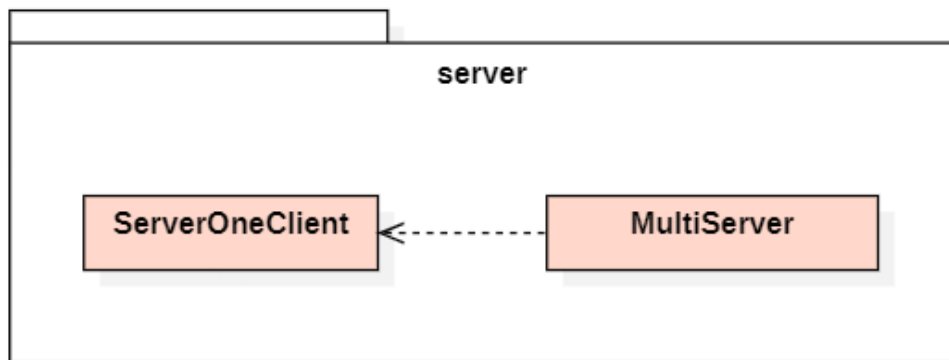
In questa sezione del documento vengono mostrati i diagrammi UML delle classi e dei package. Tutti i diagrammi sono stati realizzati utilizzando il software StarUML. Abbiamo tratto ispirazione dal libro Java Modeling Color with UML e quindi deciso di utilizzare i colori per identificare le classi di uno stesso package.

3.1 SERVER VERSIONE BASE

DIAGRAMMA DEI PACKAGE



Nello specifico:



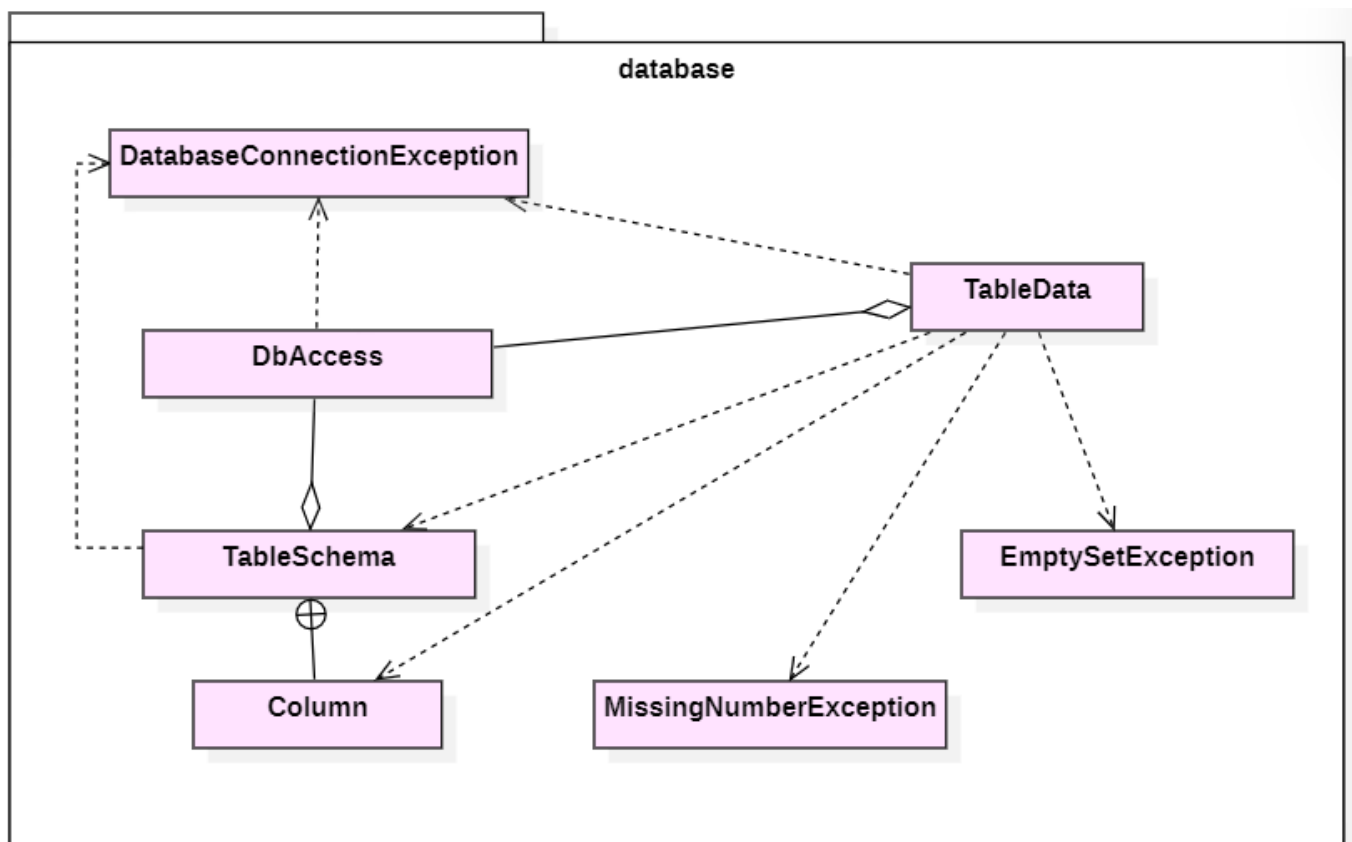
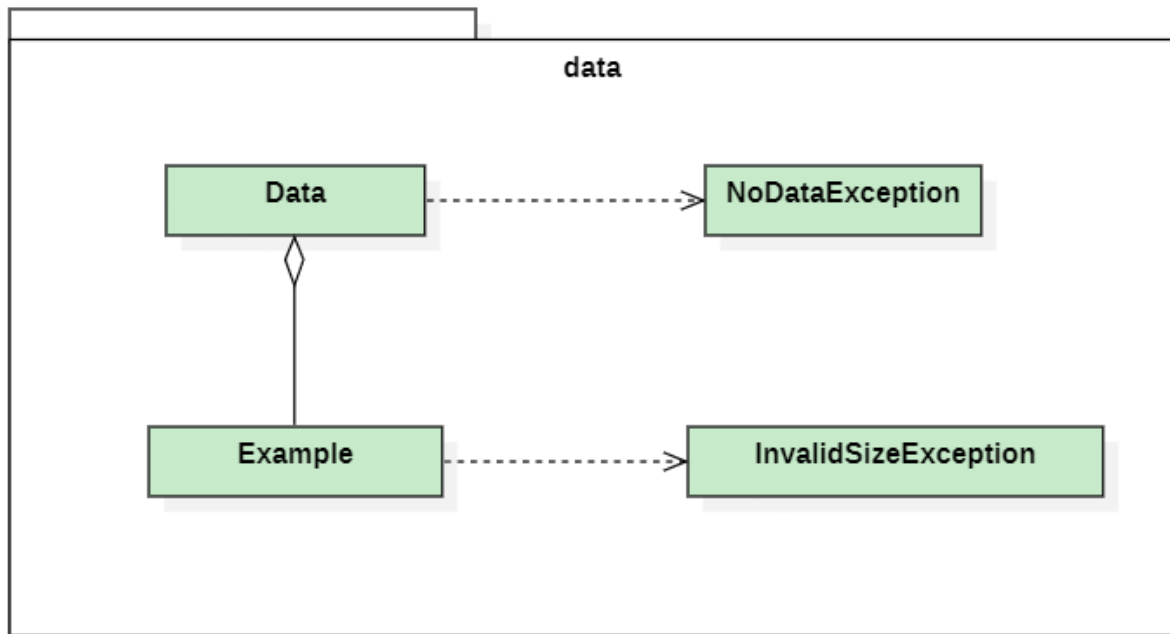
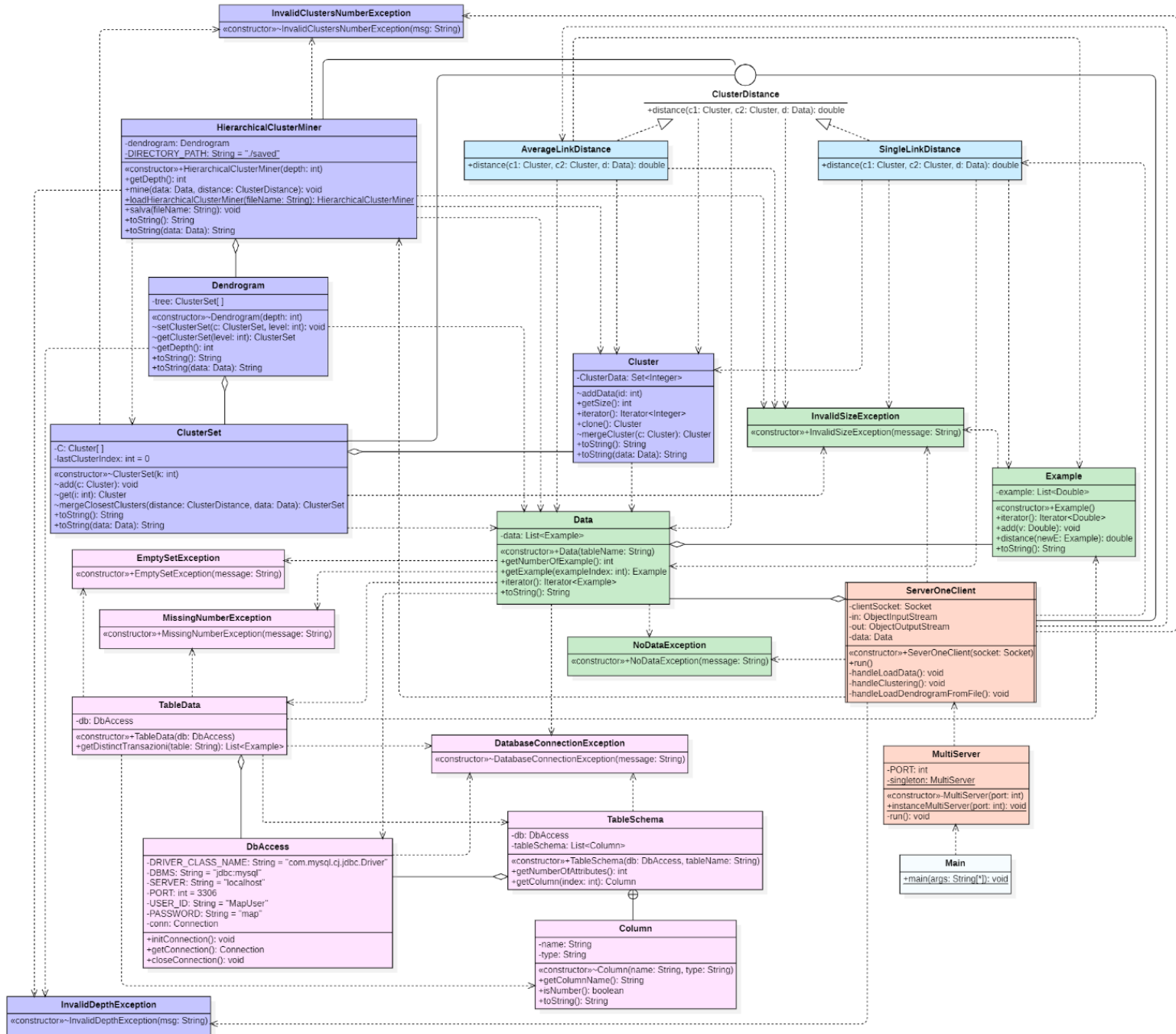
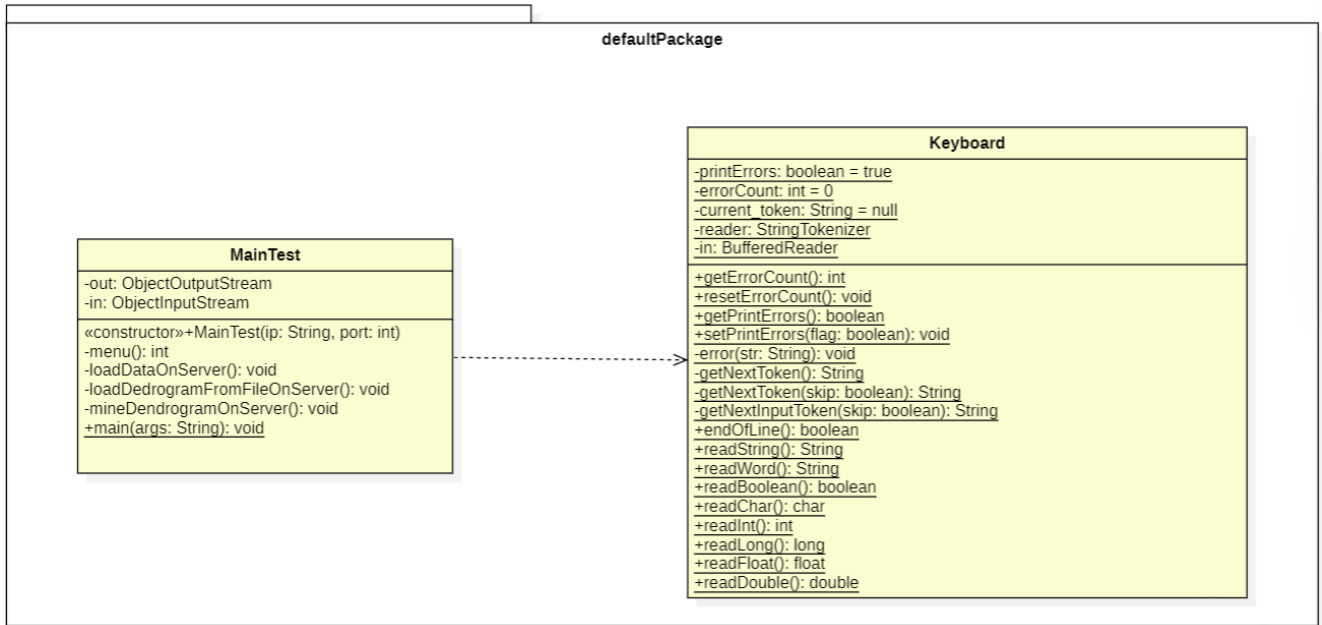


DIAGRAMMA DELLE CLASSI



3.2 CLIENT CONSOLE

DIAGRAMMA DEI PACKAGE



3.3 SERVER VERSIONE ESTENSIONE

DIAGRAMMA DEI PACKAGE

Nello specifico:

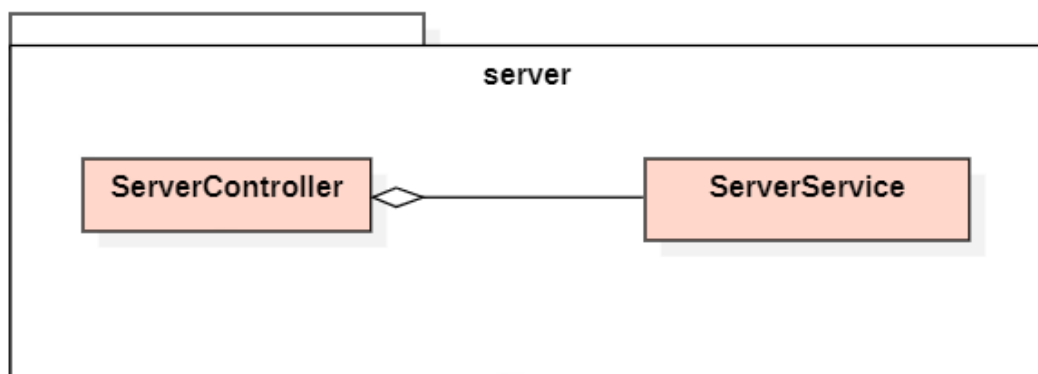
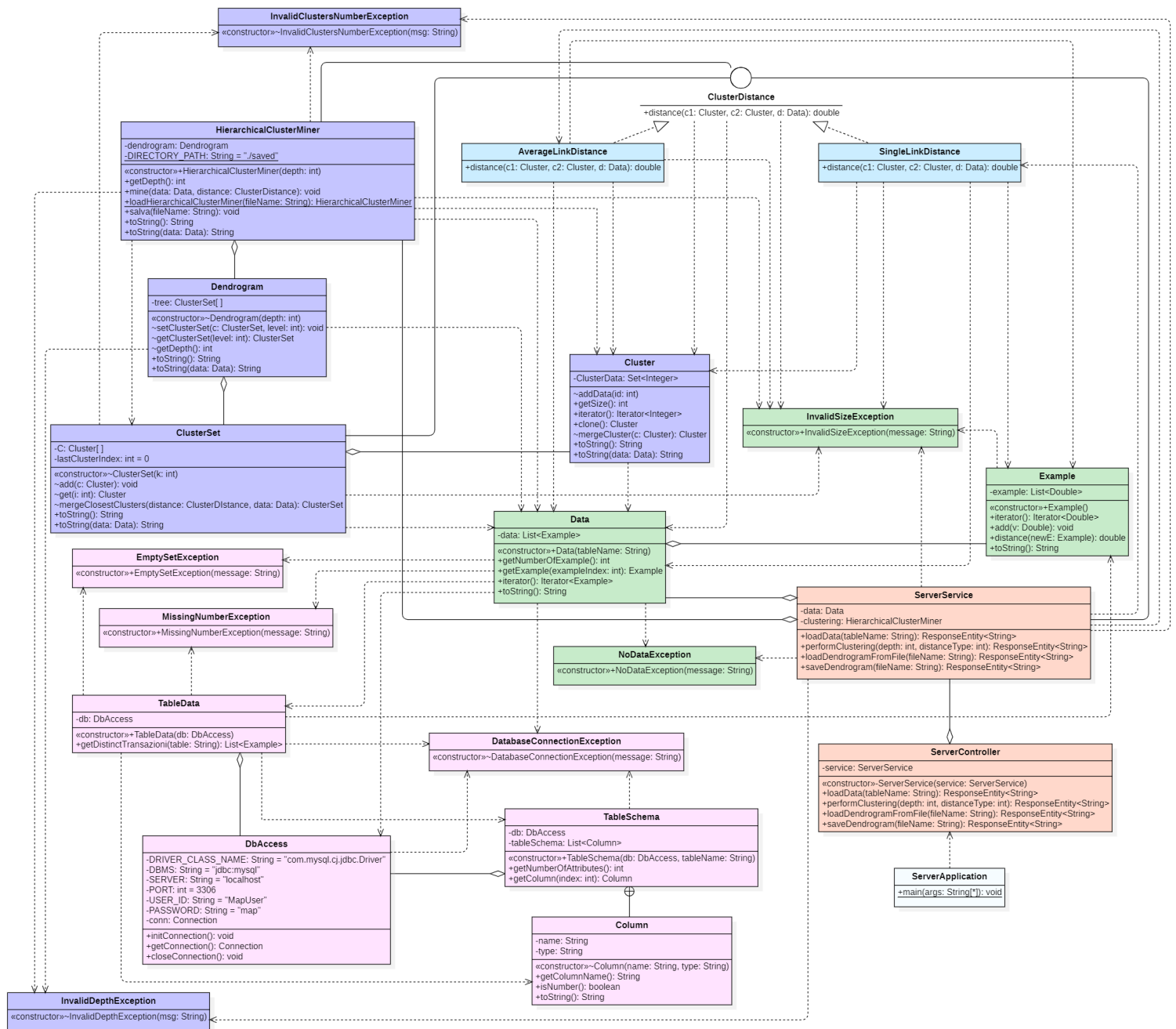


DIAGRAMMA DELLE CLASSI



3.4 CLIENT ESTENSIONE

DIAGRAMMA DEI PACKAGE

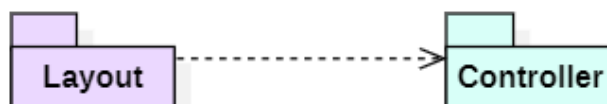
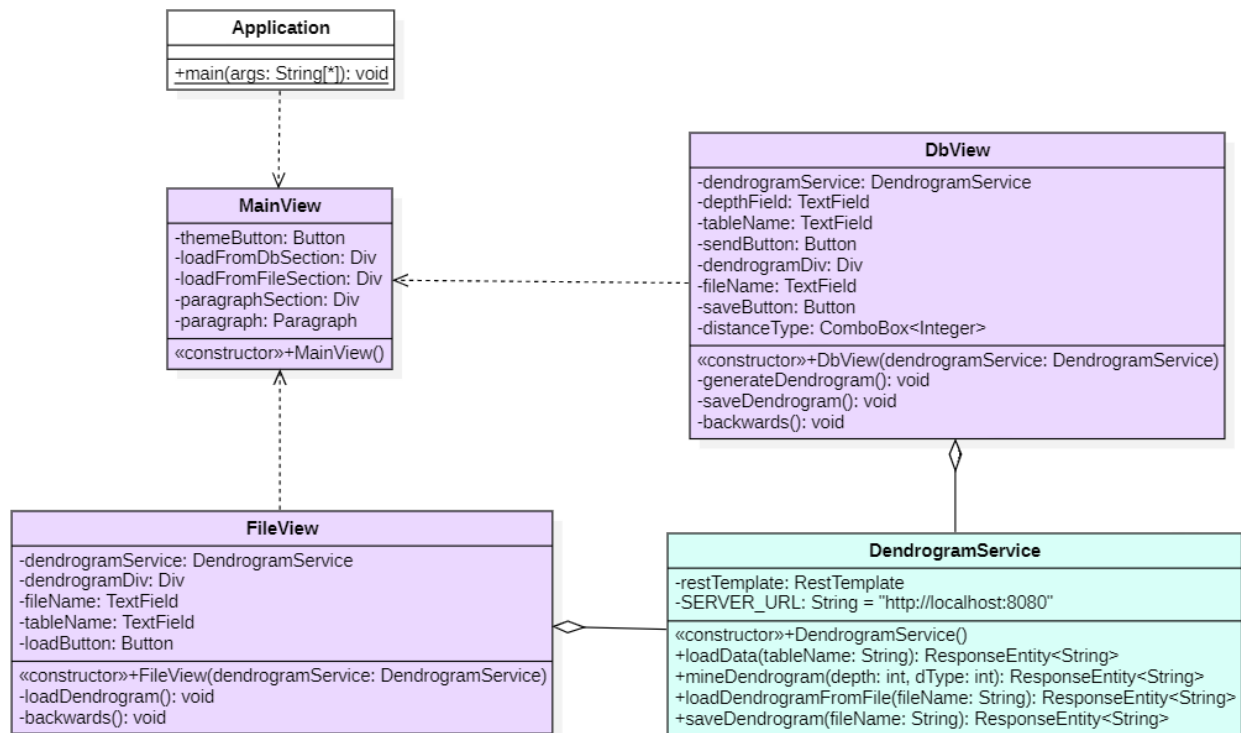


DIAGRAMMA DELLE CLASSI



4 MANUALE UTENTE

4.1 REQUISITI

Per poter eseguire il software H-CLUS non sono necessarie procedure complesse.

Per il corretto funzionamento è necessario seguire i seguenti passaggi:

- Scaricare e installare una versione aggiornata di **Java Development Kit (JDK)**. Assicurarsi di aver installato il JDK versione 22 o successiva.
- Scaricare e installare **MySQL**, assicurarsi di installare una versione 5.7 o successiva di MySQL Community Server.
- Avviare e verificare che il servizio MySQL sia in esecuzione.
- Eseguire lo script **“setup.sql”** per configurare il database con MySQL.

4.2 VERSIONE CONSOLE

ISTRUZIONI PER L'ESECUZIONE

SERVER

Prima di utilizzare il software, assicurarsi di aver rispettato i requisiti del punto precedente. È necessario che venga avviato il server prima di avviare il client. In caso contrario verrà visualizzato un messaggio di errore, interrompendo l'esecuzione del programma client.

Nella cartella principale del progetto individuare il file *run_server.bat* e prima di avviarlo, modificare il file sostituendo il valore del campo port col numero di porta sul quale avviare il server. Lasciare invariato il campo se si preferisce utilizzare il valore di default.

Per poter avviare il Server in maniera automatica è possibile utilizzare il file *run_server.bat*, infatti basta semplicemente individuare il file nella cartella del progetto e fare doppio click per eseguirlo. In alternativa si può aprire un terminale o prompt dei comandi, navigare nella directory dove si trova il file ed eseguire il comando: *run_server.bat*

È possibile avere aperto un unico Server alla volta, infatti il sistema impedirà di aprirne più di uno contemporaneamente.

CLIENT

Nella cartella principale del progetto individuare il file *run_client.bat* e prima di avviarlo, modificare il file sostituendo i valori dei campi address e port con l'indirizzo IP e il numero di porta del server avviato in precedenza. Lasciare invariati i campi se si preferisce utilizzare i valori di default.

Una volta completata la configurazione è possibile avviare il Client in maniera automatica utilizzando il file *run_client.bat*, infatti basta semplicemente individuare il file nella cartella del progetto e fare doppio click per eseguirlo. In alternativa si può aprire un terminale o prompt dei comandi, navigare nella directory dove si trova il file ed eseguire il comando: *run_client.bat*.

GUIDA UTENTE

Aperto *run_server.bat*, avviene l'avvio del server. Visualizzando il messaggio in figura, il server è pronto a ricevere richieste dai client.

```
C:\WINDOWS\system32\cmd. x + v
Server avviato sulla porta 8080
Server avviato: ServerSocket[addr=0.0.0.0/0.0.0.0,localport=8080]
Connessione client: Socket[addr=/127.0.0.1,port=60363,localport=8080]
```

Aperto *run_client.bat*, avviene l'avvio del client. Vengono visualizzate le informazioni di connessione e nome tabella dove bisogna inserire il nome della tabella su cui si vuole lavorare, in questo caso *exampleTab*.

```
C:\WINDOWS\system32\cmd. x + v
addr = /127.1.1.1
Socket[addr=/127.1.1.1,port=8080,localport=60363]
Nome tabella:
```

L'utente può quindi scegliere tra due opzioni: caricare il dendrogramma da un file oppure apprendere da una tabella da database. Scegliendo l'operazione due dal menu sarà quindi possibile eseguire nuove clusterizzazioni, salvando in un file i risultati scegliendo il nome e l'estensione. Affinché i dati vengano clusterizzati sarà necessario indicare il nome della tabella da cui estrapolare i dati, la profondità del dendrogramma e il tipo della distanza:

```
C:\WINDOWS\system32\cmd. x + v
Nome tabella:
exampletab
Scegli una opzione
(1) Carica Dendrogramma da File
(2) Apprendi Dendrogramma da Database
Risposta:2
Inserire la profondità del dendrogramma:
3
Distanza: single-link (1), average-link (2):
1
level0:
cluster0:<[1.0,2.0,0.0]>
cluster1:<[0.0,1.0,-1.0]>
cluster2:<[1.0,3.0,5.0]>
cluster3:<[1.0,3.0,4.0]>
cluster4:<[2.0,2.0,0.0]>

level1:
cluster0:<[1.0,2.0,0.0]><[2.0,2.0,0.0]>
cluster1:<[0.0,1.0,-1.0]>
cluster2:<[1.0,3.0,5.0]>
cluster3:<[1.0,3.0,4.0]>

level2:
cluster0:<[1.0,2.0,0.0]><[2.0,2.0,0.0]>
cluster1:<[0.0,1.0,-1.0]>
cluster2:<[1.0,3.0,5.0]><[1.0,3.0,4.0]>

Inserire il nome dell'archivio (comprensivo di estensione):
example3.bin
Dendrogramma salvato correttamente.
```

Invece, selezionando l'opzione uno dal menu precedente, sarà possibile leggere i dati da un file. È necessario inserire il nome del file con l'estensione:

```
C:\WINDOWS\system32\cmd. x + v
addr = /127.1.1.1
Socket[addr=/127.1.1.1,port=8080,localport=60590]
Nome tabella:
exampletab
Scegli una opzione
(1) Carica Dendrogramma da File
(2) Apprendi Dendrogramma da Database
Risposta:1
Inserire il nome dell'archivio (comprensivo di estensione):
example3.bin
level0:
cluster0:<[1.0,2.0,0.0]>
cluster1:<[0.0,1.0,-1.0]>
cluster2:<[1.0,3.0,5.0]>
cluster3:<[1.0,3.0,4.0]>
cluster4:<[2.0,2.0,0.0]>

level1:
cluster0:<[1.0,2.0,0.0]><[2.0,2.0,0.0]>
cluster1:<[0.0,1.0,-1.0]>
cluster2:<[1.0,3.0,5.0]>
cluster3:<[1.0,3.0,4.0]>

level2:
cluster0:<[1.0,2.0,0.0]><[2.0,2.0,0.0]>
cluster1:<[0.0,1.0,-1.0]>
cluster2:<[1.0,3.0,5.0]><[1.0,3.0,4.0]>
```

ESEMPI DI TEST DEL CLIENT CONSOLE

Avvio del client senza aver avviato il server

il server non è stato avviato e di conseguenza la connessione è stata rifiutata.

```
addr = /127.0.0.1
Connection refused: connect

Process finished with exit code 0
```

Inserimento nome tabella

in questo caso in input exampleTab, una tabella popolata e esistente in MapDB.

```
addr = /127.0.0.1
Socket[addr=/127.0.0.1,port=8080,localport=53974]
Nome tabella:
exampleTab
Scegli una opzione
(1) Carica Dendrogramma da File
(2) Apprendi Dendrogramma da Database
Risposta:
```

Inserimento di una tabella non esistente

in input una tabella che non esiste in MapDB.

```
addr = /127.0.0.1
Socket[addr=/127.0.0.1,port=8080,localport=54104]
Nome tabella:
nonesiste
Errore SQL durante il recupero dei dati dalla tabella: Table 'mapdb.nonesiste' doesn't exist

Nome tabella:
```

Scelta nel menu di un valore accettato

scelta compresa tra 1 o 2.

```
Scegli una opzione
(1) Carica Dendrogramma da File
(2) Apprendi Dendrogramma da Database
Risposta:2
Inserire la profondità del dendrogramma:
```

Inserimento nel menu di un valore non accettato

scelta diversa da 1 o 2.

```
Scegli una opzione
(1) Carica Dendrogramma da File
(2) Apprendi Dendrogramma da Database
Risposta:3
(1) Carica Dendrogramma da File
(2) Apprendi Dendrogramma da Database
Risposta:
```

Inserimento numero di profondità errata

inserimento di una profondità non valida.

```
Inserire la profondità del dendrogramma:
0
Distanza: single-link (1), average-link (2):
1
Profondità non valida!
```

Inserimento numero di profondità corretto

```
Inserire la profondità del dendrogramma:  
3  
Distanza: single-link (1), average-link (2):  
|
```

Inserimento numero di profondità compreso nel range – caso limite

in questo caso 1.

```
Inserire la profondità del dendrogramma:  
1  
Distanza: single-link (1), average-link (2):  
|
```

Inserimento numero di profondità compreso nel range – caso limite

in questo caso 5.

```
Inserire la profondità del dendrogramma:  
5  
Distanza: single-link (1), average-link (2):  
|
```

Inserimento numero di profondità non compreso nel range

in questo caso un numero di profondità maggiore del numero degli esempi.

```
Inserire la profondità del dendrogramma:  
12  
Distanza: single-link (1), average-link (2):  
1  
Profondità del dendrogramma maggiore del numero degli esempi!
```


Inserimento tipo di distanza errata

scelta diversa da 1 o 2.

```
Distanza: single-link (1), average-link (2):  
3  
Distanza: single-link (1), average-link (2):  
|
```

Inserimento tipo di distanza corretto

scelta compresa tra 1 o 2.

```
Distanza: single-link (1), average-link (2):  
2  
level0:  
cluster0:<[1.0,2.0,0.0]>  
cluster1:<[0.0,1.0,-1.0]>  
cluster2:<[1.0,3.0,5.0]>  
cluster3:<[1.0,3.0,4.0]>  
cluster4:<[2.0,2.0,0.0]>  
  
level1:  
cluster0:<[1.0,2.0,0.0]><[2.0,2.0,0.0]>  
cluster1:<[0.0,1.0,-1.0]>  
cluster2:<[1.0,3.0,5.0]>  
cluster3:<[1.0,3.0,4.0]>  
  
Inserire il nome dell'archivio (comprensivo di estensione):  
|
```

Inserimento nome file corretto – salvataggio

un nome del file corretto + estensione.

```
Inserire il nome dell'archivio (comprensivo di estensione):  
ex1.txt  
Dendrogramma salvato correttamente.
```

Inserimento nome file senza estensione – salvataggio

nome del file senza specificare l'estensione.

```
Inserire il nome dell'archivio (comprensivo di estensione):  
ex1  
Errore durante il salvataggio del dendrogramma: Estensione non valida. Il nome del file deve terminare con una estensione: .bin, .dat, .txt, .csv, .xml, .json, .ser
```

Inserimento nome file con caratteri non validi – salvataggio

in questo caso sono presenti caratteri speciali non validi.

```
Inserire il nome dell'archivio (comprensivo di estensione):  
ex?2.bin  
Errore durante il salvataggio del dendrogramma: Il nome del file contiene caratteri non validi.
```

Inserimento nome file già esistente – salvataggio

```
Inserire il nome dell'archivio (comprensivo di estensione):  
example3.txt  
Errore. Il file example3.txt esiste già.
```

Inserimento nome file esistente – caricamento

un file presente nei salvataggi.

```
Inserire il nome dell'archivio (comprensivo di estensione):  
example3.bin  
level0:  
cluster0:<[1.0,2.0,0.0]>  
cluster1:<[0.0,1.0,-1.0]>  
cluster2:<[1.0,3.0,5.0]>  
cluster3:<[1.0,3.0,4.0]>  
cluster4:<[2.0,2.0,0.0]>  
  
level1:  
cluster0:<[1.0,2.0,0.0]><[2.0,2.0,0.0]>  
cluster1:<[0.0,1.0,-1.0]>  
cluster2:<[1.0,3.0,5.0]>  
cluster3:<[1.0,3.0,4.0]>  
  
level2:  
cluster0:<[1.0,2.0,0.0]><[2.0,2.0,0.0]>  
cluster1:<[0.0,1.0,-1.0]>  
cluster2:<[1.0,3.0,5.0]><[1.0,3.0,4.0]>
```

Inserimento nome file non esistente – caricamento

un file che non è presente nei salvataggi.

```
Inserire il nome dell'archivio (comprensivo di estensione):
nonesiste.dat
File non trovato: nonesiste.dat
```

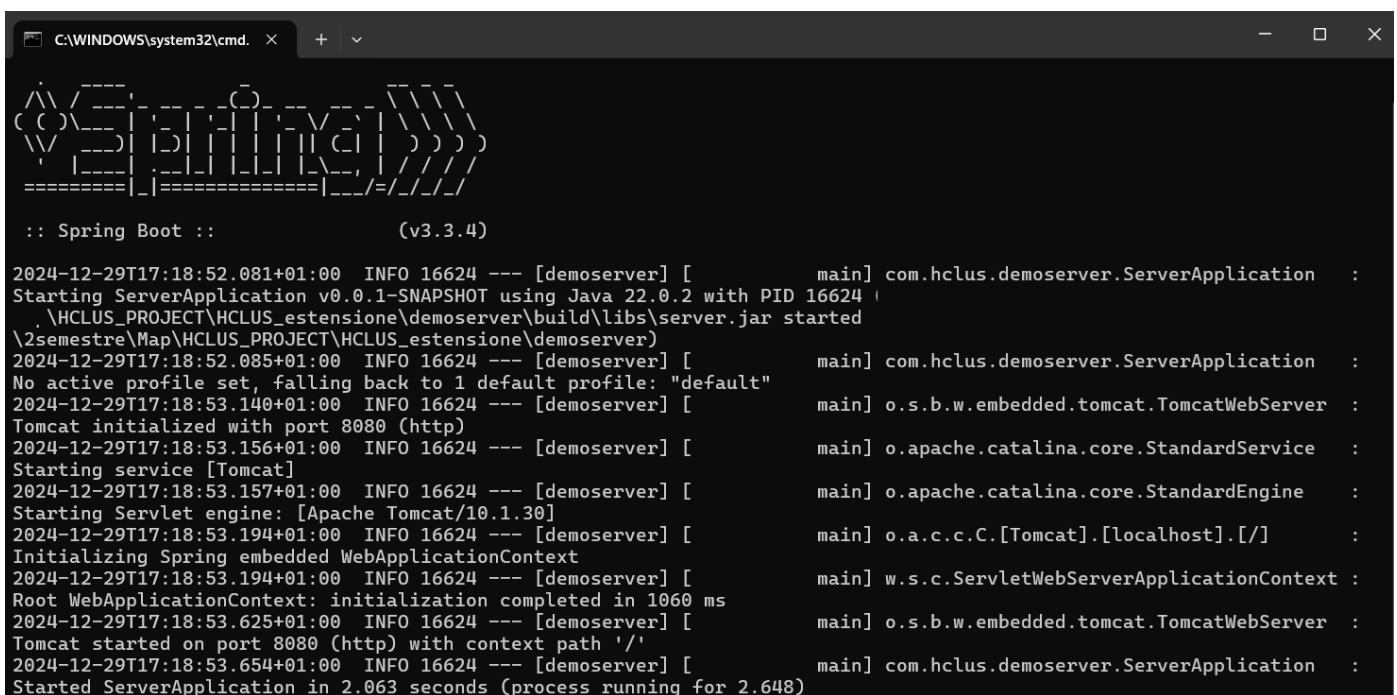
4.3 VERSIONE INTERFACCIA WEB

ISTRUZIONI PER L'ESECUZIONE

Anche in questo caso è necessario rispettare i requisiti al punto 4.1. È possibile avviare automaticamente il programma individuando nella cartella principale del progetto il file *run_server.bat* e fare doppio click per avviarlo. Stessa cosa per il client con il file *run_client.bat*.

GUIDA UTENTE

Aprendo il file *run_server.bat* si avvia il server:

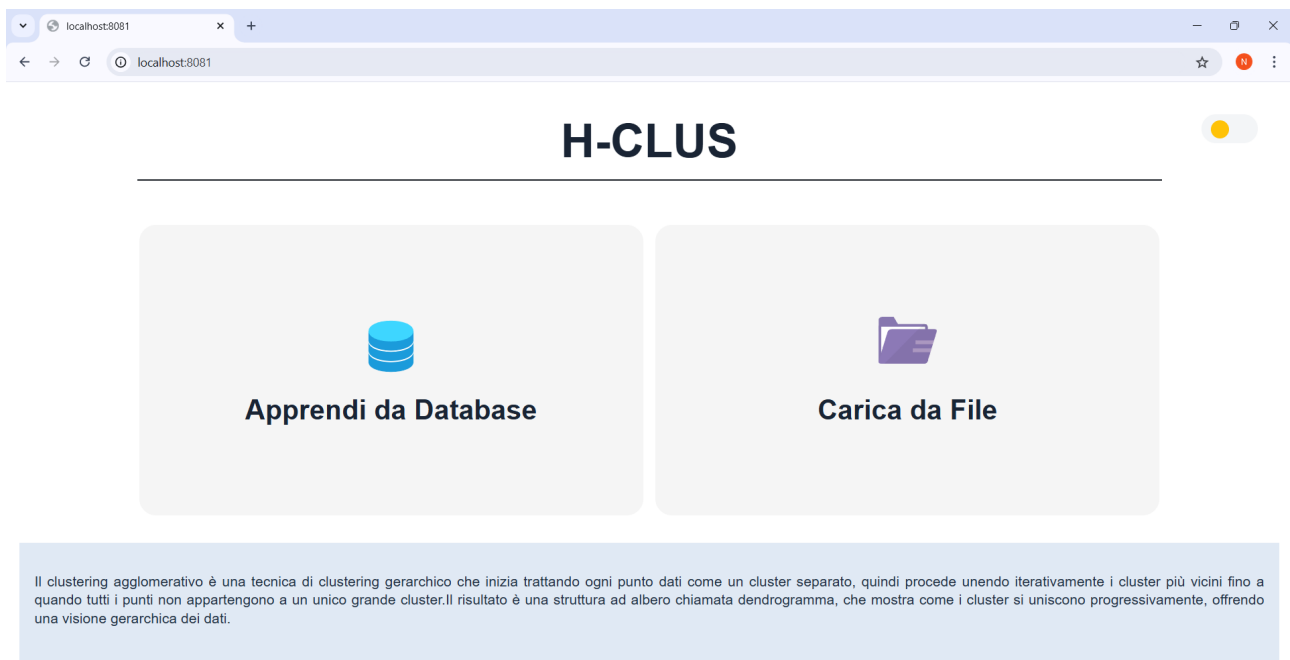


```
C:\WINDOWS\system32\cmd. X + v

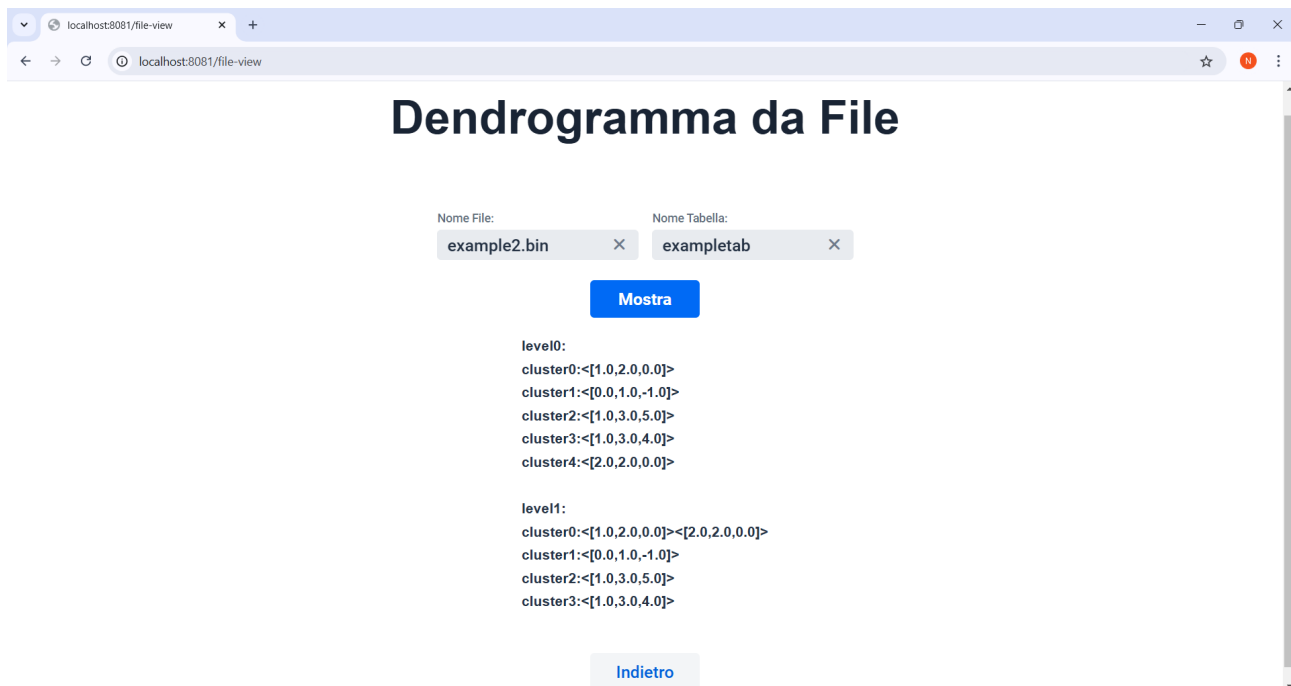
:: Spring Boot :: (v3.3.4)

2024-12-29T17:18:52.081+01:00 INFO 16624 --- [demoserver] [main] com.hclus.demoserver.ServerApplication :
Starting ServerApplication v0.0.1-SNAPSHOT using Java 22.0.2 with PID 16624 (
.\HCLUS_PROJECT\HCLUS_estensione\demoserver\build\libs\server.jar started
2024-12-29T17:18:52.085+01:00 INFO 16624 --- [demoserver] [main] com.hclus.demoserver.ServerApplication :
No active profile set, falling back to 1 default profile: "default"
2024-12-29T17:18:53.140+01:00 INFO 16624 --- [demoserver] [main] o.s.b.w.embedded.tomcat.TomcatWebServer :
Tomcat initialized with port 8080 (http)
2024-12-29T17:18:53.156+01:00 INFO 16624 --- [demoserver] [main] o.apache.catalina.core.StandardService :
Starting service [Tomcat]
2024-12-29T17:18:53.157+01:00 INFO 16624 --- [demoserver] [main] o.apache.catalina.core.StandardEngine :
Starting Servlet engine: [Apache Tomcat/10.1.30]
2024-12-29T17:18:53.194+01:00 INFO 16624 --- [demoserver] [main] o.a.c.c.C.[Tomcat].[localhost].[/] :
Initializing Spring embedded WebApplicationContext
2024-12-29T17:18:53.194+01:00 INFO 16624 --- [demoserver] [main] w.s.c.ServletWebServerApplicationContext :
Root WebApplicationContext: initialization completed in 1060 ms
2024-12-29T17:18:53.625+01:00 INFO 16624 --- [demoserver] [main] o.s.b.w.embedded.tomcat.TomcatWebServer :
Tomcat started on port 8080 (http) with context path '/'
2024-12-29T17:18:53.654+01:00 INFO 16624 --- [demoserver] [main] com.hclus.demoserver.ServerApplication :
Started ServerApplication in 2.063 seconds (process running for 2.648)
```

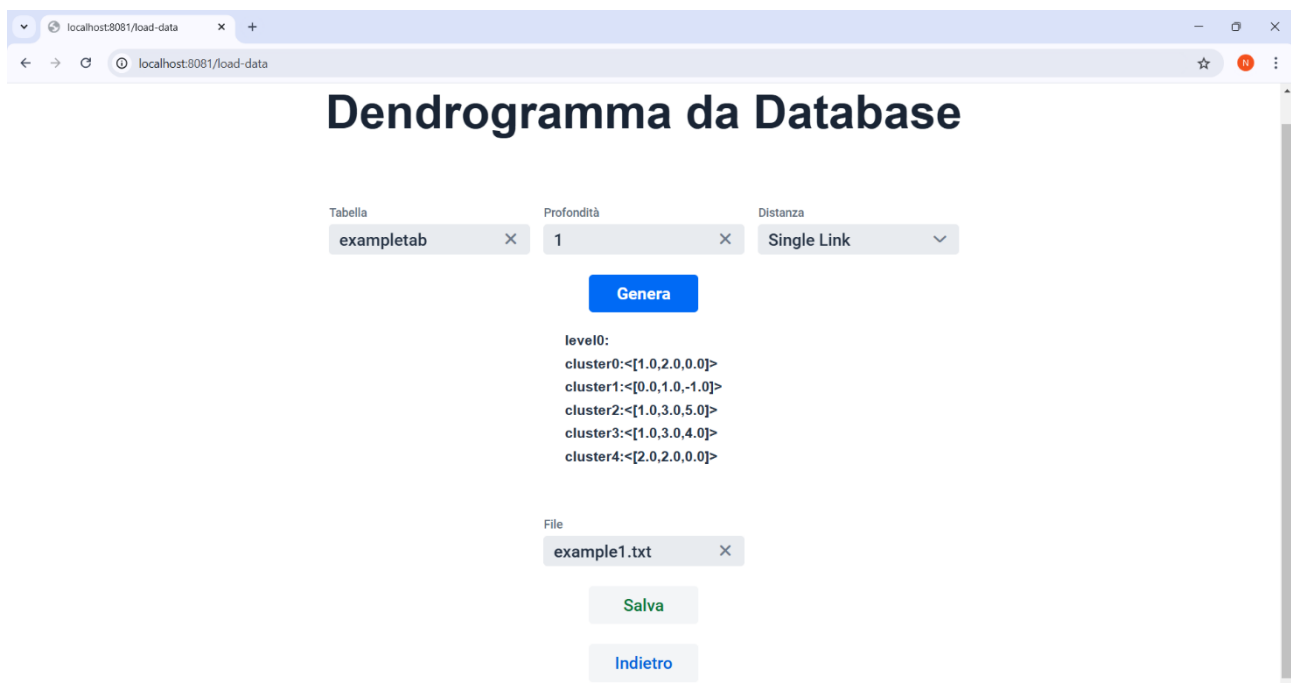
Invece dopo l'avvio del client, tramite il *file run_client.bat*, sarà possibile recarsi su un qualsiasi browser all'indirizzo locale **localhost:8081** dove sarà possibile visualizzare la pagina web principale con la finestra del menu, composta da una breve descrizione dell'algoritmo applicato e due sezioni che portano alle pagine dedicate alle funzionalità del programma:



Se nella finestra iniziale si seleziona l'opzione "Carica da File" si accederà alla pagina dedicata alla lettura dei file salvati. Sarà necessario inserire il nome della tabella e il nome del file affinché possa essere caricato il file. Una volta riempiti i campi necessari, basterà premere il bottone "Mostra" per visualizzare il risultato. Inoltre, sarà possibile tornare alla pagina principale tramite il bottone "Indietro".



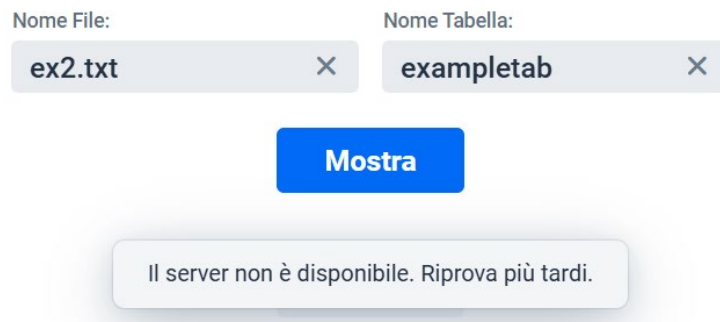
Se si seleziona l'opzione "Apprendi da Database" si accederà alla pagina dedicata alla clusterizzazione dei dati da una tabella del Database. Anche in questo caso sarà necessario inserire il nome della tabella, la profondità e il tipo di distanza. Una volta riempiti i campi necessari, basterà premere il tasto "Genera" per generare il dendrogramma. Dopo la generazione del risultato apparirà la sezione per il salvataggio. Inoltre, sarà possibile tornare alla pagina principale tramite il bottone "Indietro".



ESEMPI DI TEST DEL CLIENT ESTENSIONE

Avvio del client senza aver avviato il server

Quando si prova a fare qualsiasi operazione e il server non è stato avviato, verrà visualizzato il messaggio di errore.



Nome File:

Nome Tabella:

Il server non è disponibile. Riprova più tardi.

Inserimento di una tabella non esistente



Tabella:

Profondità:

Distanza:

Errore SQL durante il recupero dei dati dalla tabella: la tabella test non esiste.

Inserimento numero di profondità errata

inserimento di una profondità non valida.



Tabella:

Profondità:

Distanza:

Profondità non valida!

Inserimento numero di profondità non compreso nel range

in questo caso un numero di profondità maggiore del numero degli esempi.

Tabella

exampletab

×

Profondità

6

×

Distanza

Average Link

▼

Genera

Profondità del dendrogramma maggiore del numero degli esempi!

Inserimento nome file corretto – salvataggio

un nome del file corretto + estensione.

level0:
cluster0:<[1.0,2.0,0.0]>
cluster2:<[1.0,0.0,0.0]>
cluster3:<[1.0,3.0,4.0]>
cluster4:<[2.0,2.0,0.0]>

Dendrogramma salvato correttamente.

File

example1.bin

×

Salva

Inserimento nome file senza estensione – salvataggio

nome del file senza specificare l'estensione.

level0:
cluster0:<[1.0,2.0,0.0]>
cluster2:<[1.0,0.0,0.0]>
cluster3:<[1.0,3.0,4.0]>
cluster4:<[2.0,2.0,0.0]>

Estensione non valida. Il nome del file deve terminare con una estensione: .bin, .dat, .txt, .csv, .xml, .json, .ser

File

example1

×

Salva

Inserimento nome file con caratteri non validi – salvataggio

in questo caso sono presenti caratteri speciali non validi.

level0:
cluster0:<[1.0,2.0,0.0]>
cluster2:<[1.0,0.0,0.0]>
cluster3:<[1.0,3.0,4.0]>
cluster4:<[2.0,2.0,0.0]>

Il nome del file contiene caratteri non validi.

File
?test*.txt X

Salva

Inserimento nome file non esistente – caricamento

un file che non è presente nei salvataggi.

Nome File: Nome Tabella:

test2.txt X exampletab X

Mostra

File non trovato: test2.txt

Inserimento nome tabella non esistente – caricamento

Nome File: Nome Tabella:

example2.bin X example X

Mostra

Errore SQL durante il recupero dei dati dalla tabella: la tabella example non esiste.

Inserimento nome file vuoto

sarà visualizzato un messaggio di errore se non si inserisce un nome del file.

```
level0:  
cluster0:<[1.0,2.0,0.0]>  
cluster1:<[1.0,0.0,0.0]>  
cluster2:<[1.0,0.0,0.0]>  
cluster3:<[1.0,3.0,4.0]>  
cluster4:<[2.0,2.0,0.0]>
```

File

Nome file

Salva

Inserimento nome file già esistente - salvataggio

```
level1:  
cluster0:<[1.0,2.0,0.0]><[2.0,2.0,0.0]>  
cluster1:<[1.0,0.0,0.0]>  
cluster2:<[1.0,3.0,5.0]>  
cluster3:<[1.0,3.0,4.0]>
```

File

example2.dat

Salva

Caso in cui non si compilano i campi

Tabella

Nome tabella

Profondità

Inserisci profondità

Distanza

Tipo distanza

Genera

Compila tutti i campi richiesti.

Indietro