

**Secondo compito**  
**Laboratorio Basi di Dati II**  
**Modelli e DBMS di nuova generazione**  
**21 Dicembre 2004**

Questo compito contiene esercizi che devono essere fatti da chi ha seguito: solo il corso “*Laboratorio di Basi di dati II*” (LABO); solo il corso “*modelli dei dati e DBMS di nuova generazione*” (MODELLI); o entrambi i corsi.

Per chi ha seguito solo il corso di modelli, svolgere gli esercizi 1, 2, (5|6)

Per chi ha seguito solo il corso di Labo Basi di dati II, svolgere gli esercizi 3, 4, (5|6)

Per chi ha seguito entrambi i corsi, svolgere gli esercizi 1, 3, (5|6)

Se uno svolge più esercizi di quelli richiesti per la propria situazione, verrà considerata la migliore combinazione di esercizi nel caso di esercizi alternativi gli uni agli altri (a|b) e verranno assegnati punti extra negli altri casi.

Specificare nel foglio che consegnate in quale situazione vi trovate.

**Esercizio 1 (MODELLI)**

Si supponga di avere la base di dati operativa di un gruppo di emittenti radio. Di ogni emittente si conosce la sede dalla quale trasmette (che si trova in una certa città, regione, stato), il nome del direttore e il numero (medio) di persone che la seguono. Di ogni pubblicità che la radio trasmette si conosce il nome, il testo del messaggio, il nome dell'azienda che la sponsorizza la quale appartiene ad un certo settore di mercato (azienda al dettaglio, all'ingrosso, internazionale), il logo dell'azienda, e la sottotipologia e la tipologia della pubblicità. Esempi di tipologia sono: calzature, arredamento, alimentare, telecomunicazioni. Esempi di sottotipologia per la tipologia “telecomunicazioni” sono: telefonia fissa, telefonia mobile, satelliti.

Il gruppo di emittenti radio vuole confrontare le politiche pubblicitarie delle varie emittenti. A tale proposito, monitorizza giornalmente ogni spot pubblicitario in relazione al numero di volte in cui lo spot appare in ciascuna emittente, alla durata totale (cioè al tempo complessivo occupato nel contesto di uno stesso giorno dallo stesso spot nel palinsesto dell'emittente) e la percentuale di pubblico dell'emittente che segue lo spot (questa informazione viene ricavata attraverso un'indagine statistica).

Si richiede di:

1. Definire una schema ER per il dominio sopra descritto.
2. Identificare il fatto che l'azienda è interessata ad analizzare.
3. Stabilire le dimensioni di analisi e le misure che possono essere calcolate.
4. Specificare se le misure individuate sono additive, semi-additive, non additive. Per le misure semi-additive e non additive specificare se è possibile (e come) renderle additive.
5. Specificare le gerarchie che possono essere rappresentate per le dimensioni individuate.
6. Disegnare lo schema concettuale e logico del fatto.
7. Presentare l'oggetto DIMENSION di Oracle per la dimensione che presenta il maggior numero di aggregazioni
8. (Opzionale) Si supponga che occorra effettuare analisi sugli spot mandati in onda da emittenti della Liguria in relazione alla loro tipologia di appartenenza, con granularità settimanale. Si definisca una vista materializzata opportuna tenendo in considerazione che la vista venga costruita immediatamente, venga aggiornata al commit delle operazioni, venga utilizzata dall'aggregate navigator. Si preveda di poter aggregare i dati rispetto a tutte le possibili combinazioni delle dimensioni individuate.

**Esercizio 2 (MODELLI)**

Si considerino le seguenti transazioni:

1. broccoli, carciofi, fagioli, patate
2. broccoli, carciofi
3. broccoli, fagioli, patate
4. fagioli, patate
5. fagioli, patate
6. broccoli, fagioli
7. carciofi, fagioli, patate

Determinare le regole di associazioni per tale insieme di transazioni. Supporto minimo 50% e confidenza 75%.

### Esercizio 3 (LABO)

Si considerino le seguenti due relazioni:

VillaggiTuristici(codV, nome,località, #posti,responsabile)  
Ospiti(idOsp,nome,quota,inizioSoggiorno,fineSoggiorno,villaggio)

In cui InizioSoggiorno e FineSoggiorno delimitano il periodo di tempo in cui un ospite sta nel Villaggio, e Ospiti.villaggio e' chiave esterna su VillaggiTuristici.

Si supponga che l'interrogazione che viene eseguita principalmente su questo schema e':

```
SELECT codV, COUNT(*)
FROM VillaggiTuristici, Ospiti
WHERE codV=villaggio AND localita = '5 Terre' AND
      inizioSoggiorno BETWEEN '24/12/2004' AND '30/12/2004'
GROUP BY codV
```

Rispondere alle seguenti domande (giustificando le risposte):

1. Ideare una struttura di memorizzazione primaria delle due relazioni e strutture ausiliarie di accesso che permettano di ottimizzare l'esecuzione della precedente interrogazione.
2. Fare vedere come i dati vengono organizzati nel caso in cui le due tabelle, dal punto di vista logico, contengono i seguenti dati

CodV	Nome	localita	#posti	responsabile
A001	Vernazza Tour	5 Terre	100	M. Mesiti
A002	Eolie Club	Eolie	150	A. Maddalena
A003	Rio Maggiore Club Med	5 Terre	80	G. Guerrini

IdOsp	Nome	quota	InizioSoggiorno	FineSoggiorno	Villaggio
O01	S. Accordino	1.000	24/12/2004	06/01/2005	A001
O02	A. Boca	800	19/12/2004	26/12/2004	A002
O03	A. Boveri	1.100	13/12/2004	20/12/2004	A003
O04	A. Cislaghi	900	28/12/2004	10/01/2005	A001
O05	R. d'Amato	700	24/12/2004	30/12/2004	A001
O06	V. D'Errico	1.050	01/12/2004	15/12/2004	A003

3. Presentare un esempio di interrogazione la cui esecuzione diventa particolarmente costosa sulla base della organizzazione fisica presentata al punto 1.
4. Come potrebbe essere modificato lo schema presentato al punto 1 nel caso in cui si considerasse anche l'interrogazione

```
SELECT codV, nome, quota
FROM VillaggiTuristici, Ospiti
WHERE codV=villaggio AND localita = '5 Terre' AND
      inizioSoggiorno BETWEEN '24/12/2004' AND '30/12/2004' AND
      quota = 1000
```

### Esercizio 4 (LABO)

Mostrare la struttura hash estensibile, con capacita' di ogni pagina 2, supponendo di inserire nell'ordine i record le cui pseudochiavi hanno i seguenti valori:

h(r1)=1010000 h(r2)=0111000 h(r3)=1011001 h(r4)=0110000 h(r5)= 00100100 h(r6)=11101000 h(r7)=010100

<pre> &lt;a&gt;   &lt;b&gt;     &lt;e&gt;ciao&lt;/e&gt;     &lt;f&gt;20&lt;/f&gt;   &lt;/b&gt;   &lt;c&gt;     &lt;h&gt;20&lt;/h&gt;     &lt;g&gt;ciao&lt;/g&gt;     &lt;i&gt;38&lt;/i&gt;     &lt;i&gt;98&lt;/i&gt;     &lt;f&gt;20&lt;/f&gt;   &lt;/c&gt; &lt;/a&gt; </pre>	<pre> &lt;a&gt;   &lt;b&gt;     &lt;e&gt;ciao&lt;/e&gt;     &lt;z&gt;20&lt;/z&gt;     &lt;z&gt;30&lt;/z&gt;   &lt;/b&gt;   &lt;b&gt;     &lt;e&gt;ciao&lt;/e&gt;   &lt;/b&gt;   &lt;d&gt;     &lt;z&gt;20&lt;/z&gt;   &lt;/d&gt;   &lt;c alpha="4"&gt;     &lt;m&gt;20&lt;/m&gt;     &lt;g&gt;ciao&lt;/g&gt;   &lt;/c&gt; &lt;/a&gt; </pre>
---	---

Figura 1 Due documenti XML

### Esercizio 5 (MODELLI/LABO)

Si considerino i due documenti XML in Figura 1.

1. Scrivere un DTD che generalizzi la struttura dei due documenti. Il DTD deve descrivere i due documenti in modo conciso. I due documenti devono essere validi per tale DTD.
2. Se invece del DTD, si dovesse generare un XML schema, quali elementi sarebbe meglio dichiarare globali? Giustificare la risposta.
3. Scrivere la dichiarazione dell'elemento b in XML schema.

```

<?xml version="1.0" encoding="ISO-8859-1"?>
<catalog>
  <cd>
    <title>Empire Burlesque</title>
    <artist>Bob Dylan</artist>
    <country>USA</country>
    <company>Columbia</company>
    <price>10.90</price>
    <year>1985</year>
  </cd>
  <cd>
    <title>Hide your heart</title>
    <artist>Bonnie Tyler</artist>
    <country>UK</country>
    <company>CBS Records</company>
    <price>9.90</price>
    <year>1988</year>
  </cd>
  <cd>
    <title>Greatest Hits</title>
    <artist>Dolly Parton</artist>
    <country>USA</country>
    <company>RCA</company>
    <price>9.90</price>
    <year>1982</year>
  </cd>
</catalog>

```

Figura 2 Un documento XML rappresentate CD musicali

### Esercizio 6 (MODELLI/LABO)

Si consideri il documento XML in Figura 2.

Sviluppare un foglio di stile XSL che permetta di generare una pagina HTML in cui viene riportata la tabella (titolo,company,price) dei CD di Bob Dylan del 1985 che costano piu' di 10 euro.

## Comandi di Oracle per la gestione del DW

```
CREATE DIMENSION <nome>
{LEVEL <nome_l> IS <nome tabella>.<attr>}
HIERARCHY <nome gerarchia> (
    {<nomelivello> CHILD OF}
    ...)
{ATTRIBUTE <nome livello> DETERMINES <nome<tabella>.<attr>}
```

```
CREATE MATERIALIZED VIEW nome
BUILD [IMMEDIATE|DEFERRED]
REFRESH [FAST|COMPLETE|FORCE] [ON COMMIT|ON DEMAND|START WITH <date> NEXT <date expr>]
[ENABLE QUERY REWRITE]
AS <subquery >
```

```
DROP MATERIALIZED VIEW nome
```

```
ALTER MATERIALIZED VIEW ...
```

```
SELECT ...FROM... WHERE... GROUP BY CUBE (elenco colonne)
```

```
SELECT .... FROM... WHERE... GROUP BY ROLLUP (elenco colonne)
```

```
SELECT A1,...,An
FROM
    (SELECT B1,...,Bm, RANK() OVER(ORDER BY Ai ASC ORDER BY Aj DESC) AS rank
FROM ...
WHERE ...
GROUP BY (A1,...,An)
WHERE rank <= N;
```

## Comandi di XSL

```
<xsl:template
  match = pattern
  name = qname
  priority = number>
<!-- Content: (xsl:param*, template) -->
</xsl:template>

<xsl:apply-templates select = node-set-expression >
<!-- Content:(xsl:sort | xsl:with-param)* -->
</xsl:apply-templates>

<xsl:value-of select= node-set-expression />

<xsl:for-each
  select = node-set-expression>
  <!-- Content: (xsl:sort*, template) -->
</xsl:for-each>

<xsl:choose>
<xsl:when test= node-set-expression>
... some code ...</xsl:when>
<xsl:otherwise>... some code .... </xsl:otherwise>
</xsl:choose>

<xsl:if test= node-set-expression> some output ... </xsl:if>

<xsl:sort select= node-set-expression />

node-set-expression e' un'espressione Xpath
```

## Tabella in HTML

```
<html>
<head>
<title>Title of page</title>
</head>
<body>
This is my first homepage.
<b>This text is bold</b>

<table border="1">
<tr>
<td>row 1, cell 1</td>
<td>row 1, cell 2</td>
</tr>
<tr>
<td>row 2, cell 1</td>
<td>row 2, cell 2</td>
</tr>
</table>

</body>
</html>
```

## Comandi di XML schema

```
<xsd:element name="name" type="type" minOccurs="int" maxOccurs="int"/>
```

*basicType*: xsd:integer, xsd:string, xsd:date

```
<xsd:element name="name" minOccurs="int" maxOccurs="int">
  <xsd:complexType>
    <xsd:[sequence|choise]>
    ....
  </xsd:[sequence|choise]>
</xsd:complexType>
</xsd:element>
```

DEFAULT: minOccurs 1, maxOccurs 1

```
<xsd:simpleType name="name">
  <xsd:restriction base="xsd:source">
    [ <xsd:facet value="value"/>| <xsd:enumerate value="value"/>]
  </xsd:restriction>
</xsd:simpleType>
```

Element	Explanation
attribute	Defines an attribute
choice	Allows only one of the elements contained in the <choice> declaration to be present within the containing element
complexType	Defines a complex type element
element	Defines an element
extension	Extends an existing simpleType or complexType element
restriction	Defines restrictions on a simpleType, simpleContent, or a complexContent
schema	Defines the root element of a schema
sequence	Specifies that the child elements must appear in a sequence.
simpleContent	Contains extensions or restrictions on a text-only complex type or on a simple type as content and contains no elements
simpleType	Defines a simple type and specifies the constraints and information about the values of attributes or text-only elements

## XSD Restrictions/Facets for Datatypes

Constraint	Description
enumeration	Defines a list of acceptable values
length	Specifies the exact number of characters or list items allowed.
[max min]Exclusive	Specifies the [upper lower] bounds for numeric values
[max min]Inclusive	Specifies the [upper lower] bounds for numeric values
[max min]Length	Specifies the [maximum minumum] number of characters or list items allowed.