

Basi di dati II — Prova parziale — 11 aprile 2016 — Compito A

Tempo a disposizione: un'ora e trenta minuti.

Cognome _____ Nome _____ Matricola _____

Con riferimento al progetto “Carriere studenti,” si considerino gli aspetti che riguardano:

- gli studenti, identificati dal codice fiscale, con data di nascita, indirizzo di residenza e coorte (che supponiamo sia specificata con l'anno di prima iscrizione all'università in assoluto); si noti che, come nel progetto realizzato, non viene qui associato allo studente il corso di studio cui è iscritto (non è detto che sia la scelta migliore, ma assumerla come dato di fatto);
- l'anno accademico di riferimento (una stringa nella forma x-1/x, ad esempio 2014/2015);
- il corso di studi, con un codice, un titolo, una durata legale, un'utenza sostenibile (il numero massimo di studenti ammissibili), un tipo (“corso di laurea triennale”, “corso di laurea magistrale”);
- i vari corsi, con un codice, un nome, un numero di CFU e un settore scientifico disciplinare (ad esempio ING-INF/05).

Si supponga di disporre della seguente sorgente di dati che riporta gli eventi relativi al superamento di esami.

EVENTIESAMI

Studente (stringa)	Codicecorso (stringa)	Corsodistudi (stringa)	Voto (numerico)	Data (tipo data)	AnnoAccademico (stringa)
{1234-5678-1203}	503040	20802178	23	2016-01-31	2015/2016
{1234-5678-1203}	507045	20802178	29	2016-02-05	2015/2016
{1234-5678-6273}	503042	20202181	31	2016-01-31	2014/2015
{1234-5678-6273}	507045	20202181	28	2015-09-05	2014/2015
...

Domanda 1 Si mostri lo schema dimensionale di un data mart da utilizzare per effettuare analisi sul numero di **CFU conseguiti** e sui voti riportati da ciascuno studente per data solare, anno accademico, corso e corso di studi. Mostrare lo schema relazionale (in qualunque modo compatto e comprensibile), indicando gli attributi (e le chiavi) di ciascuna delle relazioni. Si tratta sostanzialmente di quanto descritto sopra, versione semplificata di quello analogo utilizzato nel progetto.

Domanda 2 Si vuole realizzare un flusso di ETL in grado di alimentare la tabella dei fatti dello schema dimensionale della domanda 1, a partire dalla sorgente EVENTIESAMI. Si supponga che le dimensioni progettate siano state opportunamente alimentate in precedenza e che la sorgente dati sia una base di dati relazionale. Si specifichi quindi il flusso ETL attraverso una interrogazione SQL del tipo:

```
INSERT INTO <fact table> (<kdim_1>, ..., <kdim_n>, <misura_1>, ..., <misura_n>)  
SELECT ...
```

Considerare disponibili funzioni di utilità quali le seguenti:

`to_date(<stringa>)`: che trasforma una stringa in formato YYYY-MM-DD in una data;
`sysdate()`: che restituisce la data corrente;
`to_number(<stringa>)`: che trasforma in numero una stringa composta di sole cifre;
`extract_year(<data>)`: che estrae l'anno da una data, restituendo una stringa in formato YYYY;
...

Domanda 3 Mostrare un'interrogazione SQL che esegua la seguente analisi sul data mart della domanda 1:

1. per l'anno accademico 2014/2015, per ogni corso di studi, visualizzare il numero medio di **CFU conseguiti** dagli studenti di ciascuna coorte;
2. per l'anno accademico 2014/2015, per ogni corso e corso di studi, visualizzare il numero medio di **CFU conseguiti** dagli studenti della coorte 2013.

Domanda 4 Si supponga di voler gestire le variazioni nel tempo **dell'indirizzo di residenza degli studenti** e di voler tenere in considerazione, nelle analisi, il valore valido ad una determinata data.

1. Modificare opportunamente lo schema dimensionale progettato nella domanda 1 per tenere conto di questo aspetto.
2. Spiegare (anche in modo informale, ma chiaro) come va modificato allo scopo il flusso di ETL mostrato in risposta alla domanda 2.

Basi di dati II — 11 aprile 2016 — Compito A

Si supponga ora di essere interessati a monitorare l'andamento nel tempo delle misure. In particolare, si vogliono effettuare interrogazioni per conoscere il profitto degli studenti ad una specifica data solare, con la granularità massima del singolo studente, ma anche raggruppando per anno accademico o corso di studi. In questi contesti, risulta particolarmente conveniente utilizzare una tabella dei fatti (indicata da alcuni autori come *periodic snapshot*), con la seguente struttura:

- FATTIESAMISNAPSHOT(KStudente, KAnnoAccademico, KCorsoDiStudi, KTempo, MediaVoti, NumeroEsami)

in cui è presente una dimensione temporale specifica che memorizza la data solare di osservazione. Una ennupla rappresenta quindi la media dello studente ad una certa data (e così ci sono più ennuple, con date diverse e tutti gli altri riferimenti a dimensioni uguali): i fatti vengono quindi inseriti con periodicità fissata (ad esempio giornalmente, anche se in questo caso è eccessivo), potendo così monitorare l'andamento delle misure. Le dimensioni sono conformi a quelle dello schema dimensionale utilizzato nelle precedenti domande (in effetti, sono sottoinsieme di esse), a parte la dimensione temporale che, pur essendo uguale, ha un ruolo diverso. Si noti che le misure in questo schema differiscono da quelle nello schema alla pagina precedente (numero esami superati vs CFU conseguiti).

Domanda 5 Si progetti un flusso di ETL in grado di alimentare la tabella FATTIESAMISNAPSHOT a partire dalla relazione EVENTIESAMI (e dalle dimensioni utilizzate nelle domande precedenti).

Domanda 6 Si supponga di voler attribuire una caratterizzazione agli studenti, per ciascun anno accademico, sulla base della media dei voti riportati nell'anno accademico stesso (per l'anno corrente, fino alla data di osservazione), attribuendo una fascia: SUFFICIENTE (media inferiore a 23), BUONO (media da 23 a 28 inclusi), OTTIMO (media superiore a 28).

1. Modificare opportunamente lo schema dimensionale con la tabella dei fatti FATTIESAMISNAPSHOT (cioè quello discusso all'inizio di questa pagina) per poter tener conto nelle analisi della fascia attribuita a ciascuno studente nel tempo.
2. Mostrare inoltre una query SQL che modifichi il contenuto della tabella dei fatti, aggiungendo per ogni riga la relativa fascia.

Cenni sulle soluzioni (compito A, con qualche indicazione sulle varianti)

Domanda 1

Dimensioni

- STUDENTE(KStud, CF, Coorte, DataNascita, AnnoNascita, IndirizzoResidenza, ComuneResidenza, ...)
- ANNOACCADEMICO(KAA, AnnoAccademico, AnnoAccademicoX, ...)
- CORSODISTUDI(KCdS, CodiceCdS, TitoloCdS, ...)
- CORSO(KCorso, CodiceCorso, NomeCorso, CFUCorso, SSDCorso, ...)
- TEMPO(KTempo, Anno, Mese, Giorno, ...)

Tabella dei fatti (la grana è il singolo esame)

- (per il compito A)
FATTIESAMI(KStud, KAA, KCdS, KCorso, KTempo, Voto, NumeroCFU)
- (per il compito B)
FATTIESAMI(KStud, KAA, KCdS, KCorso, KTempo, Voto, NumeroEsami); NumeroEsami è sempre pari a 1

Cenni sulle soluzioni (compito A, con qualche indicazione sulle varianti)

Domanda 2

```
insert into FattiEsami(kstud, kaa, kcds, kcorso, ktempo, voto, numerocfu)
select kstud, kaa, kcds, kcorso, ktempo, voto, cfucorso
from EventiEsami ee join Studente s on (ee.studente = s.cf)
  join AnnoAccademico a
      on (extract( year from to_date(ee.data, 'YYYY-MM-DD')) = a.AnnoAccademicoX)
  join CorsoDiStudi cds on (ee.corsodistudi = cds.codicecgs)
  join Corso c on (ee.CodiceCorso = c.CodiceCorso)
  join Tempo t on ((t.anno = extract(year from to_date(ee.data, 'YYYY-MM-DD')) and
    (t.mese = extract(month from to_date(ee.data, 'YYYY-MM-DD')) and
    (t.giorno = extract(day from to_date(ee.data, 'YYYY-MM-DD'))))
```

Cenni sulle soluzioni (compito A, con qualche indicazione sulle varianti)

Domanda 3

1.

```
create or replace view cfustudente as (  
select f.kstud as kstud, kaa, kcds, sum(numerocfu) as totalecfu  
from fattiesami f join studente s on (f.kstud = s.kstud)  
group by f.kstud, Kaa, kcds  
);
```

```
select titolocds, coorte, avg(totalecfu) as mediacfu  
from cfustudente v  
      join annoaccademico a on (v.kaa = a.kaa)  
      join studente s on (v.kstud = s.kstud)  
      join corsodistudi cds on (v.kcds = cds.kcds)  
where a.annoaccademico = '2014-2015'  
group by titolocds, coorte
```

2.

```
create or replace view cfustudenteBis as (  
select f.kstud as kstud, kaa, kcorso, kcds, sum(numerocfu) as totalecfu  
from fattiesami f join studente s on (f.kstud = s.kstud)  
group by f.kstud, Kaa, kcorso, kcds  
);
```

```
select titolocds, nomecorso, avg(totalecfu) as mediacfu  
from cfustudenteBis v  
      join annoaccademico a on (v.kaa = a.kaa)  
      join studente s on (v.kstud = s.kstud)  
      join corsodistudi cds on (v.kcds = cds.kcds)  
      join corso c on (v.kcorso = c.kcorso)  
where a.annoaccademico = '2014-2015'  
      and s.coorte = 2013  
group by titolocds, nomecorso
```

Cenni sulle soluzioni (compito A, con qualche indicazione sulle varianti)

Domanda 4

- La dimensione studente evolve lentamente ed è opportuno gestirla con la cosiddetta tecnica di tipo 2, cioè con una ennumera per ogni “versione” dello studente con diverso indirizzo. Possono essere utili date di inizio e fine validità della versione.
- Il flusso di ETL deve verificare la data in cui è stato sostenuto l’esame, per scegliere la versione corretta dello studente; ad esempio qualcosa del genere:

```
insert into FattiEsami(kstud, kaa, kcds, kcorso, ktempo, voto, numerocfu)
select kstud, kaa, kcds, kcorso, ktempo, voto, cfucorso
from EventiEsami ee join Studente s on (ee.studente = s.cf)
    join AnnoAccademico a
        on extract( year from to_date(ee.data, 'YYYY-MM-DD')) = a.AnnoAccademicoX
    join CorsoDiStudi cds on (ee.corsodistudi = cds.codicecads)
    join Corso c on (ee.CodiceCorso = c.CodiceCorso)
    join Tempo t on ((t.anno = extract(year from to_date(ee.data, 'YYYY-MM-DD')) and
        (t.mese = extract(month from to_date(ee.data, 'YYYY-MM-DD')) and
        (t.giorno = extract(day from to_date(ee.data, 'YYYY-MM-DD'))))
where (to_date(ee.data, 'YYYY-MM-DD') >= datainiziovalidita)
    and (to_date(ee.data, 'YYYY-MM-DD') <= datafinevalidita or datafinevalidita is null)
```

Cenni sulle soluzioni (compito A, con qualche indicazione sulle varianti)

Domanda 5

```
insert into FattiEsamiSnapshot(kstud, kaa, kcds, ktempo, mediavoti, numeroesami)
select kstud, kaa, kcds, ktempo, avg(voto), count(*)
from EventiEsami ee join Studente s on (ee.studente = s.cf)
  join AnnoAccademico a
      on extract( year from to_date(ee.data, 'YYYY-MM-DD')) = a.AnnoAccademicoX
  join CorsoDiStudi cds on (ee.corsodistudi = cds.codicecds)
  join Corso c on (ee.CodiceCorso = c.CodiceCorso)
  join Tempo t on ((t.anno = extract(year from current_date)) and
                  (t.mese = extract(month from current_date)) and
                  (t.giorno = extract(day from current_date)))
group by kstud, kaa, kcds, ktempo
```

Cenni sulle soluzioni (compito A, con qualche indicazione sulle varianti)

Domanda 6

1. si può aggiungere una dimensione (secondaria) FASCIADIMERITO(KFascia,Descrizione,MediaDa,MediaA) con la relativa chiave nella tabella dei fatti
2. Una volta aggiunta la colonna alla tabella dei fatti

```
update fattiesamisnapshot s
set kfascia = (
select kfascia
from fasciadimerito f
where s.mediovoti >= f.mediaDa and s.mediovoti <= f.mediaA
);
```