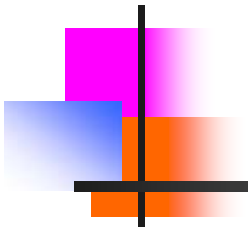


Esercitazione 03: JDBC e Prepared Statements

Sistemi Informativi L-A

Home Page del corso:
http://www-db.deis.unibo.it/courses/SIL-A/

Versione elettronica: Esercitazione03.pdf

Applicazione Java **`Sample1`** (1)

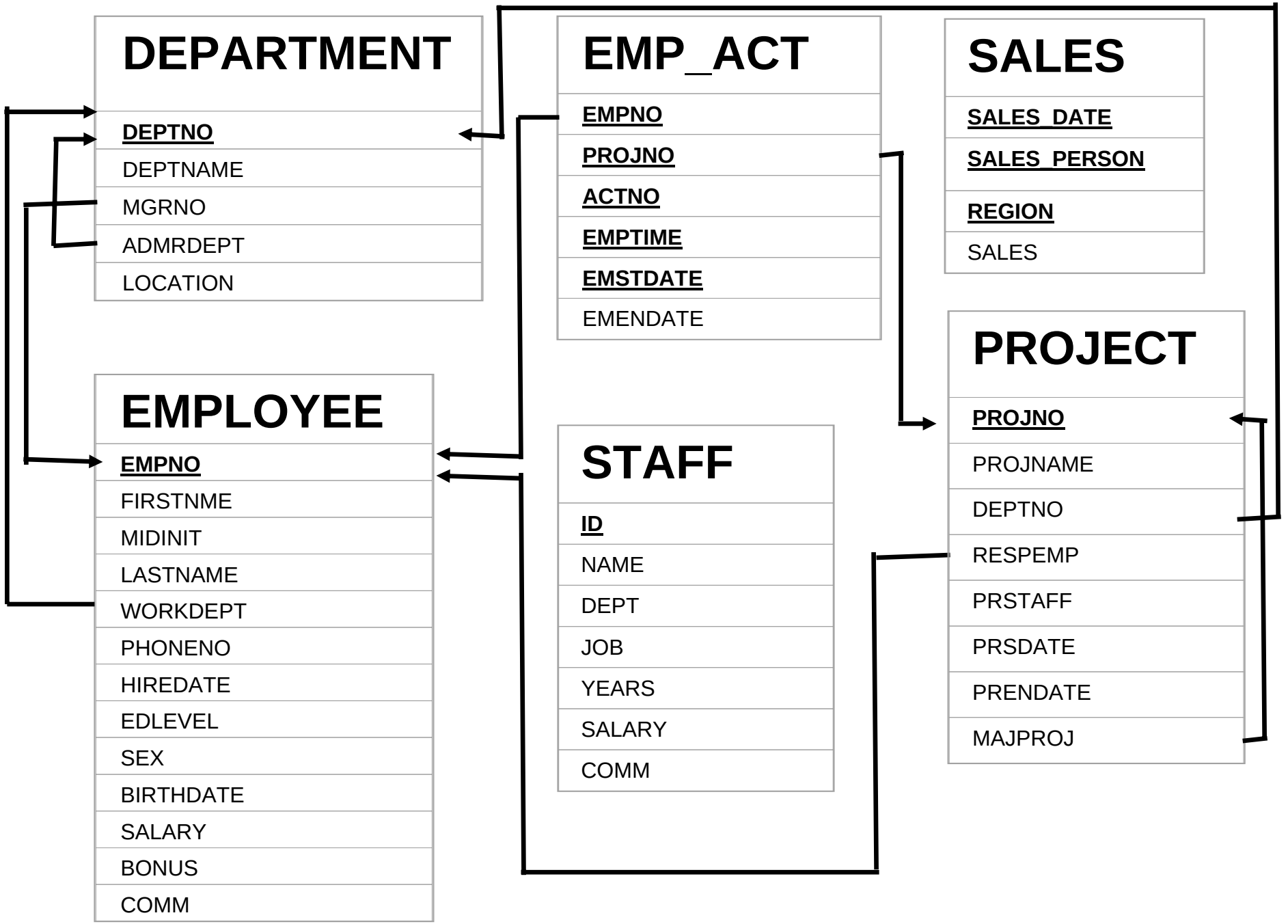
- Scrivere l'applicazione Java **`"Sample1"`** che, caricando l'opportuno driver JDBC, si connette al database **`SAMPLE`** e:
  1. Richiede all'utente, da standard input, username e password (a tale fine fare riferimento al metodo **`readEntry`** di cui è riportata la specifica nelle slide relative a JDBC)
  2. Definisce un vettore di interi (**`anno[]`**) adatto a contenere le date comprese tra il **`1973`** e **`1980`**
  3. Definisce un oggetto di tipo **`PreparedStatement`** a cui viene passata la stringa rappresentante la soluzione dell'interrogazione "Nome, cognome e data di assunzione degli impiegati che sono stati assunti negli anni **`anno[0]`** o **`anno[1]`**o … **`anno[n]`**, ordinati per ordine crescente di data (osservazione: i dati sconosciuti in fase di formulazione dell' interrogazione SQL vengono rappresentati con **`?`**)"
  4. Inizializza il vettore con le date 1973-1980 e attribuisce i valori degli elementi dell'array (**`anno[i]`**) ai valori dell'attributo data assunzione

Applicazione Java **`Sample1`** (2)

5. Esegue lo statement
6. Produce la stampa di nome, cognome e data di assunzione sul file di testo **`Censimento73-80.txt`**

…varianti:

1. Produrre gli stessi dati, oltre che per l'intervallo 73-80, per i sottointervalli 73-75, 76-78 e 79-80
2. Calcolare, da statement SQL, anche il numero rappresentante i giorni di anzianità di ciascun impiegato (facendo riferimento alla data corrente: **`current date`**) producendone la stampa sullo stesso file
3. Come al punto 2. ma calcolando l'anzianità in anni


DEPARTMENT
DEPTNO
DEPTNAME
MGRNO
ADMRDEPT
LOCATION
EMP_ACT
EMPNO
PROJNO
ACTNO
EMPTIME
EMSTDATE
EMENDATE
SALES
SALES_DATE
SALES_PERSON
REGION
SALES
PROJECT
PROJNO
PROJNAME
DEPTNO
RESPEMP
PRSTAFF
PRSDATE
PRENDATE
MAJPROJ
EMPLOYEE
EMPNO
FIRSTNME
MIDINIT
LASTNAME
WORKDEPT
PHONENO
HIREDATE
EDLEVEL
SEX
BIRTHDATE
SALARY
BONUS
COMM
STAFF
ID
NAME
DEPT
JOB
YEARS
SALARY
COMM

Applicazione Java **Progetti** (1)

- Scrivere l'applicazione Java **"Progetti"** che, caricando l'opportuno driver JDBC, si connette al database **STUDENTI** e:
  1. Crea (e popola) gli schemi riportati di seguito

```
PROGETTI(
   COD_PROG CHAR(3) NOT NULL PRIMARY KEY,    -- codice progetto
   IMP_ASSEGNATO DECIMAL(8,2) NOT NULL CHECK (IMP_ASSEGNATO > 0))
                                        --importo assegnato al progetto
SPESE(
   COD_SPESA CHAR(4) NOT NULL PRIMARY KEY,    -- id di una spesa
   COD_PROG  CHAR(3) NOT NULL,  -- progetto a cui la spesa è imputata
   COD_RIC CHAR(3) NOT NULL,  -- ricercatore che ha eseguito la spesa
   IMPORTO DECIMAL(8,2) NOT NULL CHECK (IMPORTO > 0),
                                        -- importo della spesa
   FOREIGN KEY COD_PROG REFERENCES PROGETTI)
```

Applicazione Java **Progetti** (2)

2. Genera la tabella **CONSUNTIVO** con i seguenti campi (e vincoli opportuni):

```
COD_PROG          CHAR(3),
IMP_ASSEGNATO     DECIMAL(8,2),
RESIDUO           DECIMAL(8,2),
COEFF_SPESA       CHAR(1)
```

dove:

- **RESIDUO**: corrisponde all'importo assegnato a un dato progetto e non ancora speso (ad esempio, se un progetto ha assegnati 6000 Euro e ha due spese, rispettivamente di 1000 e di 800 Euro, il residuo è pari a 4200 Euro)
- **COEFF_SPESA**: è un valore appartenenente all'insieme ('A','B','C','D') che viene calcolato dal metodo Java **fasciaSpesa**. Tale metodo viene invocato fornendo in input il valore (di tipo double) del rapporto tra il residuo e l'importo assegnato (ad esempio, nel caso illustrato, viene fornito in input il valore 4200/6000)

3. Inserisce in **CONSUNTIVO**, sulla base dei dati presenti in **PROGETTI** e **SPESE**, una tupla per ogni progetto, inclusi quelli che non hanno avuto alcuna spesa
4. Stampa sul file consuntivo.txt il contenuto di **CONSUNTIVO**

Query database **Videonoleggio**

Facendo riferimento alle tabelle **CLIENTI**, **DVD** e **NOLEGGI** relative all'esercitazione scorsa, risolvere le seguenti interrogazioni SQL mediante CLP e file di input e di output:

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"