

Esercitazione 02: JDBC, SQL e DB SAMPLE

Sistemi Informativi L-A

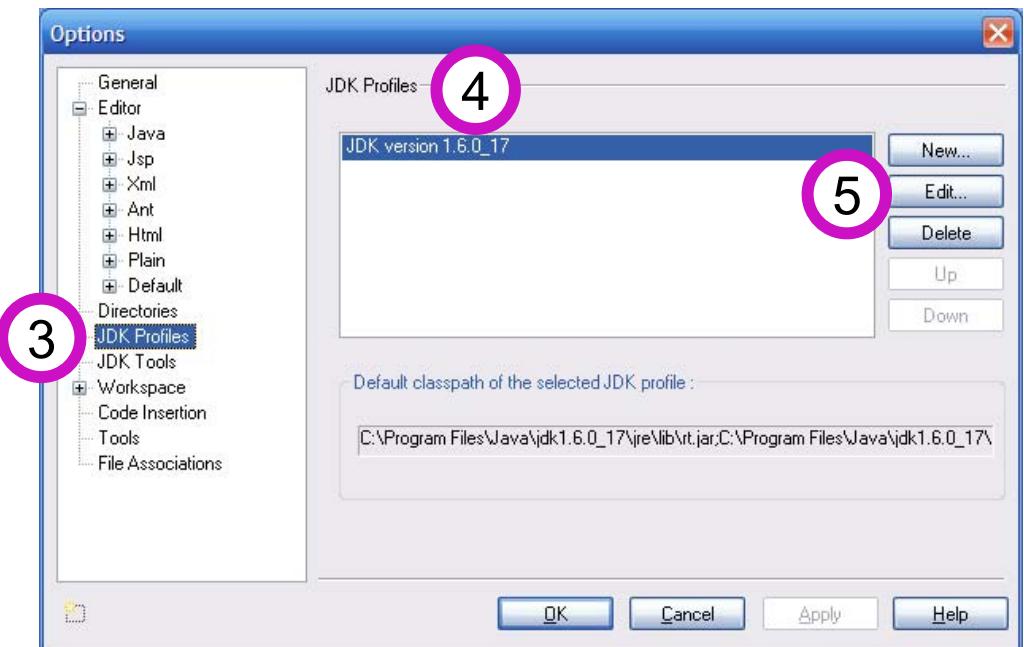
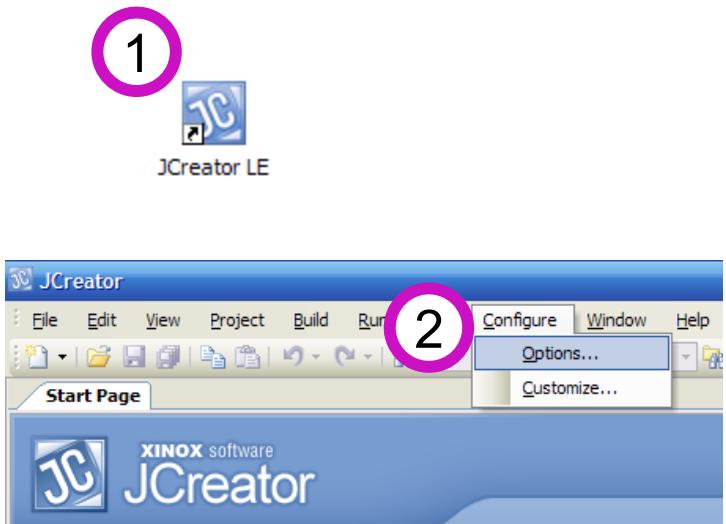
Home Page del corso:

<http://www-db.deis.unibo.it/courses/SIL-A/>

Versione elettronica: [Esercitazione02.pdf](#)

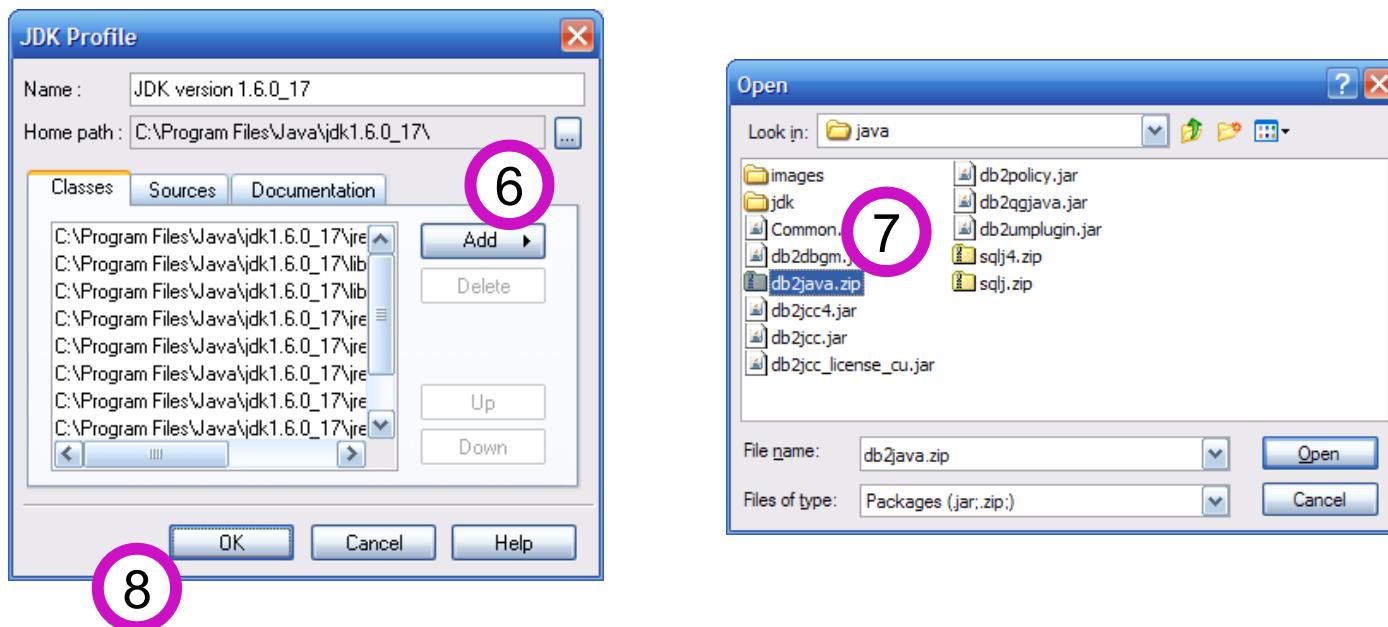
Come configurare JCREATOR (1/3)

1. Avviare JCreator
2. Dal menu **Configure** selezionare **Options**
3. Nella finestra di dialogo dalla voce **Editor** selezionare **JDK Profiles**
4. Selezionate la JDK attualmente in uso (ad es.: JDK version 1.6.0_17)
5. Cliccate su **Edit**



Come configurare JCREATOR (2/3)

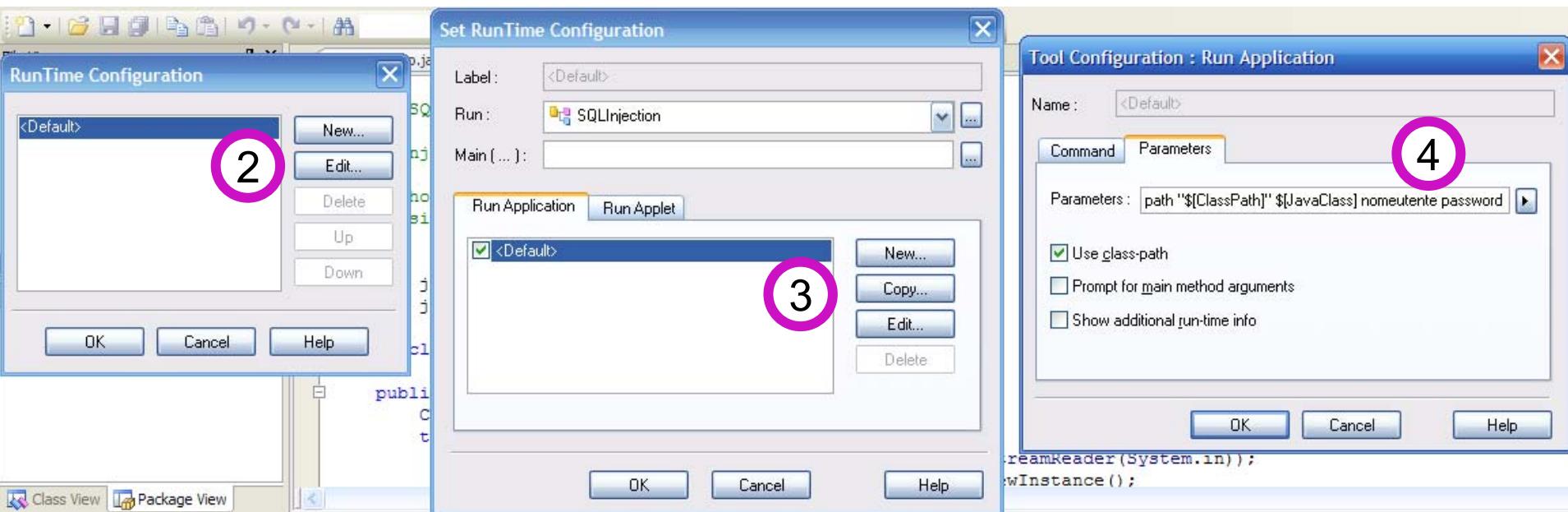
6. Cliccare su **Add** → **Add Archive**
7. Selezionare il .zip contenente il driver JDBC di DB2
 - C:\Programmi\IBM\SQLLIB\java\db2java.zip
8. Confermare tutte le scelte cliccando su **OK**.



Come configurare JCREATOR (3/3)

Può essere utile passare nome utente e password come parametri alla classe main del programma. Per impostare tali parametri in un progetto:

1. Dal menu **Run** selezionare **Run Configuration**
2. Selezionare la configurazione **<Default>** e cliccare su **Edit**
3. Nella linguetta **Run Application** selezionare **<Default>** e cliccare su **Edit**
4. Nella linguetta **Parameters** aggiungere in coda ai parametri il proprio nome utente e password
5. Confermare cliccando su **OK**



Applicazione Java **Videonoleggio** (1)

- Scrivere l'applicazione Java **“Videonoleggio”** che, caricando l'opportuno driver JDBC, si connette al database **STUDENTI** e lavora sulle tabelle **CLIENTI**, **DVD** e **NOLEGGI** relative all'esercizio dell'esercitazione scorsa.

In particolare, l'applicazione deve:

1. Inserire nella tabella **CLIENTI** le tuple:

```
(‘001’, ‘Carlo’, ‘Rossi’, ‘+390596606633’)
```

```
(‘002’, ‘Giovanni’, ‘Bianchi’, ‘+390516706103’)
```

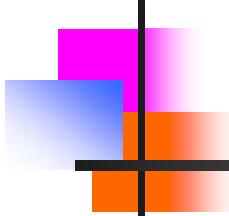
2. Aggiornare la tabella **CLIENTI** relativamente alla tupla con tessera **‘002’** ponendo l'attributo **TELEFONO** uguale a **‘+393385577890’**
3. Cancellare dalla tabella **CLIENTI** le tuple con cognome **“Bianchi”**

N.B. Produrre un opportuno messaggio su standard output relativo al risultato ottenuto ai punti precedenti

Applicazione Java **videonoleggio** (2)

Eseguire le interrogazioni riportate di seguito producendo su standard output la stampa (opportunamente formattata) dei dati ottenuti:

1. “Numero complessivo di noleggi relativi al dvd noleggiato più volte”
2. “Codice e trama dei dvd per cui non ci sono noleggi”
3. “Numero Tessera e numero totale di dvd noleggiati relativi ai clienti che hanno noleggiato almeno 2 dvd”



...per i veri duri

- Provare a costruire un metodo con la seguente *signature*:
 - `public void PrintFile(ResultSet rs, String filename)`

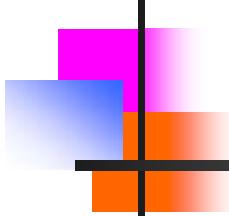
utilizzando l’interfaccia `ResultSetMetaData`, che viene restituita dal metodo `<ResultSetObj>.getMetaData()`.

Il metodo dovrà prendere in ingresso un `ResultSet` ed il nome del file di output e scrivere su quest’ultimo il contenuto di tale `ResultSet` formattato in maniera opportuna, senza conoscere a priori la struttura della tabella (numero di colonne, nomi, tipi di dato ecc...).

Se il nome del file ha valore ‘`null`’ la stampa avverrà direttamente a video.

Query database “SAMPLE” (1)

- Con riferimento al database **SAMPLE** :
 - Connetersi al database
CONNECT TO SAMPLE USER xxxxxxx
 - Autenticarsi al server immettendo la **password personale**
 - Risolvere le seguenti interrogazioni **mediante CLP e file di input e di output**:
 1. “Impiegati donna del dipartimento ‘D11’ con un salario annuale superiore a 22000 dollari”
 2. “Elenco ordinato dei nomi dei dipartimenti che finiscono in ‘S’ e che fanno capo al dipartimento ‘D01’”
 3. “Nomi di tutti le persone appartenenti allo staff che guadagnano oltre 20000 dollari ma che non sono manager”
 4. “Gli impiegati con 16 anni di formazione relativi ai dipartimenti ‘ADMINISTRATION SYSTEMS’ e ‘SUPPORT SERVICES’”



Query database “SAMPLE” (2)

5. “Nomi dei dipartimenti che non hanno un manager di riferimento”
6. “Identificativo e nome delle persone appartenenti allo staff che non hanno commissione nulla e che hanno un’anzianità di servizio compresa tra i 3 e i 5 anni”
7. “Nome, lavoro e compenso totale delle persone appartenenti allo staff il cui salario, maggiorato della commissione, è minore o uguale di 17000 dollari (in ordine crescente di compenso totale)”
8. “Numero e nome dei progetti che sono terminati in data 1982-09-15 o in data 1982-12-01 in ordine alfabetico ‘A-Z’ per il numero progetto e ‘Z-A’ per il nome (utilizzando operatori insiemistici)”

Query database “SAMPLE” (3)

9. “Regioni di vendita in cui sia l’impiegato ‘LUCCHESSI’ che l’impiegato ‘LEE’ hanno venduto in data 1996-03-29 (utilizzando operatori insiemistici)”
10. “Cognome (ordine alfabetico ‘A-Z’ su entrambi) degli impiegati che hanno effettuato vendite in Quebec in data 1996-04-01”
11. “Dati relativi ai progetti i cui responsabili sono di sesso maschile, hanno una formazione scolastica maggiore o uguale a 18 anni e il cui dipartimento di afferenza è quello di PLANNING”
12. “Nome della regione e totale vendite per ogni regione, ordinando il risultato per il totale vendite (ordine decrescente) ”
13. “Nome dei progetti e numero di impiegati distinti che lavorano ad ogni progetto in ordine decrescente di numero impiegati (selezionare solo i primi 5 record)”

DEPARTMENT

DEPTNO

DEPTNAME

MGRNO

ADMRDEPT

LOCATION

EMPLOYEE

EMPNO

FIRSTNME

MIDINIT

LASTNAME

WORKDEPT

PHONENO

HIREDATE

EDLEVEL

SEX

BIRTHDATE

SALARY

BONUS

COMM

EMP_ACT

EMPNO

PROJNO

ACTNO

EMPTIME

EMSTDATE

EMENDATE

SALES

SALES DATE

SALES PERSON

REGION

SALES

PROJECT

PROJNO

PROJNAME

DEPTNO

RESPEMP

PRSTAFF

PRSDATE

PRENDATE

MAJPROJ

STAFF

ID

NAME

DEPT

JOB

YEARS

SALARY

COMM

Gestione di un'azienda meccanica...

- Con riferimento alle tabelle relative alla gestione dell'azienda meccanica dell'esercitazione precedente, ovvero **clienti**, **pezzi** e **pezzi_spediti**:

CLIENTI (CODICE_FISC, RAG_SOC, TELEFONO)

PEZZI (CODICE_PEEZO, DIM1, DIM2, DIM3, DESCRIZ)

**PEZZI_SPEDITI (CODICE_SPED, CODICE_PEEZO, CODICE_FISC,
QUANTITA)**

...gli **attributi in blu** rappresentano la **chiave primaria** di ciascuna tabella

Applicazione Java **AziendaMeccanica**

- Scrivere l'applicazione Java **“AziendaMeccanica”** che, caricando l'opportuno driver JDBC, si connette al database **STUDENTI** e, interrogando le tabelle **CLIENTI**, **PEZZI** e **PEZZI_SPEDITI**, risolve le seguenti interrogazioni:
 - “Dati relativi alle spedizioni verso clienti la cui ragione sociale contiene la parola ‘Vini’ e il cui numero di telefono sia diverso da null”
 - “Codice spedizione e quantità relativi alle spedizioni i cui pezzi hanno larghezza e altezza minori di X cm, in ordine decrescente di quantità e crescente di codice spedizione”
 - Unione dei codici spedizione relativi ai pezzi con descrizione nulla e relativi ai clienti con telefono nullo (con e senza duplicati)