

Gruppo 2 -Pseudocodice funzioni di struttura

FUNZIONE LeggerePlaylistDaTastiera()

INPUT	DESCRIZIONE	TIPO	VINCOLI
NuovoPlaylist	struttura contenente le informazioni della playlist da in serire in musica	Playlist	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
NuovoPlaylist	struttura contenente le informazioni aggiornate della playlist da in serire in musica	Playlist	
VARIABILI LAVORO	DESCRIZIONE	TIPO	VINCOLI
nomePlaylist	contiene il nome della playlist letto da tastiera	Array di caratteri	
descrizione	contiene la descrizione della playlist letta da tastiera	Array di caratteri	

LeggerePlaylistDaTastiera(NuovoPlaylist)

ESEGUI

Stampa a video "Inserisci il nome del Playlist musicale (almeno 1 carattere)> "

nomePlaylist <- LeggereStringa()

FINCHE' (LeggereLunghezzaStringa(nomePlaylist) < GEN_MIN)

ESEGUI

Stampa a video "Inserisci la descrizione della Playlist musicale (almeno 1 carattere)> "

descrizione <-LeggereStringa()

FINCHE'(LeggereLunghezzaStringa(descrizione) < GEN_MIN

NuovoPlaylist <- ScrivereNomePlaylist(NuovoPlaylist, nomePlaylist);

NuovoPlaylist <- ScrivereDescrizionePlaylist(NuovoPlaylist, descrizione);

fine

FUNZIONE LeggereUtente()

INPUT	DESCRIZIONE	TIPO	VINCOLI
BaseDati	Database in cui ricercare la l'utente	Database	
I	Posizione all'interno del vettore database in cui leggere l'elemento Utente	Intero	>=0
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
utenteLetto	Struttura utente ricercata	Utente	

LeggereUtente(Baseдати,i)

UtScelto;

UtScelto<-i-esimo utente di BaseDati Fine scrivereUtente(Baseдати,i,UtScelto)

i-esimo utente di BaseDati<-UtScelto

fine

FUNZIONE ScrivereUtente()

INPUT	DESCRIZIONE	TIPO	VINCOLI
BaseDati	Database in cui ricercare il la struttura dell'utente	Database	
I	Posizione all'interno del database in cui scrivere la struttura dell'utente	Intero	>=0
UtScelto	Struttura da scrivere all'interno del database	Utente	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
BaseDati	Database in cui è stata scritta la struttura Utente	Database	

ScrivereUtente(BaseDati, i, UtScelto)

i-esimo utente di BaseDati Fine scrivereUtente=UtScelto

FUNZIONE LeggereidUtente()

INPUT	DESCRIZIONE	TIPO	VINCOLI
UtScelto	Utente di cui leggere l'idUtente	Utente	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
idUtente	Intero contenente il valore id dell'utente richiesto	Intero	>=0

LeggereidUtente(UtScelto,idUtente)

idUtente<-0;

idUtente<-campo idUtente di UtScelto

fine

FUNZIONE ScrivereidUtente()

INPUT	DESCRIZIONE	TIPO	VINCOLI
UtScelto	Utente in cui scrivere il campo idUtente	Utente	
idUtente	idUtente da scrivere all'interno dell'Utente	Intero	>=0
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
UtScelto	Utente in cui è stato scritto il campo idUtente	Utente	

ScrivereidUtente(UtScelto,idUtente)

campo idUtente UtScelto<-idUtente

fine

FUNZIONE leggereNomeUtente()

INPUT	DESCRIZIONE	TIPO	VINCOLI
UtScelto	Utente di cui leggere il nomeUtente	Utente	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
nomeUtente	Stringa contenente il nome dell'utente richiesto	Stringa	STRING_MAX

LeggereNomeUtente(UtScelto)

nomeUtente;

copiarestringa(nomeUtente,campo nomeUtente di UtScelto)

fine

FUNZIONE ScrivereNomeUtente()

INPUT	DESCRIZIONE	TIPO	VINCOLI
UtScelto	Utente in cui scrivere il campo nomeUtente	Utente	
nomeUtente	nomeUtente da scrivere all'interno dell'Utente	Stringa	STRING_MAX
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
UtScelto	Utente in cui è stato scritto il campo nomeUtente	Utente	

ScrivereNomeUtente(nomeUtente,UtScelto)

copiarestringa(campo nomeUtente di UtScelto,nomeUtente)

fine

FUNZIONE LeggerePassword()

INPUT	DESCRIZIONE	TIPO	VINCOLI
UtScelto	Utente di cui leggere la password	Utente	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
Password	Stringa contenente la password dell'utente richiesto	Stringa	STRING_MAX

LeggerePassword(UtScelto)

password;

copiarestringa(password,campo password di UtScelto) fine

FUNZIONE ScriverePassword()

INPUT	DESCRIZIONE	TIPO	VINCOLI
UtScelto	Utente in cui scrivere il campo password	Utente	
Password	password da scrivere all'interno dell'Utente	Stringa	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
UtScelto	Utente in cui è stato scritto il campo password	Utente	

ScriverePassword(UtScelto>Password)

copiarestringa(campo password di UtScelto>Password)

fine

FUNZIONE LeggereAdmin()

INPUT	DESCRIZIONE	TIPO	VINCOLI
UtScelto	Utente di cui leggere lo stato di admin	Utente	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
Admin	intero contenente lo stato di admin dell'utente richiesto	intero	>=0

LeggereAdmin(UtScelto)

admin<-0;

admin<-campo admin di UtScelto

fine

FUNZIONE ScrivereAdmin()

INPUT	DESCRIZIONE	TIPO	VINCOLI
UtScelto	Utente in cui scrivere il campo admin	Utente	
admin	Stato di admin da scrivere all'interno dell'Utente	Intero	>=0
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
UtScelto	Utente in cui è stato scritto il campo admin	Utente	

ScrivereAdmin(UtScelto,admin)

campo admin di UtScelto <-admin

fine

FUNZIONE LeggereGenere()

INPUT	DESCRIZIONE	TIPO	VINCOLI
BaseDati	Database in cui leggere la struttura Genere	Database	
I	Posizione all'interno del database in cui cercare la struttura Genere	Intero	>=0
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
GenScelto	Struttura del genere richiesto	Genere	

LeggereGenere(BaseDati,i)

GenScelto;

GenScelto<-i-esimo genere di BaseDati

Fine

FUNZIONE ScrivereGenere()

INPUT	DESCRIZIONE	TIPO	VINCOLI
BaseDati	Database in cui ricercare il la struttura del genere	Database	
I	Posizione all'interno del database in cui scrivere la struttura Genere	Intero	>=0
GenScelto	Struttura da scrivere all'interno del database	Genere	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
BaseDati	Database in cui è stata scritta la struttura Genere	Database	

scrivereGenere (BaseDati,i,GenScelto)

i-esimo genere di BaseDati<-GenScelto

fine

FUNZIONE LeggereIdGenere()

INPUT	DESCRIZIONE	TIPO	VINCOLI
GenScelto	Genere di cui leggere l'idGenere	Genere	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
idGenere	intero contenente l' idGenere del Genere richiesto	intero	>=0

leggereidGenere(GenScelto)

idGenere;

idGenere<-campo idGenere di GenScelto fine

FUNZIONE ScrivereIdGenere()

INPUT	DESCRIZIONE	TIPO	VINCOLI
GenScelto	Genere in cui scrivere il campo idGenere	Genere	
idGenere	idGenere da scrivere all'interno della struttura Genere	Intero	>=0
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
GenScelto	Genere in cui è stato scritto il campo idGenere	Genere	

ScrivereidGenere(GenScelto,idGenere)

campo idGenere di GenScelto <-idGenere

Fine

FUNZIONE LeggereNomeGenere()

INPUT	DESCRIZIONE	TIPO	VINCOLI
GenScelto	Genere di cui leggere il nome	Genere	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
nomeGenere	stringa contenente il nome del Genere richiesto	stringa	STRING_MAX

LeggereNomeGenere(GenScelto)

nomeGenere

copiarestringa(nomeGenere,camponome di GenScelto)

fine

FUNZIONE ScrivereNomeGenere()

INPUT	DESCRIZIONE	TIPO	VINCOLI
GenScelto	Genere in cui scrivere il campo nomeGenere	Genere	
nomeGenere	Nome del genere da scrivere all'interno della struttura Genere	Stringa	STRING_MAX
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
GenScelto	Genere in cui è stato scritto il campo nome	Genere	

ScrivereNomeGenere(GenScelto,nomeGenere)

copiarestringa(campo nome di GenScelto,nomeGenere)

fine

FUNZIONE LeggereUltimoEsito()

INPUT	DESCRIZIONE	TIPO	VINCOLI
BaseDati	il database nel quale verrà letto l'ultimo esito inserito	Database	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
ultimoEsito	Ultimo esito registrato all'interno del database (se =0, allora l'ultima operazione è andata a buon fine, altrimenti se >0 allora in base al numero ci sarà un codice di errore)	Intero	>=0

LeggereUltimoEsito(BaseDati)

ultimoEsito<-0

ultimoEsito<-campo ultimoEsito di BaseDati

fine

FUNZIONE ScrivereUltimoEsito()

INPUT	DESCRIZIONE	TIPO	VINCOLI
BaseDati	il database nel quale verrà scritto l'ultimo esito inserito	Database	
ultimoEsito	Intero da inserire all'interno della	Intero	>=0
	struttura del database in base all'esito dell'ultima operazione		
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
BaseDati	Database aggiornato con l'esito dell'ultima operazione effettuata	Database	

ScrivereUltimoEsito(BaseDati, ultimoEsito)

campo ultimoEsito di BaseDati<-ultimoEsito

fine

FUNZIONE LeggereUtenteDaTasitera()

INPUT	DESCRIZIONE	TIPO	VINCOLI
NuovoUtente	Struttura dati vuota contenente i punti in cui inserire i nuovi dati	Utente	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
NuovoUtente	Struttura dati completa contenente i dati inseriti dall'utente e verificati (con id=0)	Utente	
VARIABILI LAVORO	DESCRIZIONE	TIPO	VINCOLI
I	Contatore crescente, e controlla i primi caratteri minimi	Intero	>0, >LUNG_MIN
J	Contatore crescente, e controlla se i primi caratteri sono tutti vuoti	Intero	>=0, >=LUNG_MIN
Nomeutente	Nome dell'utente inserito dall'utente	Stringa	Lunghezza minima:6 caratteri; lunghezza massima:30 caratteri
P password	Password dell'utente inserito dall'utente	Stringa	Lunghezza minima:6 caratteri; lunghezza massima:30 caratteri

LeggereUtenteDaTastiera(NuovoUtente)

i<-0;

j<-0;

nomeutente;

password;

stampa("inserisci il nome dell'utente")

esegui

nomeutente<-leggidatastiera() j<-0

i<-0

mentre (i<LUNG_MIN)

se(i-esima posizione di nomeutente =" ")

j<-j+1

fine

se (j=LUNG_MIN) //se tutti i primi 6 caratteri sono vuoti

stampa("inserisci un nome di almeno 6 caratteri!")

fine

finché (j<6) //per non essere fuorviante, vuol dire finché nei primi 6 posti non ci sia un carattere

stampa("bene, ora inserisci una password")

esegui

password<-leggidatastiera() j<-0

i<-0

mentre (i<LUNG_MIN)

se(i-esima posizione di password =" ")

j<-j+1

fine

se (j=LUNG_MIN) //se tutti i primi 6 caratteri sono vuoti

stampa("inserisci un nome di almeno 6 caratteri!")

fine

finché (j<LUNG_MIN) //per non essere fuorviante, vuol dire finché nei primi 6 posti non ci sia un carattere

stampa("questo utente deve essere amministratore? Premi 1 se si, altrimenti 0")

esegui

admin<-leggidatastiera()

se(admin<>1 AND admin<>0)

stampa("inserisci un valore valido! Premi 1 se l'utente deve essere admin, altrimenti 0")

fine

```

finché(admin<>1 AND admin<>0)
scrivere nomeUtente(NuovoUtente,nomeUtente)
scrivere password(NuovoUtente,password)
scrivere admin(NuovoUtente,admin)
fine

```

FUNZIONE LeggereGenereDaTastiera()

INPUT	DESCRIZIONE	TIPO	VINCOLI
NuovoGenere	Struttura dati vuota contenente i punti in cui inserire i nuovi dati	Genere	
OUTPUT	DESCRIZIONE	TIPO	VINCOLI
NuovoGenere	Struttura dati completa contenente i dati inseriti del genere e verificati (con id=0)	Genere	
VARIABILI LAVORO	DESCRIZIONE	TIPO	VINCOLI
I	Contatore crescente, e controlla i primi caratteri minimi	Intero	<0, >LUNG_MIN
J	Contatore crescente, e controlla se i primi caratteri sono tutti vuoti	Intero	<=0, >=LUNG_MIN
nomegenere	Nome dell'utente inserito dall'utente	Stringa	Lunghezza minima:1 carattere; lunghezza massima:30 caratteri

LeggereGenereDaTastiera(nuovoGenere)

```

i<-0;
j<-0;
nomegenere;
stampa("inserisci il nome del genere musicale")
esegui
    nomegenere<-leggidatastiera() j<-0
    mentre (i<GENERE_MIN)
        se(i-esima posizione di nomeutente =" ")
            j<-j+1
        fine
    se (j=GENERE_MIN) //se il primo carattere è vuoto
        stampa("inserisci un nome di almeno 6 caratteri!")
    fine
finché (j<GENERE_MIN) //per non essere fuorviante, vuol dire finché nel primo posto non ci sia un carattere

```

FUNZIONE “ LeggereVettoreBrano”

INPUT

BaseDati	Database	struttura contenente tutte le informazioni del sistema, da cui leggere il vettore Brano
----------	----------	---

OUTPUT

vettoreBrano	vettore Brano	vettore contenente i brani del database
--------------	---------------	---

DATI DI LAVORO

i	intero ≥ 0	contatore dell'iesimo elemento del vettore letto
---	-----------------	--

PSEUDOCODICE

$i \leftarrow 0$

Mentre($i < \text{ELEM_MAX}$)

lesimo elemento di vettoreBrano \leftarrow iesimo elemento di Brano di Basedati

$i \leftarrow i+1$

fine

FUNZIONE “ ScrivereVettoreBrano”

INPUT

vettoreBrano	vettore Brano	struttura contenente tutti i dati relativi ai brani	da inserire in BaseDati
--------------	---------------	---	-------------------------

BaseDati	Database	struttura contenente tutte le informazioni del sistema in cui inserire vettoreBrano
----------	----------	---

OUTPUT

BaseDati	Database	struttura contenente tutte le informazioni del sistema
----------	----------	--

DATI DI LAVORO

i	intero ≥ 0	contatore dell'iesimo elemento del vettore
---	-----------------	--

PSEUDOCODICE

$i \leftarrow 0$

Mentre($i < \text{ELEM_MAX}$)

iesimo elemento di Brano di Basedati \leftarrow iesimo elemento di vettoreBrano

$i \leftarrow i+1$

fine

FUNZIONE “ LeggereVettoreArtista”

INPUT

BaseDati	Database	struttura contenente tutte le informazioni del sistema da cui leggere il vettore Artista
----------	----------	--

OUTPUT

vettoreArtista	vettore Artista	struttura contenente tutti i dati relativi agli Artisti
----------------	-----------------	---

DATI DI LAVORO

i	intero ≥ 0	contatore dell'iesimo elemento del vettore letto
---	-----------------	--

PSEUDOCODICE

$i \leftarrow 0$

Mentre($i < \text{ELEM_MAX}$)

lesimo elemento di vettoreArtista \leftarrow iesimo elemento di Artista di BaseDati

$i \leftarrow i+1$

fine

FUNZIONE “ ScrivereVettoreArtista”

INPUT

vettoreArtista	vettore Artista	struttura contenente tutti i dati relativi agli Artisti da inserire in BaseDati
----------------	-----------------	---

BaseDati	Database	struttura contenente tutte le informazioni del sistema in cui
----------	----------	---

inserire vettoreArtista

OUTPUT

BaseDati Database struttura contenente tutte le informazioni del sistema

DATI DI LAVORO

i intero ≥ 0 contatore dell'iesimo elemento del vettore

PSEUDOCODICE

$i \leftarrow 0$

Mentre($i < \text{ELEM_MAX}$)

iesimo elemento di Artista di BaseDati \leftarrow iesimo elemento di vettoreArtista

$i \leftarrow i+1$

fine

FUNZIONE “ LeggereVettoreAlbum”

INPUT

BaseDati DataBase struttura contenente tutte le informazioni del sistema da cui leggere il vettore Album

OUTPUT

vettoreAlbum vettore Album struttura contenente tutti i dati relativi agli Album letto da BaseDati

DATI DI LAVORO

i intero ≥ 0 contatore dell'iesimo elemento del vettore letto

PSEUDOCODICE

$i \leftarrow 0$

Mentre($i < \text{ELEM_MAX}$)

iesimo elemento di vettoreAlbum \leftarrow iesimo elemento di Album di BaseDati

$i \leftarrow i+1$

fine

FUNZIONE “ ScrivereVettoreAlbum”

INPUT

vettoreAlbum vettore Album struttura contenente tutti i dati relativi agli Album da inserire in BaseDati

BaseDati Database struttura contenente tutte le informazioni del sistema in cui inserire vettoreAlbum

OUTPUT

BaseDati Database struttura contenente tutte le informazioni del sistema

DATI DI LAVORO

i intero ≥ 0 contatore dell'iesimo elemento del vettore

PSEUDOCODICE

$i \leftarrow 0$

Mentre($i < \text{ELEM_MAX}$)

iesimo elemento di Album di BaseDati \leftarrow iesimo elemento di vettoreAlbum

$i \leftarrow i+1$

fine

FUNZIONE “ LeggereVettoreBranoAlbum”

INPUT

BaseDati Database struttura contenente tutte le informazioni del sistema da cui leggere il vettore BranoAlbum

OUTPUT

vettoreBranoAlbum vettore BranoAlbum vettore di strutture di collegamento tra i Brani e Album letto da BaseDati

DATI DI LAVORO

i intero ≥ 0 contatore dell'iesimo elemento del vettore letto

PSEUDOCODICE

$i \leftarrow 0$

Mentre($i < \text{ELEM_MAX}$)

lesimo elemento di vettoreBranoAlbum \leftarrow iesimo elemento di BranoAlbum di BaseDati

$i \leftarrow i + 1$

fine

FUNZIONE “ ScrivereVettoreBranoAlbum”

INPUT

vettoreBranoAlbum vettore BranoAlbum vettore di strutture di collegamento tra i Brani e Album da scrivere in BaseDati

BaseDati Database struttura contenente tutte le informazioni del sistema in cui scrivere vettoreBranoAlbum

OUTPUT

BaseDati Database struttura contenente tutte le informazioni del sistema

DATI DI LAVORO

i intero ≥ 0 contatore dell'iesimo elemento del vettore

PSEUDOCODICE

$i \leftarrow 0$

Mentre($i < \text{ELEM_MAX}$)

iesimo elemento di BranoAlbum di BaseDati \leftarrow iesimo elemento di vettoreBranoAlbum

$i \leftarrow i + 1$

fine

FUNZIONE “ LeggereVettoreBranoArtista”

INPUT

BaseDati Database struttura contenente tutte le informazioni del sistema da cui leggere il vettore BranoArtista

OUTPUT

vettoreBranoArtista vettore BranoArtista vettore di strutture di collegamento tra i Brani e Artista letto da BaseDati

DATI DI LAVORO

i intero ≥ 0 contatore dell'iesimo elemento del vettore letto

PSEUDOCODICE

$i \leftarrow 0$

Mentre($i < \text{ELEM_MAX}$)

lesimo elemento di vettoreBranoArtista \leftarrow iesimo elemento di BranoArtista di BaseDati

$i \leftarrow i + 1$

fine

FUNZIONE “ ScrivereVettoreBranoArtista”

INPUT

vettoreBranoArtista vettore BranoArtista vettore di strutture di collegamento tra i Brani e Artista da scrivere in BaseDati

BaseDati DataBase struttura contenente tutte le informazioni del sistema in cui inserire vettoreBranoArtista

OUTPUT

BaseDati DataBase struttura contenente tutte le informazioni del sistema

DATI DI LAVORO

i intero ≥ 0 contatore dell'iesimo elemento del vettore

PSEUDOCODICE

$i \leftarrow 0$

Mentre($i < \text{ELEM_MAX}$)

iesimo elemento di BranoArtista di BaseDati \leftarrow iesimo elemento di vettoreBranoArtista

i ← i+1
fine

FUNZIONE “ LeggereBranoGenere”

INPUT

vettoreBranoGenere	vettore BranoGenere	vettore di strutture di collegamento tra i Brani e Generi da cui leggere l'iesimo elemento
i	intero >= 0	posizione dell'elemento da leggere in vettoreBranoGenere

OUTPUT

gen	BranoGenere	iesimo elemento di vettoreBranoGenere
-----	-------------	---------------------------------------

PSEUDOCODICE

gen ← iesimo elemento di vettoreBranoGenere

FUNZIONE “ LeggereIdBranoBranoGenere”

INPUT

gen	BranoGenere	struttura BranoGenere da cui leggere il campo idBrano
-----	-------------	---

OUTPUT

id	intero >= 0	campo idBrano letto da gen
----	-------------	----------------------------

PSEUDOCODICE

id ← campo idBrano di gen

FUNZIONE “ LeggereIdGenereBranoGenere”

INPUT

gen	BranoGenere	struttura BranoGenere da cui leggere il campo idGenere
-----	-------------	--

OUTPUT

id	intero >= 0	campo idGenere letto da gen
----	-------------	-----------------------------

PSEUDOCODICE

id ← campo idGenere di gen

FUNZIONE “ ScrivereIdBranoBranoGenere”

INPUT

gen	BranoGenere	struttura BranoGenere in cui scrivere il campo idBrano
id	intero >= 0	valore da inserire al campo idBrano di gen

OUTPUT

gen	BranoGenere	struttura BranoGenere aggiornata
-----	-------------	----------------------------------

PSEUDOCODICE

campo idBrano di gen ← id

FUNZIONE “ ScrivereIdGenereBranoGenere”

INPUT

gen	BranoGenere	struttura BranoGenere in cui scrivere il campo idGenere
id	intero >= 0	valore da inserire al campo idGenere di gen

OUTPUT

gen	BranoGenere	struttura BranoGenere aggiornata
-----	-------------	----------------------------------

PSEUDOCODICE

campo idGenere di gen ← id

FUNZIONE Brano LeggereBrano()

INPUT

i	intero >= 0	indice di posizione
vettoreBrano	vettore brano	struttura brano da cui leggere

OUTPUT

br	struttura Brano	contiene dati relativi al brano letto
----	-----------------	---------------------------------------

PSEUDOCODICE

br = iesimo elemento di vettoreBrano

fine

FUNZIONE LeggereIdBrano()

INPUT

br	struttura brano	contiene i dati relativi al brano letto
OUTPUT		
Id	intero >=0	contiene l'id di un determinato brano

PSEUDOCODICE
Id = campo idBrano di Brano
Fine

FUNZIONE ScrivereIdBrano()**INPUT**

br	struttura Brano	contiene i dati relativi al brano da scrivere
id	intero >=0	contiene l'id di un determinato brano

OUTPUT

Br	struttura Brano	contiene i dati relativi al brano scritto
----	-----------------	---

PSEUDOCODICE
Campo idBrano di Brano ← id
Fine

FUNZIONE LeggereNomeBrano()**INPUT**

br	struttura Brano	contiene i dati relativi al nome del brano da leggere
parola	stringa	nome del brano

OUTPUT

br	struttura Brano	contiene i dati relativi al nome del brano
----	-----------------	--

PSEUDOCODICE
parola ← Copia stringa(parola, campo nome di Brano)
Fine

FUNZIONE ScrivereNomeBrano ()**INPUT**

br	struttura Brano	contiene i dati relativi al nome del brano che verrà scritto
parola	stringa	nome del brano da scrivere

OUTPUT

br	struttura Brano	contiene il nome i dati relativi ad un brano, col nome aggiornato
----	-----------------	---

PSEUDOCODICE
campo nome di Brano ← Copia stringa(campo nome di Brano, parola)
Fine

FUNZIONE LeggereDurataBrano()**INPUT**

br	struttura brano	contiene i dati relativi al brano letto
----	-----------------	---

OUTPUT

durata	intero >=0	contiene la durata di un determinato brano
--------	------------	--

PSEUDOCODICE

durata = campo durata di Brano
Fine

FUNZIONE ScrivereDurataBrano()**INPUT**

br	struttura Brano	contiene i dati relativi al brano da scrivere
durata	intero >=0	contiene la durata di un determinato brano

OUTPUT

Br	struttura Brano	contiene i dati relativi al brano
----	-----------------	-----------------------------------

PSEUDOCODICE
Campo durata di Brano ← durata
Fine

FUNZIONE LeggereAnnoBrano()**INPUT**

```

br                                struttura brano                                contiene i dati relativi al brano letto
OUTPUT
anno                                intero >1000                                contiene l'anno d'uscita di un brano
PSEUDOCODICE
anno = campo anno di Brano
Fine

```

INPUT

```

br          struttura Brano          contiene i dati relativi al brano da scrivere
anno       intero > 1000            contiene l'anno d'uscita di un brano
OUTPUT
br          struttura Brano          contiene i dati relativi al brano
PSEUDOCODICE
Campo anno di Brano ← anno
Fine

```

INPUT

```

br                struttura brano                contiene i dati relativi al brano letto
OUTPUT
ascolti           intero>=0                      contiene gli ascolti di un brano
PSEUDOCODICE
ascolti = campo ascolti di Brano
Fine

```

INPUT

```

br          struttura Brano          contiene i dati relativi al brano da scrivere
ascolti     intero>=0               contiene gli ascolti di un brano
OUTPUT
Br          struttura Brano          contiene i dati relativi al brano
PSEUDOCODICE
Campo ascolti di Brano ← ascolti
Fine

```

INPUT

i	intero ≥ 0	indice di posizione
vettoreAlbum	vettore Album	struttura Album da cui leggere
OUTPUT		
al	struttura Album	contiene dati relativi all' Album letto
PSEUDOCODICE		
al= iesimo elemento di vettoreAlbum		
fine		

INPUT

al	struttura Album	contiene i dati relativi all'album letto
OUTPUT		
Id	intero >=0	contiene l'id di un determinato Album
PSEUDOCODICE		
Id = campo idAlbum di Album		
Fine		

INPUT

al	struttura Album	contiene i dati relativi all'Album da scrivere
id	intero >=0	contiene l'id di un determinato Album

OUTPUT

al	struttura Album	contiene i dati relativi all'Album scritto
----	-----------------	--

PSEUDOCODICE

Campo idAlbum di Album ← id

Fine

FUNZIONE LeggereTitoloAlbum()

INPUT

al	struttura Brano	contiene i dati relativi al nome del brano da leggere
parola	stringa	nome del brano

OUTPUT

al	struttura Album	contiene i dati relativi al titolo dell'album
----	-----------------	---

PSEUDOCODICE

parola ← Copia stringa(parola, campo titolo di Album)

Fine

FUNZIONE ScrivereTitoloAlbum()

INPUT

al	struttura Album	contiene i dati relativi al titolo dell'Album che verrà scritto
parola	stringa	titolo dell'Album da scrivere

OUTPUT

al	struttura Album	contiene il titolo e i dati relativi ad un Album, col nome aggiornato
----	-----------------	---

PSEUDOCODICE

campo titolo di Album ← Copia stringa(campo titolo di Album, parola)

Fine

FUNZIONE LeggereAnnoAlbum()

INPUT

al	struttura Album	contiene i dati relativi all'Album letto
----	-----------------	--

OUTPUT

anno	intero > 1000	contiene l'anno d'uscita di un Album
------	---------------	--------------------------------------

PSEUDOCODICE

anno = campo anno di Album

Fine

FUNZIONE ScrivereAnnoAlbum()

INPUT

al	struttura Album	contiene i dati relativi all'Album da scrivere
anno	intero > 1000	contiene l'anno d'uscita di un Album

OUTPUT

al	struttura Brano	contiene i dati relativi all'Album
----	-----------------	------------------------------------

PSEUDOCODICE

Campo anno di Album ← anno

Fine

FUNZIONE LeggereArtista()

INPUT

i	intero	indice di posizione
vettoreArtista	vettore Artista	struttura Artista da cui leggere

OUTPUT

ar	struttura Artista	contiene dati relativi all'Artista letto
----	-------------------	--

PSEUDOCODICE

ar= iesimo elemento di vettoreArtista

fine

FUNZIONE LeggereIdArtista()

INPUT

ar struttura Artista

contiene i dati relativi all'Artista letto

OUTPUT

Id intero >=0

contiene l'id di un determinato Artista

Id = campo idArtista di Artista

Fine

FUNZIONE ScrivereIdArtista()**INPUT**

ar struttura Artista

contiene i dati relativi all'Artista da scrivere

id intero >=0

contiene l'id di un determinato Artista

OUTPUT

ar struttura Artista

contiene i dati relativi all'Artista scritto

PSEUDOCODICE

Campo idArtista di Artista ← id

Fine

FUNZIONE LeggereNomeArtista()**INPUT**

ar struttura Artista contiene i dati relativi al nome dell'Artista da leggere

parola stringa nome dell'Artista

OUTPUT

ar struttura Artista contiene i dati relativi al nome dell'Artista

PSEUDOCODICE

parola ← Copia stringa(parola, campo nome di Artista)

Fine

FUNZIONE : "LeggereNomeArtista"**Input:**

ar struttura di tipo Artista da cui leggere il campo nome Artista

Output:

parola nome dell'artista stringa

Pseudocodice:

parola ← Copia(parola , campo nome di ar)

FUNZIONE : "ScrivereNomeArtista"**Input:**

ar struttura di tipo Artista in cui scrivere il campo nome Artista

parola nome dell'artista stringa

Output:

ar struttura di tipo Artista Artista

Pseudocodice:

campo nome di ar ← Copia(campo nome di ar , parola)

FUNZIONE : "LeggereCognomeArtista"**Input:**

ar struttura di tipo Artista da cui leggere il campo cognome Artista

Output:

parola nome dell'artista stringa

Pseudocodice:

parola ← Copia(parola , campo cognome di ar)

FUNZIONE : "ScrivereCognomeArtista"**Input:**

ar struttura di tipo Artista in cui scrivere il campo cognome Artista

parola cognome dell'artista stringa

Output:

ar struttura di tipo Artista in cui è stato scritto il campo cognome Artista
Pseudocodice:
campo cognome di ar \leftarrow Copia(campo cognome di ar , parola)

FUNZIONE : “LeggereNomeArteArtista”

Input:
ar struttura di tipo Artista da cui leggere il campo NomeArte Artista
Output:
parola cognome dell’artista stringa
Pseudocodice:
parola \leftarrow Copia(parola , campo NomeArte di ar)

FUNZIONE : “ScrivereNomeArteArtista”

Input:
ar struttura di tipo Artista in cui scrivere il campo NomeArte Artista
parola nome d’arte dell’artista stringa
Output:
ar struttura di tipo Artista in cui è stato scritto il campo NomeArte Artista
Pseudocodice:
campo NomeArte di ar \leftarrow Copia(campo NomeArte di ar , parola)

FUNZIONE : “LeggereBranoAlbum”

Input:
vettoreBranoAlbum vettore contenente strutture di tipo BranoAlbum BranoAlbum
Output:
ba struttura di tipo BranoAlbum BranoAlbum
Lavoro:
i contatore del vettore vettoreBranoAlbum intero ≥ 0
Pseudocodice:
ba \leftarrow i-esimo elemento di vettoreBranoAlbum

FUNZIONE : “LeggereIdBranoBranoAlbum”

Input:
ba struttura di tipo BranoAlbum da cui leggere il campo idBrano BranoAlbum
Output:
id il campo idBrano di ba BranoAlbum
Pseudocodice:
id \leftarrow campo idBrano di ba

FUNZIONE : “ScrivereIdBranoBranoAlbum”

Input:
ba struttura di tipo BranoAlbum da cui leggere il campo idBrano BranoAlbum
id numero identificativo dei brani da scrivere in ba intero ≥ 0
Output:
ba struttura di tipo BranoAlbum BranoAlbum
Pseudocodice:
campo idBrano di Ba \leftarrow id

FUNZIONE : “LeggereIdAlbumBranoAlbum”

Input:
ba struttura di tipo BranoAlbum da cui leggere il campo idAlbum BranoAlbum
Output:
id numero identificativo dei brani letto in ba intero ≥ 0
Pseudocodice:
id \leftarrow campo idAlbum di ba

FUNZIONE : “ScrivereIdAlbumBranoAlbum”

Input:
ba struttura di tipo BranoAlbum in cui scrivere il campo idAlbum BranoAlbum

id	numero identificativo dei brani	intero	>=0
Output:			
ba	struttura di tipo BranoAlbum in cui scrivere il campo idAlbum	BranoAlbum	
Pseudocodice:			
campo idAlbum di ba	← id		

Input:			
vettoreBranoArtista	vettore contenente strutture di tipo BranoArtista	BranoArtista	
i	posizione del vettore vettoreBranoArtista	intero	>=0
Output:			
ba	struttura di tipo BranoAlbum	BranoAlbum	
Pseudocodice:			
ba = i-esimo elemento di vettoreBranoArtista			

Input:			
ba	struttura di tipo BranoAlbum da cui leggere il campo idBrano BranoAlbum		
Output:			
id	il campo idBrano di ba	intero	>=0
Pseudocodice:			
id ← campo idBrano di ba			

Input:

ba	struttura di tipo BranoAlbum in cui scrivere il campo idBrano	BranoAlbum	
id	numero identificativo dei brani	intero	>=0

Output:

ba	struttura di tipo BranoAlbum in cui scrivere il campo idBrano	BranoAlbum	
----	---	------------	--

Pseudocodice:

campo idBrano di ba \leftarrow id

Input:			
ba	struttura di tipo BranoAlbum da cui leggere il campo idArtista	BranoAlbum	
Output:			
id	il campo idArtista di ba	intero	>=0
Pseudocodice:			
id	← campo diArtista di ba		

Input:			
ba	struttura di tipo BranoAlbum in cui scrivere il campo idArtistaBranoAlbum		
id	numero identificativo dei brani	intero	>=0
Output:			
ba	struttura di tipo BranoAlbum in cui scrivere il campo idArtistaBranoAlbum		
Pseudocodice:			
campo idArtista di ba	←	id	

Input:			
vettorePlaylistBrano	vettore contenente strutture di tipo PlaylistBrano	PlaylistBrano	
i	posizione del vettore vettorePlaylistBrano	intero	>=0
Output:			
pl	struttura contenente le tabelle di PlaylistBrano	PlaylistBrano	
Pseudocodice:			
pl = i-esimo elemento di vettorePlaylistBrano			

Input:

pl struttura di tipo PlaylistBranò da cui leggere il campo idBranò PlaylistBranò
Output:
id il campo idBranò di pl intero >=0
Pseudocodice:
id ← campo idBranò di pl

FUNZIONE : “ScrivereIdBranòPlaylistBranò”

Input:
pl struttura di tipo PlaylistBranò in cui scrivere il campo idBranò PlaylistBranò
id numero identificativo dei brani intero >=0
Output:
pl struttura di tipo PlaylistBranò in cui scrivere il campo idBranò PlaylistBranò
Pseudocodice:
campo idBranò di pl ← id

FUNZIONE : “LeggereIdPlaylistPlaylistBranò”

Input:
pl struttura di tipo PlaylistBranò da cui leggere il campo idPlaylist PlaylistBranò
Output:
id il campo idPlaylist di pl intero >=0
Pseudocodice:
id ← campo idPlaylist di pl

FUNZIONE : “ScrivereIdPlaylistPlaylistBranò”

Input:
pl struttura di tipo PlaylistBranò in cui scrivere il campo idPlaylist PlaylistBranò
id numero identificativo delle Playlist intero >=0
Output:
pl struttura di tipo PlaylistBranò in cui scrivere il campo idPlaylist PlaylistBranò
Pseudocodice:
campo idPlaylist di pl ← id
