

UNIVERSITÀ DEGLI STUDI DI SALERNO

Fondamenti di Informatica

Gestione dei File in MATLAB

Prof. Christian Esposito

Corso di Laurea in Ingegneria Meccanica e Gestionale (Classe I)

A.A. 2017/18

MATLAB



OUTLINE

- Caratteristiche dei File
- Apertura/Chiusura di File
- Scrittura/Lettura da File
- Importare Dati da File

Caratteristiche dei File

- I **file** possono essere visti come dei **contenitori permanenti** di informazioni
- Il Sistema Operativo si occupa della loro gestione ed offre la possibilità, agli sviluppatori, di:
 - Apertura/chiusura di un file
 - Lettura/scrittura di un file
 - Ecc.

Apertura/Chiusura di File – 1/2

```
fid = fopen(percorso_del_file, modalita_apertura)
```

Sintassi

- Tramite **fopen** è possibile aprire un file, memorizzato nel percorso indicato da **percorso_del_file** nella modalità d'apertura specificata da **modalita_apertura**
- La modalità di apertura (**modalita_apertura**) può essere:
 - Scrittura (**modalita_apertura = 'w'**)
 - Lettura (**modalita_apertura = 'r'**)
 - Scrittura e lettura (**modalita_apertura = 'w+'** e **modalita_apertura = 'r+'**)
 - Append (**modalita_apertura = 'a '**): apre un file esistente o crea un nuovo file in scrittura con posizionamento alla fine del file.
- La funzione **fopen** restituisce il valore **fid**, che è un identificatore associato al file aperto

Apertura/Chiusura di File – 2/2

fclose(fid)

Sintassi

- Chiude il file identificato da **fid**
- Esempio utilizzo **fopen** e **fclose**

```
>> fid = fopen('C:\test.txt', 'r');  
>> fclose(fid);
```

Apertura/Chiusura di File – 2/2

fclose(fid)

Sintassi

- Chiude il file identificato da **fid**
- **Esempio utilizzo fopen e fclose**

```
>> fid = fopen('C:\test.txt', 'r');  
>> fclose(fid);
```

- **Percorso:** C:\test.txt
- **Modalità di apertura:** lettura ('r')

Scrittura di un File (1/2)

- Alcune funzioni per la **scrittura** su file
 - Il file deve essere precedentemente aperto mediante la **fopen**
 - **fwrite**
 - Scrive un array, passato come parametro, su un file specificato
 - `help fwrite` per maggiori dettagli
 - **fprintf**
 - Scrive su file, secondo un formato specificato dall'utente
 - `help fprintf` per maggiori dettagli

Scrittura di un File (2/2)

- `cont = fwrite(fid, array, formato)`
 - `cont`: indica il numero di valori effettivamente scritti nel file
 - `fid`: identificatore del file su cui scrivere (nb: il file deve essere stato aperto in precedenza)
 - `array`: array contenente i dati da salvare
 - `formato`: specifica il formato in cui i dati verranno salvati. Formati principali:
 - `char`, `int8`, `int16`, `int32`, `int64`, `float32`, `float64` (i numeri indicano il numero di bit usati per rappresentare i valori)

Lettura da File (1/2)

- Alcune funzioni per la **lettura** da file
 - Il file deve essere precedentemente aperto mediante la **fopen**
 - **fread**
 - Legge un array, di lunghezza specificata, da un file indicato tramite parametro
 - `help fread` per maggiori dettagli
 - **fscanf**
 - Legge da file, specificando esplicitamente il formato di tali dati
 - `help fscanf` per maggiori dettagli

Lettura da File (2/2)

- `[array cont]=fread(fid, size, formato)`
 - `cont`, `fid`, e `formato` hanno il significato per `fwrite`
 - I dati letti vengono memorizzati in `array`, e `size` è la dimensione dei dati da leggere. Tre possibilità
 - `n`: legge esattamente `n` valori. Dopo l'esecuzione di questa istruzione `array` sarà un vettore colonna contenente questi `n` valori
 - `Inf`: legge fino alla fine del file. Dopo l'esecuzione di questa istruzione `array` sarà un vettore colonna contenente tutti i valori letti
 - `[n m]`: legge esattamente `nxm` valori. Dopo l'esecuzione di questa istruzione `array` sarà una matrice `nxm` contenente tutti i valori letti

Esempio di Scrittura e Lettura da File (1/2)

```
% genera un vettore riga contenente numeri casuali
a=rand(1,1000);
%richiede all'utente il nome del file
filename=input('inserisci un nome di file ');
[fid msg]=fopen(filename, 'w'); %apre il file
%se il file e` stato aperto con successo...
if(fid>0)
    cont=fwrite(fid,a,'float64'); %scrive a su file
    %informa l'utente dell'avvenuta scrittura
    disp([num2str(cont) ' valori scritti...']);
    fclose(fid); %chiude il file
else
    disp(msg);
end
```

Esempio di Scrittura e Lettura da File (2/2)

```
% genera un vettore riga contenente numeri casuali
a=rand(1,1000);
%richiede all'utente il nome del file
filename=input('inserisci un nome di file ');
[fid msg]=fopen(filename, 'r'); %apre il file
%se il file e` stato aperto con successo...
if(fid>0)
    [vett cont]=fread(fid,[1 1000],'float64');
    %informa l'utente dell'avvenuta lettura
    disp([num2str(cont) ' valori letti...']);
    fclose(fid); %chiude il file
else
    disp(msg);
end
```

Importare Dati da File – 1/4

- Con la funzione **importdata** è possibile di importare dati da diversi tipi di file:
 - Testuali
 - CSV
 - Ecc.
- Non è necessario che il file sia stato aperto precedentemente (tramite **fopen**)
- `help importdata` per maggiori dettagli

Importare Dati da File – 2/4

Esempio 1

```
1, 4, 6  
4, 2, 5  
6, 5, 4
```

matrice.txt

```
>> A = importdata('matrice.txt');
```

```
A =
```

```
    1     4     6  
    4     2     5  
    6     5     4
```

Importare Dati da File – 2/4

Esempio 1

```
1, 4, 6  
4, 2, 5  
6, 5, 4
```

matrice.txt

```
>> A = importdata('matrice.txt');
```

A =

```
1    4    6  
4    2    5  
6    5    4
```

La virgola (simbolo ,) è interpretata come *separatore di colonne*

Importare Dati da File – 3/4

Esempio 2

```
1 4 6
4 2 5
6 5 4
```

matrice.txt

```
>> A = importdata('matrice.txt', ' ');
```

```
A =
```

```
1     4     6
4     2     5
6     5     4
```

Importare Dati da File – 3/4

Esempio 2

```
1 4 6
4 2 5
6 5 4
```

matrice.txt

```
>> A = importdata('matrice.txt', ' ');
```

A =

```
1 4 6
4 2 5
6 5 4
```

È possibile specificare anche un delimitatore di colonne in **importdata** (**spazio** nell'esempio)

Importare Dati da File – 4/4

Esempio 3

Studenti/Voti esame	Esame 1	Esame 2	Esame 3	Esame 4	Esame 5
Matricola 1	28	25	30	23	19
Matricola 2	24	27	28	21	24
Matricola 3	25	25	19	18	22
Matricola 4	21	30	30	22	30

Importare Dati da File – 4/4

Esempio 3

Studenti/Voti esame	Esame 1	Esame 2	Esame 3	Esame 4	Esame 5
Matricola 1	28	25	30	23	19
Matricola 2	24	27	28	21	24
Matricola 3	25	25	19	18	22
Matricola 4	21	30	30	22	30

vs.txt

```
Studenti/Voti, E1, E2, E3, E4, E5  
Matricola 1, 28, 25, 30, 23, 19  
Matricola 2, 24, 27, 28, 21, 24  
Matricola 3, 25, 25, 19, 18, 22  
Matricola 4, 21, 30, 30, 22, 30
```

Importare Dati da File – 4/4

Esempio 3

Studenti/Voti esame	Esame 1	Esame 2	Esame 3	Esame 4	Esame 5
Matricola 1	28	25	30	23	19
Matricola 2	24	27	28	21	24
Matricola 3	25	25	19	18	22
Matricola 4	21	30	30	22	30

Nome del file

vs.txt

```
Studenti/Voti, E1, E2, E3, E4, E5  
Matricola 1, 28, 25, 30, 23, 19  
Matricola 2, 24, 27, 28, 21, 24  
Matricola 3, 25, 25, 19, 18, 22  
Matricola 4, 21, 30, 30, 22, 30
```

Importare Dati da File – 4/4

Esempio 3

vs.txt

```
Studenti/Voti, E1, E2, E3, E4, E5  
Matricola 1, 28, 25, 30, 23, 19  
Matricola 2, 24, 27, 28, 21, 24  
Matricola 3, 25, 25, 19, 18, 22  
Matricola 4, 21, 30, 30, 22, 30
```

```
>> file = importdata('vs.txt');  
file =  
      data: [4x5 double]  
   textdata: {5x6 cell}
```

Importare Dati da File – 4/4

Esempio 3

vs.txt

```
Studenti/Voti, E1, E2, E3, E4, E5  
Matricola 1, 28, 25, 30, 23, 19  
Matricola 2, 24, 27, 28, 21, 24  
Matricola 3, 25, 25, 19, 18, 22  
Matricola 4, 21, 30, 30, 22, 30
```

```
>> file = importdata('vs.txt');  
file =  
    data: [4x5 double]  
    textdata: {5x6 cell}
```

Tutti i dati numerici vengono
immagazzinati in data

Importare Dati da File – 4/4

Esempio 3

vs.txt

```
Studenti/Voti, E1, E2, E3, E4, E5  
Matricola 1, 28, 25, 30, 23, 19  
Matricola 2, 24, 27, 28, 21, 24  
Matricola 3, 25, 25, 19, 18, 22  
Matricola 4, 21, 30, 30, 22, 30
```

```
>> file = importdata('vs.txt');  
file =  
      data: [4x5 double]  
      textdata: {5x6 cell}
```

textdata conterrà tutti i
dati testuali (es., intestazione
delle righe e colonne)

Importare Dati da File – 4/4

Esempio 3

```
>> file = importdata('vs.txt');  
  
file =  
      data: [4x5 double]  
      textdata: {5x6 cell}  
  
>> file.textdata  
  
ans =  
  
      'Studenti/Voti'      ' E1'      ' E2'      ' E3'      ' E4'      ' E5'  
      'Matricola 1'      ''      ''      ''      ''      ''  
      'Matricola 2'      ''      ''      ''      ''      ''  
      'Matricola 3'      ''      ''      ''      ''      ''  
      'Matricola 4'      ''      ''      ''      ''      ''
```

Importare Dati da File – 4/4

Esempio 3

```
>> file = importdata('vs.txt');
```

```
file =
```

```
    data: [4x5 double]
```

```
  textdata: {5x6 cell}
```

```
>> A = file.data
```

```
A =
```

```
    28    25    30    23    19
    24    27    28    21    24
    25    25    19    18    22
    21    30    30    22    30
```

Importare Dati da File Tramite Interfaccia Grafica

- Tramite **uiimport** è possibile selezionare, tramite *interfaccia grafica*, diverse opzioni per l'import di dati:
 - Tipo di separatore di colonna
 - Quante righe/colonne occupa l'intestazione
 - Ecc.
- Non è necessario che il file sia stato aperto precedentemente (tramite **fopen**)

Importare Dati da File Tramite Interfaccia Grafica

```
>> uiimport('data.txt')
```

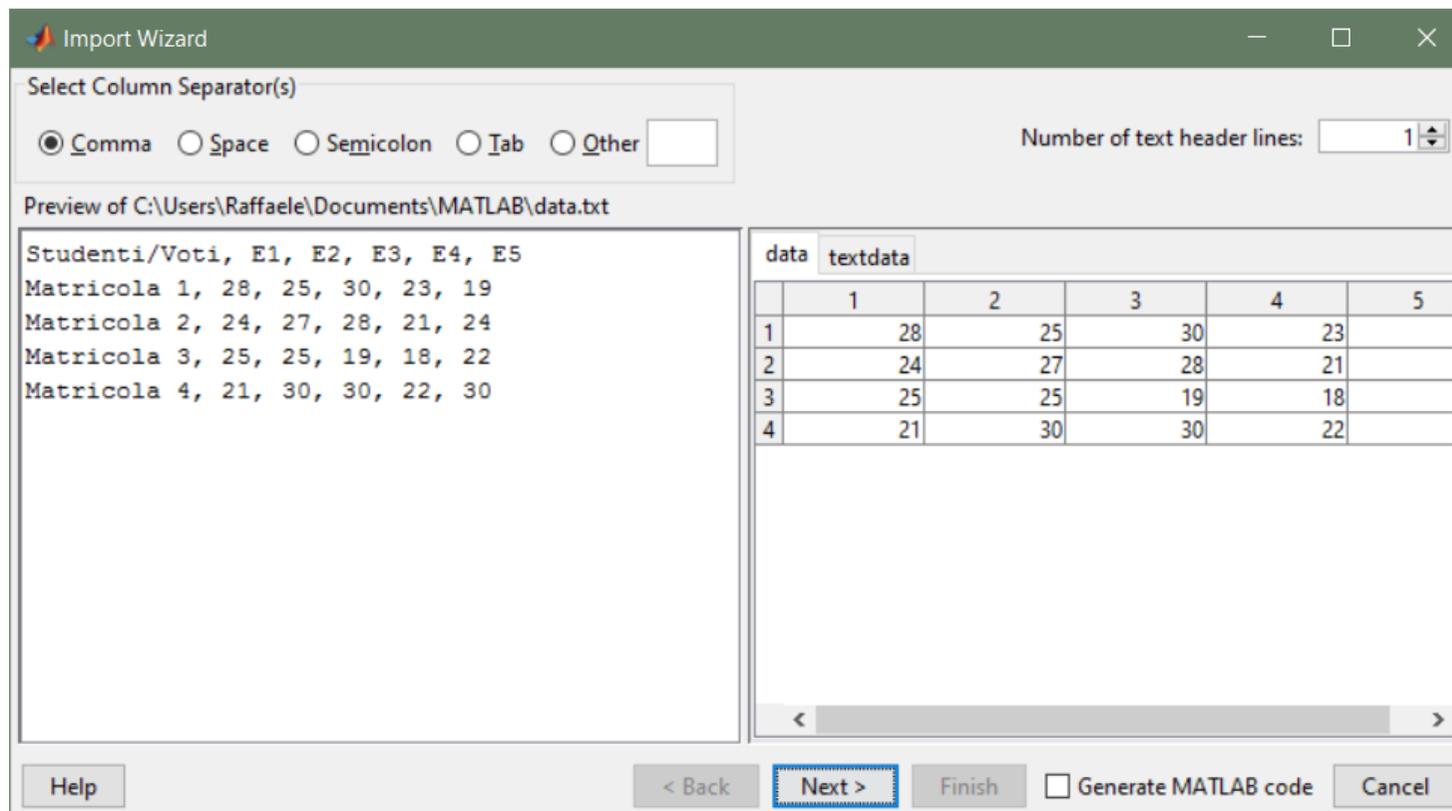
Importare Dati da File Tramite Interfaccia Grafica

```
>> uiimport('data.txt')
```

'data.txt' è il percorso del file

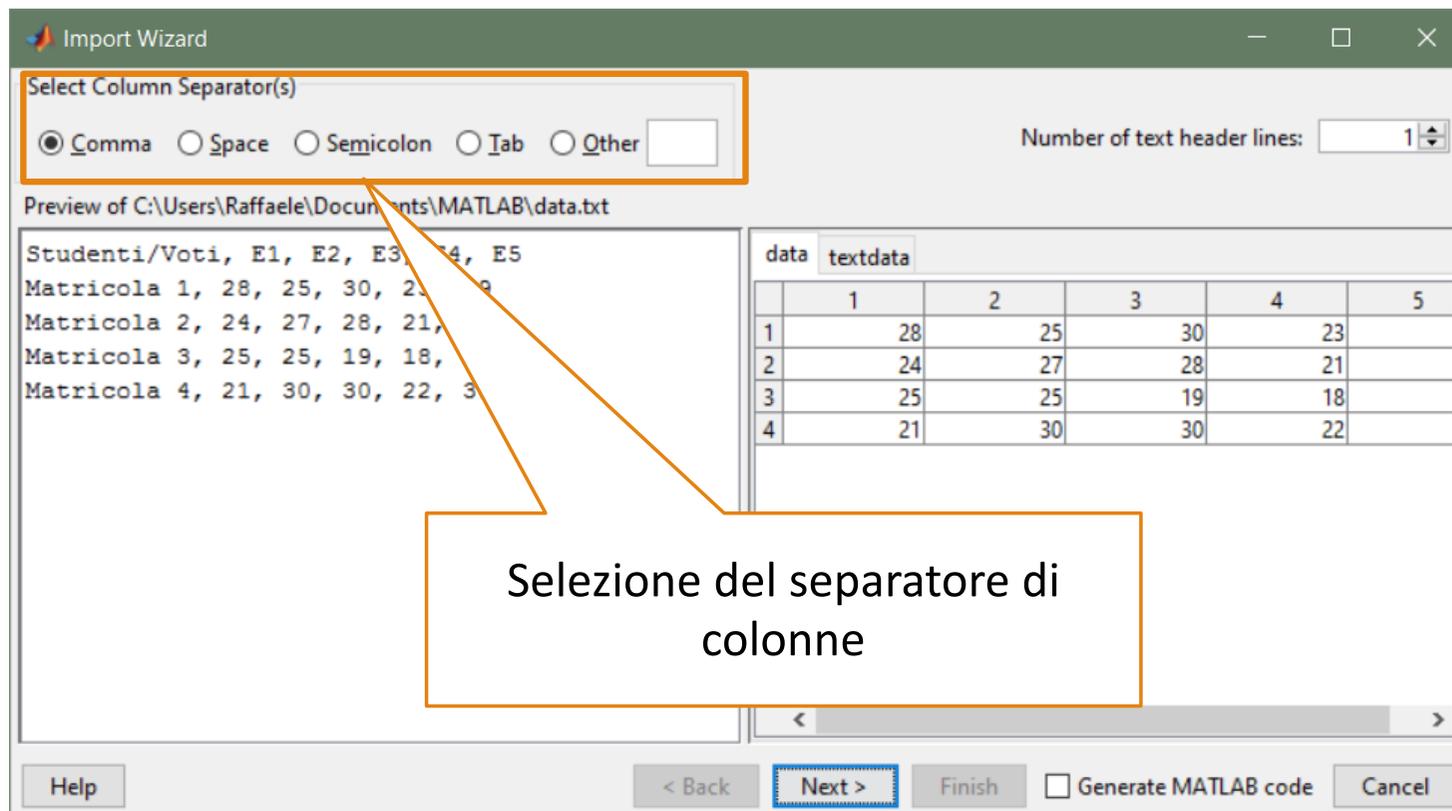
Importare Dati da File Tramite Interfaccia Grafica | Passaggio 1

```
>> uiimport('data.txt')
```



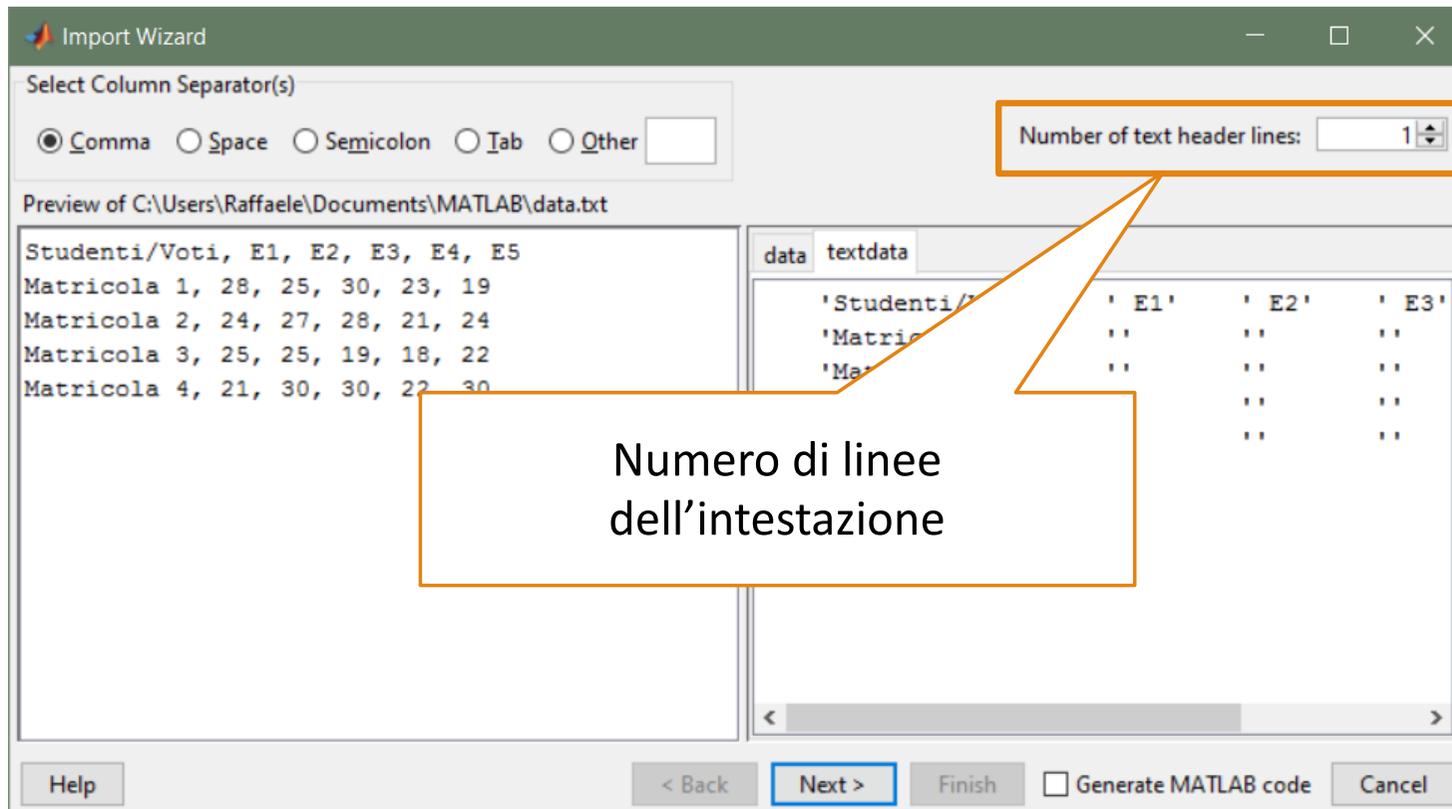
Importare Dati da File Tramite Interfaccia Grafica | Passaggio 1

```
>> uiimport('data.txt')
```



Importare Dati da File Tramite Interfaccia Grafica | Passaggio 1

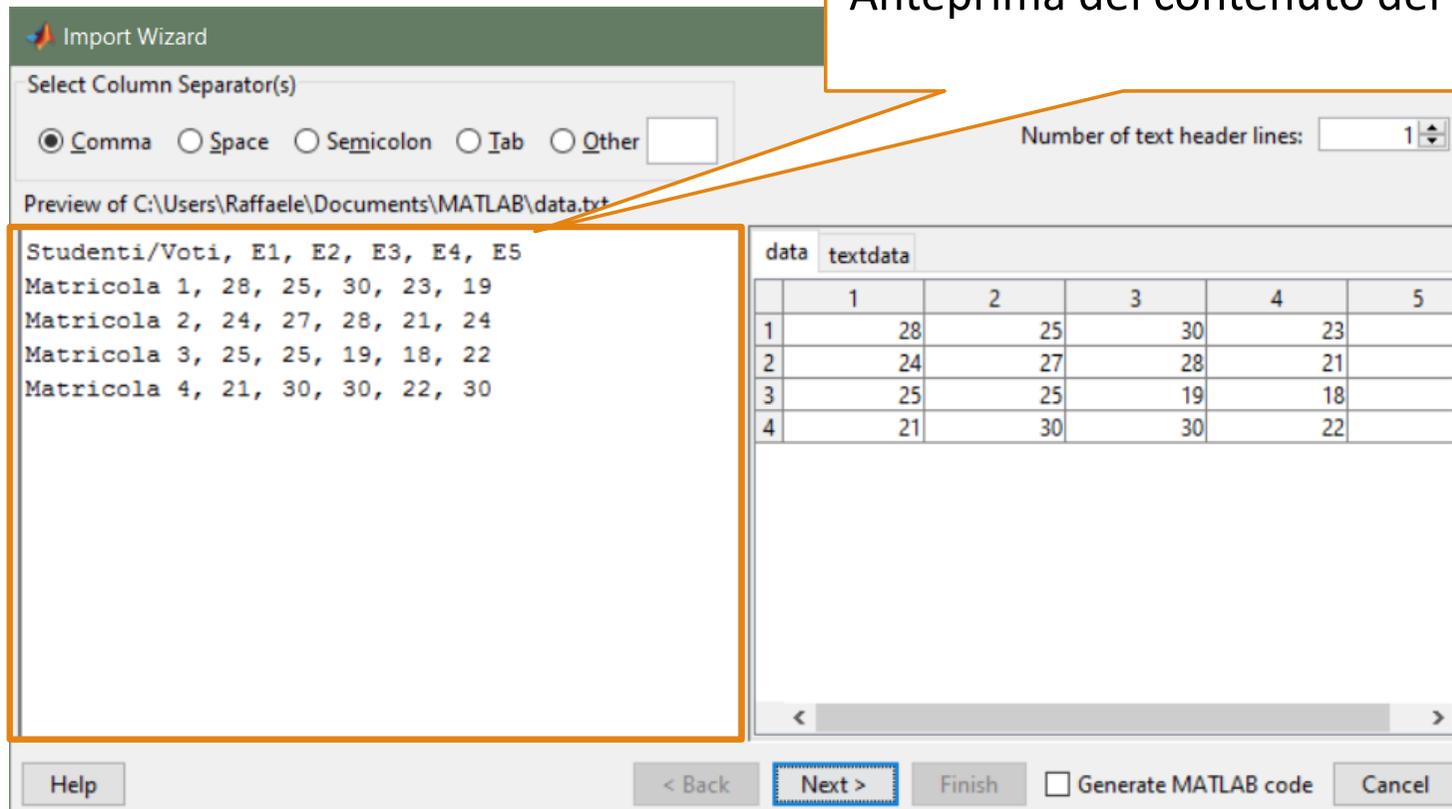
```
>> uiimport('data.txt')
```



Importare Dati da File Tramite Interfaccia Grafica | Passaggio 1

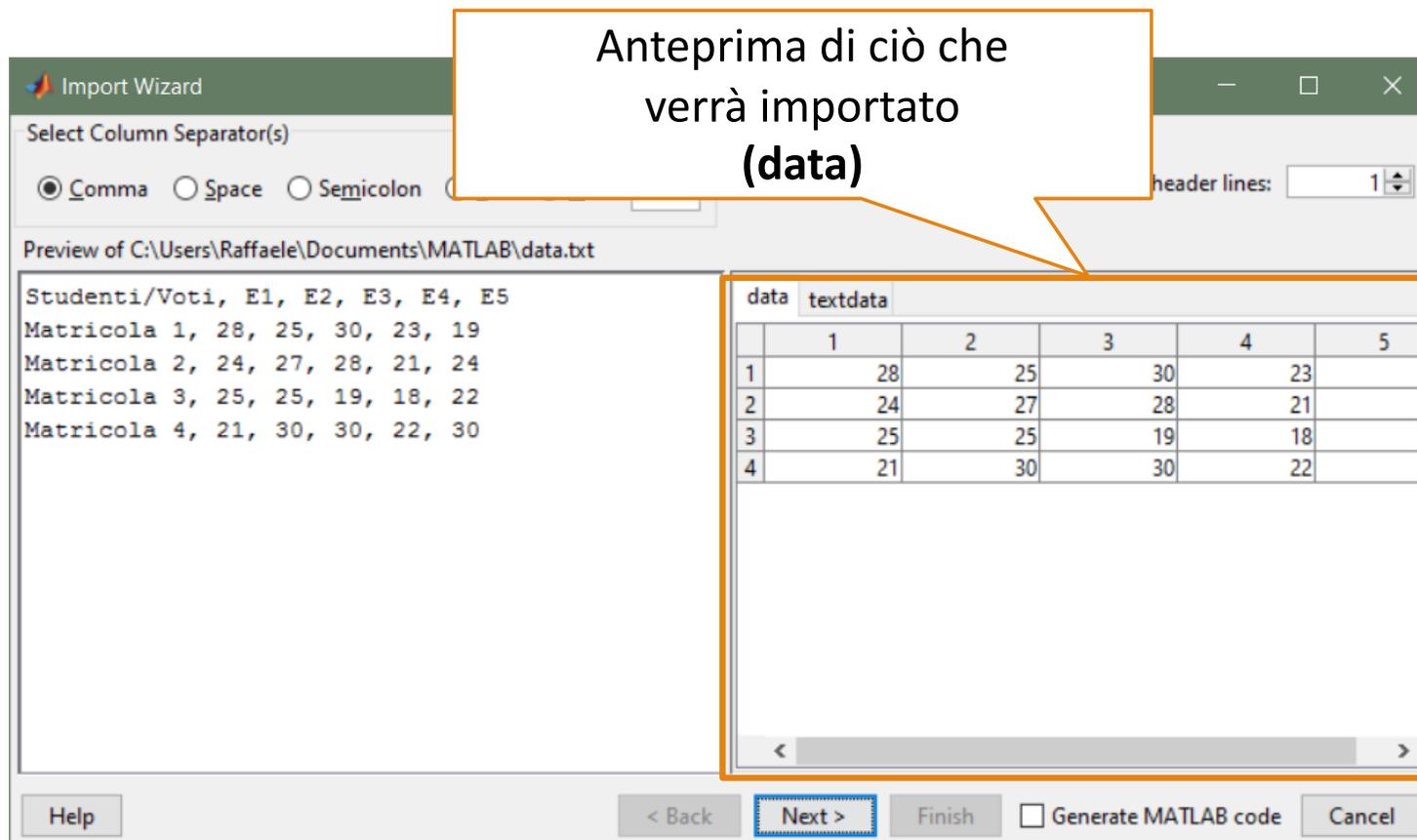
```
>> uiimport('data.txt')
```

Anteprima del contenuto del file



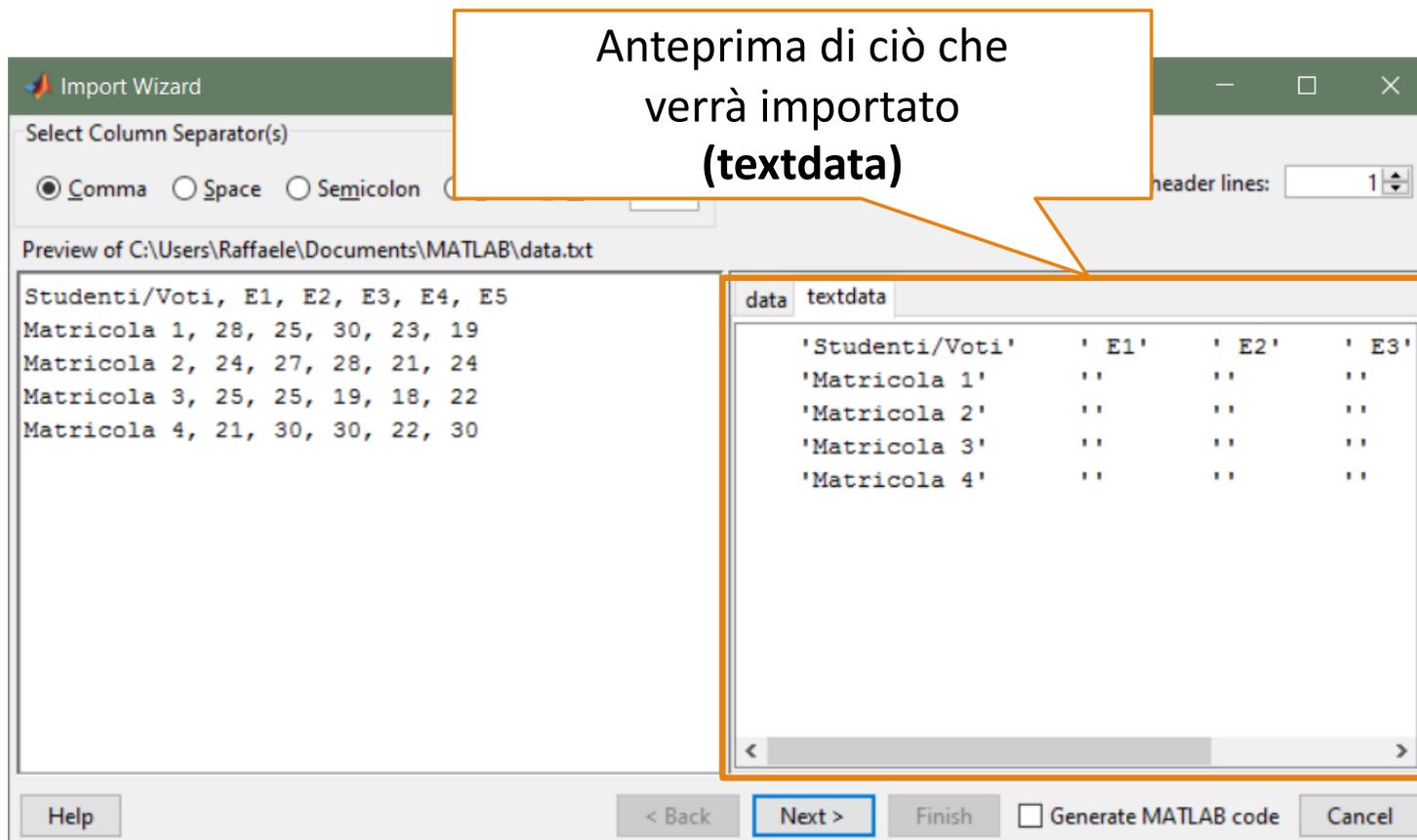
Importare Dati da File Tramite Interfaccia Grafica | Passaggio 1

```
>> uiimport('data.txt')
```



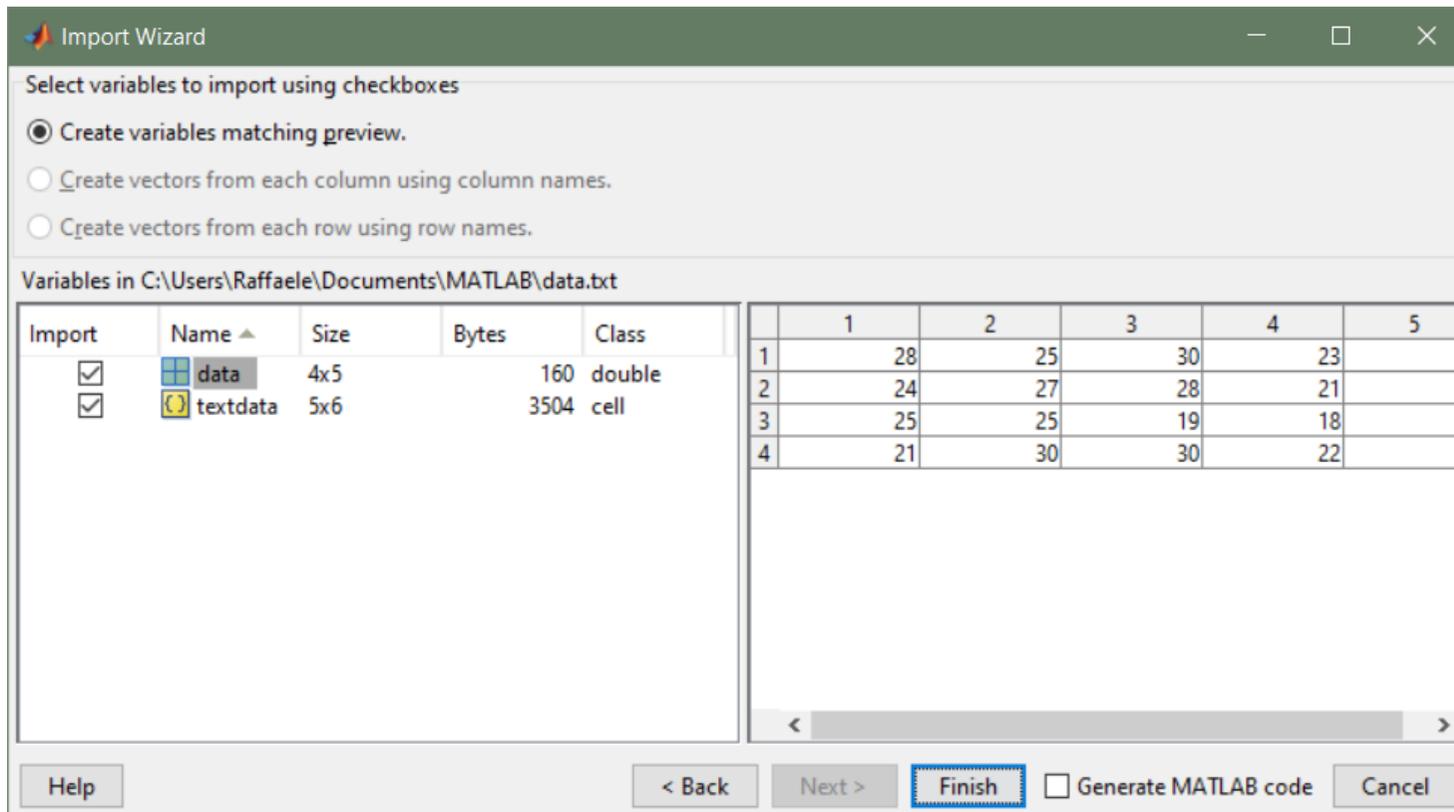
Importare Dati da File Tramite Interfaccia Grafica | Passaggio 1

```
>> uiimport('data.txt')
```



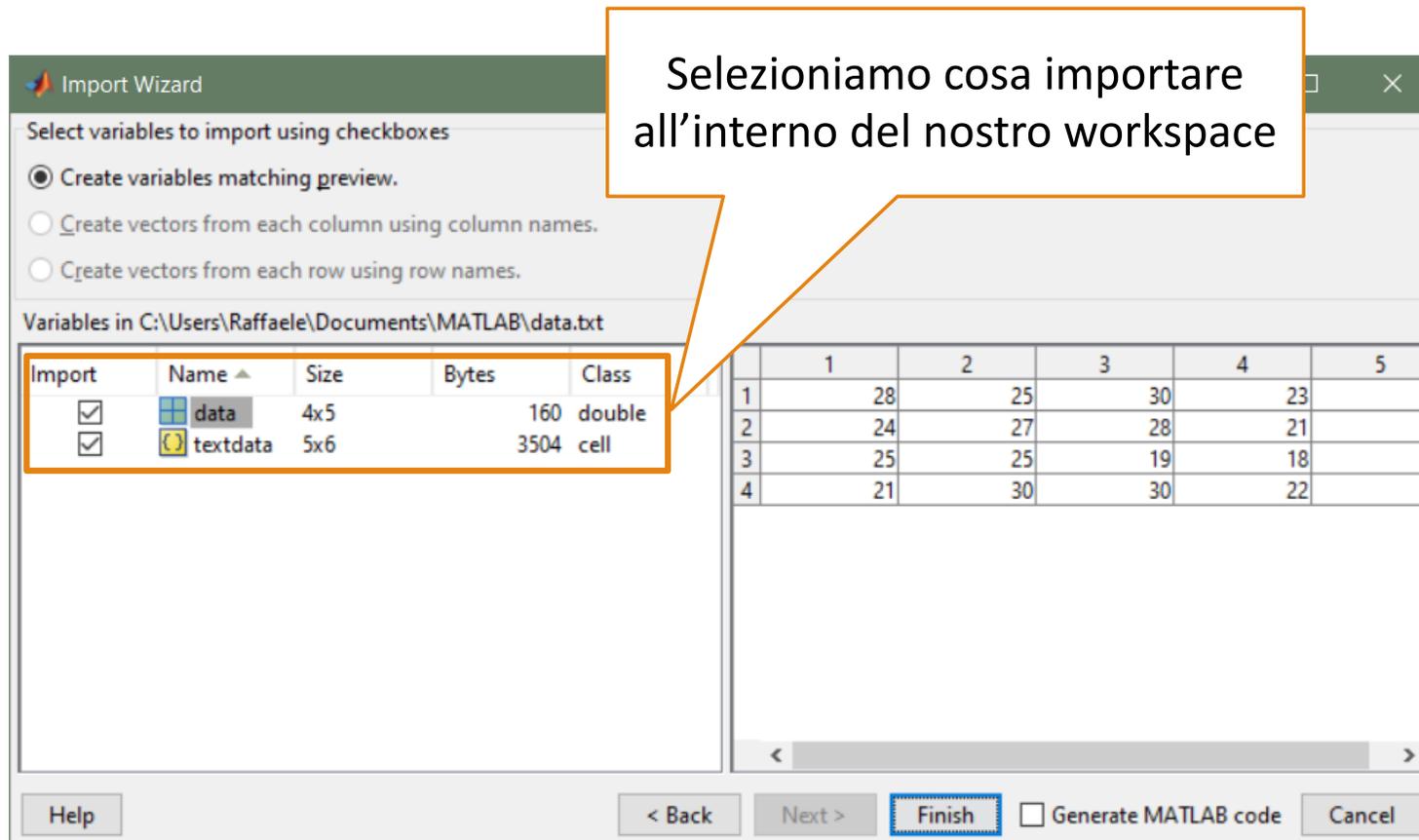
Importare Dati da File Tramite Interfaccia Grafica | Passaggio 2

```
>> uiimport('data.txt')
```



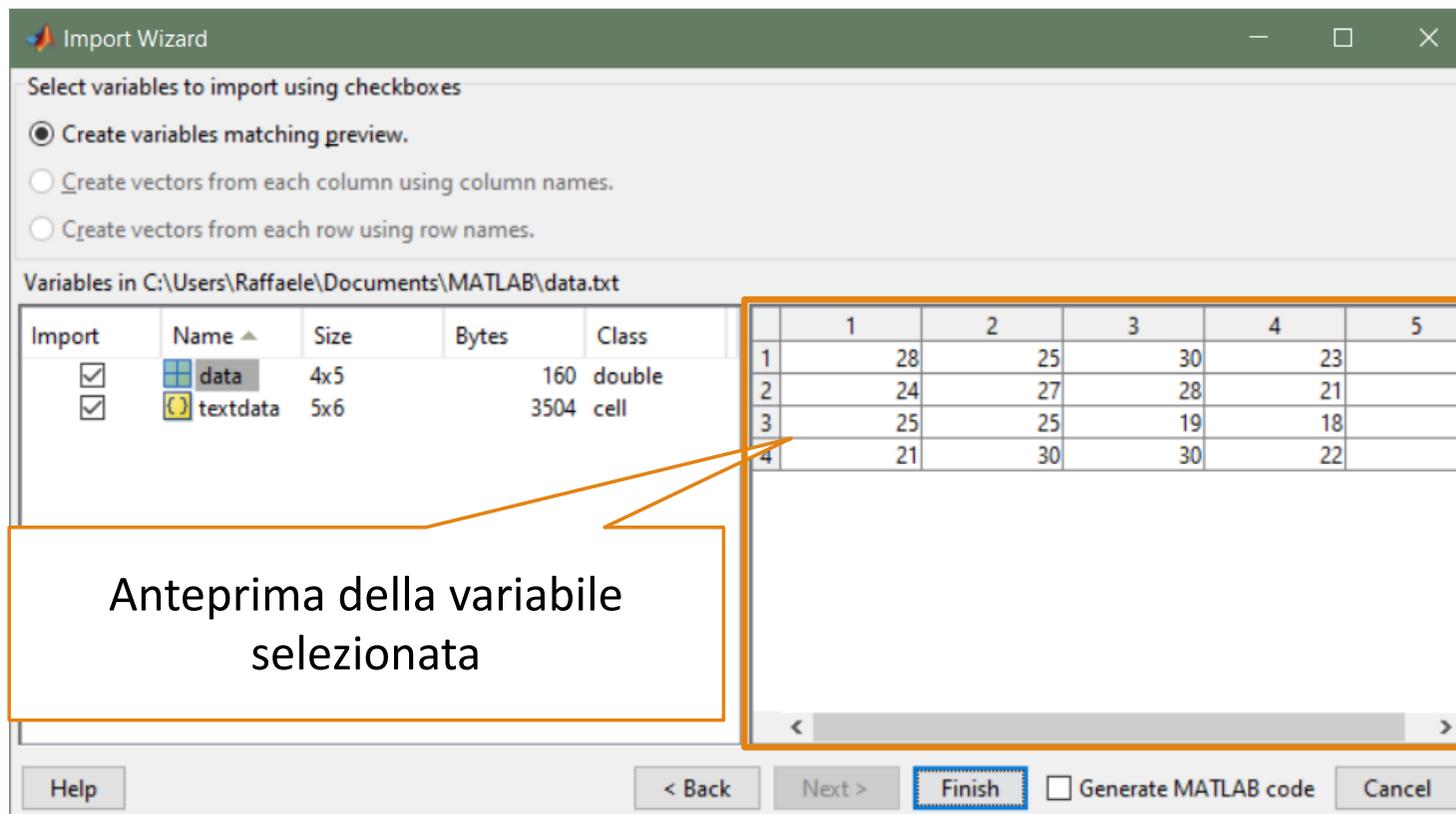
Importare Dati da File Tramite Interfaccia Grafica | Passaggio 2

```
>> uiimport('data.txt')
```



Importare Dati da File Tramite Interfaccia Grafica | Passaggio 2

```
>> uiimport('data.txt')
```



Importare Dati da File

Tramite Interfaccia Grafica | Passaggio 3

- Cliccando su **Finish**, nel nostro Workspace di MATLAB, saranno importate le variabili selezionate ed avranno i valori visti nell'anteprima

