

Homework

Business Intelligence - A.A 2020/2021

Ivan Ruben Mera Franco - matr. 783086

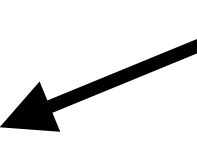
Lorenzo Pirola - matr. 816418

1. Who's playing?

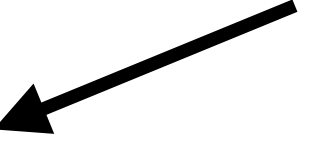
Queries

Estrazione anno dalla data

```
select Cds as corso_di_laurea,  
       AD as appello,  
       year(bos_denormalizzato.dtappello) as anno,  
       count(*) as studenti_iscritti  
from bos_denormalizzato  
group by Cds, AD, year(bos_denormalizzato.dtappello)  
order by Cds, AD, year(bos_denormalizzato.dtappello) asc;
```



```
select c.cds          as corso_di_laurea,  
       a2.ad          as appello,  
       year(a.dtappello) as anno,  
       count(i.studente) as studenti_iscritti  
from appelli a  
       join ad a2 on a.adcod = a2.adcod  
       join cds c on a.cdscod = c.cdscod  
       join iscrizioni i on a.appcod = i.appcod  
group by a2.ad, c.cds, a2.ad, year(a.dtappello)  
order by c.cds, a2.ad, year(a.dtappello) asc;
```



Join

1. Who's playing?

Tempo di esecuzione

DB denormalizzato	DB normalizzato
396 ms	477 ms

Risultati

Corso di studi	Appello	Anno	Studenti
[581] GIURISPRUDENZA	[A5810003] SCIENZA DELLE FINANZE	2017	493
[581] GIURISPRUDENZA	[A5810006] FILOSOFIA DEL DIRITTO	2017	304
[581] GIURISPRUDENZA	[A5810007] DIRITTO INTERNAZIONALE	2017	342
[581] GIURISPRUDENZA	[A5810010] DIRITTO AMMINISTRATIVO	2016	3
[581] GIURISPRUDENZA	[A5810010] DIRITTO AMMINISTRATIVO	2017	288

Visualizzazione



2. I'll never graduate

Queries

```
select tab3.CdS,
       tab3.AD,
       tab3.appelli,
       tab3.tasso_di_superamento as tasso_di_superamento,
       tab3.tasso_di_superamento_std as tasso_di_superamento_std
from (
  select tab2.CdS,
         tab2.AD,
         tab2.appelli,
         tab2.tasso_di_superamento,
         tab2.tasso_di_superamento_std,
         @rn := IF(@prev = tab2.CdS, @rn + 1, 1) AS rn,
         @prev := tab2.CdS
  from (
    select tab1.CdS,
           tab1.AD,
           count(*) as appelli,
           avg(tab1.tasso) as tasso_di_superamento,
           coalesce(
             round(stddev_samp(tab1.tasso),2),
             0) as tasso_di_superamento_std
    from (
      select b.CdS,
             b.AD,
             round((sum(b.Superamento)/count(*))*100, 2) as tasso
      from bos_denormalizzato b
      where b.Superamento = 1 or b.Insufficienza = 1 or b.Ritiro = 1
      group by b.CdS, b.AD, b.dtappello) tab1
    group by tab1.CdS, tab1.AD
    order by tab1.CdS, tasso_di_superamento, tab1.AD) tab2
  join (select @prev := NULL, @rn := 0) vars) tab3
where rn <= 10;
```

Ordinamento per CdS e limit

Tasso di superamento medio per materia

Tasso di superamento medio per appello

Gli assenti non sono considerati

```
select c.cds,
       ad.ad,
       tab3.appelli,
       tab3.tasso_di_superamento as tasso_di_superamento,
       tab3.tasso_di_superamento_std as tasso_di_superamento_std
from (
  select tab2.cdscod,
         tab2.adcod,
         tab2.appelli,
         tab2.tasso_di_superamento,
         tab2.tasso_di_superamento_std,
         @rn := IF(@prev = tab2.cdscod, @rn + 1, 1) AS rn,
         @prev := tab2.cdscod
  from (
    select tab1.cdscod,
           tab1.adcod,
           count(*) as appelli,
           avg(tab1.tasso) as tasso_di_superamento,
           coalesce(
             round(stddev_samp(tab1.tasso),2),
             0) as tasso_di_superamento_std
    from (
      select a.adcod,
             a.cdscod,
             round((sum(i.Superamento)/count(*))*100, 2) as tasso
      from appelli a
      join iscrizioni i on a.appcod = i.appcod
      where i.Superamento = 1 or i.Insufficienza = 1 or i.Ritiro = 1
      group by a.adcod, a.cdscod, a.dtappello) tab1
    group by tab1.cdscod, tab1.adcod
    order by tab1.cdscod, tasso_di_superamento, tab1.adcod) tab2
  join (select @prev := NULL, @rn := 0) vars) tab3
  join ad on tab3.adcod = ad.adcod
  join cds c on tab3.cdscod = c.cdscod
where rn <= 10;
```

Ordinamento per CdS e limit

Tasso di superamento medio per materia

Tasso di superamento medio per appello

Join

2. I'll never graduate

Tempo di esecuzione

DB denormalizzato	DB normalizzato
502 ms	260 ms

Risultati

Cds	AD	Appelli	TdS	TdS_std
[581] GIURISPRUDENZA	[A5810127] DIRITTO COMUNITARIO DELLA...	1	0.000000	0
[581] GIURISPRUDENZA	[A5810064] DIRITTO PROCESSUALE CIVILE C.P.	5	60.000000	54.77
[581] GIURISPRUDENZA	[A5810018] PROCEDURA PENALE	9	62.543333	37.45
[581] GIURISPRUDENZA	[A5810017] DIRITTO PROCESSUALE CIVILE	9	64.014444	25.86
[581] GIURISPRUDENZA	[A5810070] MEDICINA LEGALE	9	66.026667	42.39

Visualizzazione



3. Commitment Queries

```
select tab1.CdS,  
       tab1.date_disponibili,  
       coalesce(tab4.date_con_sovrapposizione,  
                0) as date_con_sovrapposizione,  
       round(coalesce(tab4.date_con_sovrapposizione,  
                      0)/tab1.date_disponibili*100,  
             2) as commitment
```

```
from (
```

```
    select b.CdS, Date disponibili per CdS  
           count(distinct (b.dtappello)) as date_disponibili  
    from bos_denormalizzato b  
    group by b.CdS) tab1
```

```
join (
```

```
    select tab3.CdS, Numero di date con sovrapposizioni  
           count(*) as date_con_sovrapposizione  
    from (
```

```
        select tab2.CdS, Date con sovrapposizioni  
               tab2.dtappello,  
               count(*) as sovrapposizioni  
        from(
```

```
            select b.CdS, Appelli per materia  
                   b.AdCod,  
                   b.dtappello  
            from bos_denormalizzato b  
            group by b.CdS, b.AdCod, b.dtappello) tab2
```

```
        group by tab2.CdS, tab2.dtappello  
        having sovrapposizioni > 1) tab3
```

```
    group by tab3.CdS) tab4 on tab1.CdS = tab4.CdS
```

```
order by commitment desc;
```

Considero solo le date con
almeno una sovrapposizione

```
select c.cds,  
       tab1.date_disponibili,  
       coalesce(tab4.date_con_sovrapposizione,  
                0) as date_con_sovrapposizione,  
       round(coalesce(tab4.date_con_sovrapposizione,  
                      0)/tab1.date_disponibili*100,  
             2) as commitment
```

```
from (
```

```
    select a.cdscod, Date disponibili per CdS  
           count(distinct (a.dtappello)) as date_disponibili  
    from appelli a  
    group by a.cdscod) tab1
```

```
join
```

```
(select tab3.cdscod, Numero di date con sovrapposizioni  
   count(*) as date_con_sovrapposizione  
from (
```

```
    select tab2.cdscod, Date con sovrapposizioni  
           tab2.dtappello,  
           count(*) as sovrapposizioni  
    from(
```

```
        select a.cdscod, Appelli per materia  
               a.adcod,  
               a.dtappello  
        from appelli a  
        group by a.cdscod, a.adcod, a.dtappello) tab2
```

```
    group by tab2.cdscod, tab2.dtappello  
    having sovrapposizioni > 1) tab3
```

```
group by tab3.cdscod) tab4 on tab1.cdscod = tab4.cdscod
```

```
join cds c on tab1.cdscod = c.cdscod
```

```
order by commitment desc;
```

Join

3. Commitment

Tempo di esecuzione

DB denormalizzato	DB normalizzato
585 ms	54 ms

Risultati

CdS	Date Disponibili	Date con sovrapposizioni	Commitment
[581] GIURISPRUDENZA	106	87	82.08
[E1901R] SCIENZE DELL'EDUCAZIONE	93	70	75.27
[E1803M] ECONOMIA DELLE BANCHE, DELLE...	84	62	73.81
[E1802M] ECONOMIA E AMMINISTRAZIONE DELLE...	78	55	70.51
[E2001R] COMUNICAZIONE INTERCULTURALE	91	64	70.33

Visualizzazione



4. Hardest/Easiest Exams

Queries

Tipo classificazione

```
select tab2.CdS,  
       tab2.AD,  
       'Hardest' as type,  
       tab2.sufficienti,  
       tab2.partecipanti,  
       tab2.voto_medio  
from (  
  select tab1.CdS,  
         tab1.AD,  
         tab1.sufficienti,  
         tab1.partecipanti,  
         tab1.voto_medio,  
         @rn := IF(@prev = tab1.CdS, @rn + 1, 1) AS rn,  
         @prev := tab1.CdS  
  from (  
    select b.CdS,  
           b.AD,  
           sum(b.Superamento) as sufficienti,  
           count(*) as partecipanti,  
           coalesce(round(sum(b.Voto)/count(*),2),0) as voto_medio  
    from bos_denormalizzato b  
    where b.Superamento = 1 or b.Insufficienza = 1 or b.Ritiro = 1  
    group by b.CdS, b.AD  
    having sum(b.Voto) is not null  
    order by b.CdS, voto_medio asc, b.AD) tab1  
  join (select @prev := NULL, @rn := 0) vars tab2  
where rn <= 3
```

Ordinamento per CdS e limit

Voto medio per materia

Asc per esami difficili
Desc per esami facili

Gli insufficienti e i ritirati sono considerati
Gli assenti non sono considerati

Tipo classificazione

```
select c.cds,  
       a.ad,  
       'Easiest' as type,  
       tab2.sufficienti,  
       tab2.partecipanti,  
       tab2.voto_medio  
from (  
  select tab1.cdscod,  
         tab1.adcod,  
         tab1.sufficienti,  
         tab1.partecipanti,  
         tab1.voto_medio,  
         @rn := IF(@prev = tab1.cdscod, @rn + 1, 1) AS rn,  
         @prev := tab1.cdscod  
  from (  
    select a.cdscod,  
           a.adcod,  
           sum(i.Superamento) as sufficienti,  
           count(*) as partecipanti,  
           coalesce(round(sum(i.Voto)/count(*),2),0) as voto_medio  
    from appelli a  
    join iscrizioni i on a.appcod = i.appcod  
    where i.Superamento = 1 or i.Insufficienza = 1 or i.Ritiro = 1  
    group by a.cdscod, a.adcod  
    having sum(i.Voto) is not null  
    order by a.cdscod, voto_medio desc, a.adcod) tab1  
  join (select @prev := NULL, @rn := 0) vars tab2  
join cds c on tab2.cdscod = c.cdscod  
join ad a on tab2.adcod = a.adcod  
where rn <= 3
```

Ordinamento per CdS e limit

Voto medio per materia

Join

4. Hardest/Easiest Exams

Tempo di esecuzione

DB denormalizzato	DB normalizzato
801 ms	571 ms

Risultati

CdS	AD	Type	Suff.	Part.	Voto Medio
[581] GIURISPRUDENZA	[A5810064] DIRITTO PROCESSUALE CIVILE C.P.	Hardest	3	5	16.60
[581] GIURISPRUDENZA	[A5810081] ECONOMIA AZIENDALE	Hardest	5	6	17.50
[581] GIURISPRUDENZA	[A5810070] MEDICINA LEGALE	Hardest	32	47	18.30
[581] GIURISPRUDENZA	[A5810139] DIRITTO AMMINISTRATIVO -PARTE ...	Easiest	5	5	30.00
[581] GIURISPRUDENZA	[A5810170] TRANSITIONAL JUSTICE AND STATE...	Easiest	5	5	30.00
[581] GIURISPRUDENZA	[A5810169] EUROPEAN UNION COMMON AND...	Easiest	2	2	30.00

Visualizzazione



5. Fast & Furious Queries

```
select tab1.Studente,  
       tab1.CdS,  
       tab1.StuGen,  
       tab2.data_primo_appello,  
       tab2.data_ultimo_appello,  
       tab2.periodo,  
       tab1.esami,  
       tab1.media,  
       tab1.media/tab2.periodo as rapporto
```

```
from (  
  select b.Studente,          #esami e voto medio per studente  
         b.StuGen,  
         b.CdS,  
         avg(b.Voto) as media,  
         count(*) as esami  
  from bos_denormalizzato b  
 where b.Voto is not null  
 group by b.Studente, b.StuGen, b.CdS  
 having esami > 1) tab1
```

Studenti che hanno svolto
almeno un esame

```
join (  
  select b.Studente,          Periodo di attività per studente  
         b.CdS,  
         min(dtappello) as data_primo_appello,  
         max(dtappello) as data_ultimo_appello,  
         datediff(max(dtappello),min(dtappello)) + 1 as periodo  
  from bos_denormalizzato b  
 group by b.Studente, b.CdS) tab2  
 on tab1.Studente = tab2.Studente and tab1.CdS = tab2.CdS  
 order by tab1.CdS, rapporto desc;
```

Giorno in cui è stato svolto
l'ultimo esame

```
select tab1.Studente,  
       c.cds,  
       s.genere,  
       tab2.data_primo_appello,  
       tab2.data_ultimo_appello,  
       tab2.periodo,  
       tab1.esami,  
       tab1.media,  
       tab1.media/tab2.periodo as rapporto
```

```
from (  
  select i.Studente,          #esami e voto medio per studente  
         a.cdscod,  
         avg(i.Voto) as media,  
         count(*) as esami  
  from appelli a  
       join iscrizioni i on a.appcod = i.appcod  
 where i.Voto is not null  
 group by i.Studente, a.cdscod  
 having esami > 1) tab1
```

```
join (  
  select i.Studente,          Periodo di attività per studente  
         a.cdscod,  
         min(a.dtappello) as data_primo_appello,  
         max(a.dtappello) as data_ultimo_appello,  
         datediff(max(a.dtappello),min(a.dtappello)) + 1 as periodo  
  from appelli a  
       join iscrizioni i on a.appcod = i.appcod  
 group by i.Studente, a.cdscod) tab2
```

```
 on tab1.Studente = tab2.Studente and tab1.cdscod = tab2.cdscod  
 join studenti s on tab1.studente = s.studente  
 join cds c on tab1.cdscod = c.cdscod  
 order by c.cds, rapporto desc;
```

Join

5. Fast & Furious

Tempo di esecuzione

DB denormalizzato	DB normalizzato
863 ms	777 ms

Risultati

CdS	Genere	Primo appello	Ultimo Appello	Giorni	Esami	Voto Medio	Rapporto
[581] GIURISPRUDENZA	M	2017-05-03	2017-05-04	2	2	27.0000	13.50000000
[581] GIURISPRUDENZA	F	2017-01-25	2017-01-30	6	2	28.5000	4.75000000
[581] GIURISPRUDENZA	F	2017-06-29	2017-07-11	13	3	28.6667	2.20513077
[581] GIURISPRUDENZA	M	2017-01-25	2017-02-07	14	3	29.3333	2.09523571
[581] GIURISPRUDENZA	F	2017-05-05	2017-05-29	25	2	30.0000	1.20000000
[581] GIURISPRUDENZA	F	2017-01-23	2017-02-15	24	3	27.6667	1.15277917

Visualizzazione



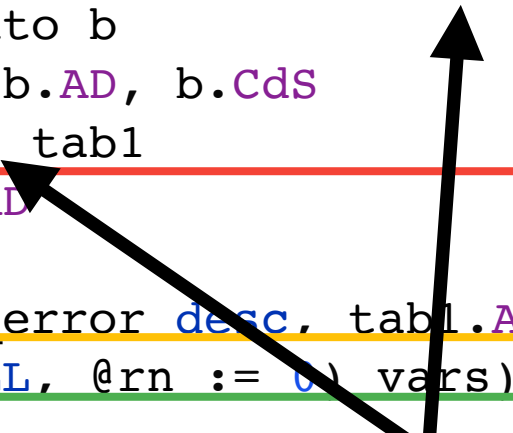
6. Trial & Error Queries

```
select tab3.CdS,
       tab3.AD,
       tab3.studenti,
       tab3.totale_tentativi,
       tab3.trial_error
from (
  select tab2.CdS,
         tab2.AD,
         tab2.studenti,
         tab2.totale_tentativi,
         tab2.trial_error,
         @rn := IF(@prev = tab2.CdS, @rn + 1, 1) AS rn,
         @prev := tab2.CdS
  from (
    select tab1.CdS,
           tab1.AD,
           count(*) as studenti,
           sum(tab1.tentativi) as totale_tentativi,
           avg(tab1.tentativi) as trial_error
    from (
      select b.CdS,
             b.AD,
             sum(b.Superamento+b.Insufficienza+b.Ritiro) as tentativi
      from bos_denormalizzato b
      group by b.Studente, b.AD, b.CdS
      having tentativi > 0) tab1
    group by tab1.CdS, tab1.AD
    having studenti > 1
    order by tab1.CdS, trial_error desc, tab1.AD) tab2
  join (select @prev := NULL, @rn := 0) vars) tab3
where rn <= 3;
```

Ordinamento per CdS e limit

Media tentativi per materia

Tentativi per materia e studente



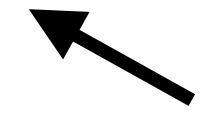
Gli insufficienti e i ritirati sono considerati
Gli assenti non sono considerati

```
select c.cds,
       a.ad,
       tab3.studenti,
       tab3.totale_tentativi,
       tab3.trial_error
from (
  select tab2.cdscod,
         tab2.adcod,
         tab2.studenti,
         tab2.totale_tentativi,
         tab2.trial_error,
         @rn := IF(@prev = tab2.cdscod, @rn + 1, 1) AS rn,
         @prev := tab2.cdscod
  from (
    select tab1.cdscod,
           tab1.adcod,
           count(*) as studenti,
           sum(tab1.tentativi) as totale_tentativi,
           avg(tab1.tentativi) as trial_error
    from (
      select a.cdscod,
             a.adcod,
             sum(i.Superamento+i.Insufficienza+i.Ritiro)+1 as tentativi
      from appelli a
      join iscrizioni i on a.appcod = i.appcod
      group by i.Studente, a.adcod, a.cdscod
      having tentativi > 0) tab1
    group by tab1.cdscod, tab1.adcod
    having studenti > 1
    order by tab1.cdscod, trial error desc, tab1.adcod) tab2
  join (select @prev := NULL, @rn := 0) vars) tab3
  join cds c on tab3.cdscod = c.cdscod
  join ad a on tab3.adcod = a.adcod
where rn <= 3;
```

Ordinamento per CdS e limit

Media tentativi per materia

Tentativi per materia e studente



Join

6. Trial & Error

Tempo di esecuzione

DB denormalizzato	DB normalizzato
755 ms	700 ms

Risultati

CdS	AD	Studenti	Tentativi	Media T&E
[581] GIURISPRUDENZA	[A5810064] DIRITTO PROCESSUALE CIVILE C.P.	4	6	1.5000
[581] GIURISPRUDENZA	[A5810070] MEDICINA LEGALE	43	58	1.3488
[581] GIURISPRUDENZA	[A5810015] DIRITTO PENALE	237	319	1.3460

Visualizzazione

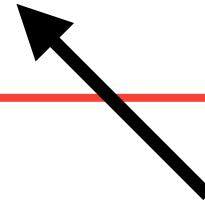


7. Where are you from?

Queries

```
select tab1.CdS,  
       tab1.CittNaz,  
       count(*)  
from (  
    select distinct(b.Studente), Studenti non italiani  
                b.CdS,  
                b.CittNaz  
    from bos_denormalizzato b  
    where b.CittNaz <> 'ITALIA') tab1  
group by tab1.CdS, tab1.CittNaz  
order by tab1.CdS, count(*) desc;
```

```
select tab1.CdS,  
       tab1.CittNaz,  
       count(*)  
from (  
    select distinct(i.studente), Studenti non italiani  
                c.cds,  
                s.cittnaz  
    from studenti s  
    join iscrizioni i on s.studente = i.studente  
    join appelli a on i.appcod = a.appcod  
    join cds c on a.cdscod = c.cdscod  
    where s.CittNaz <> 'ITALIA') tab1  
group by tab1.CdS, tab1.CittNaz  
order by tab1.CdS, count(*) desc;
```



Individuazione dei CdS degli studenti

7. Where are you from?

Tempo di esecuzione

DB denormalizzato	DB normalizzato
115 ms	637 ms

Risultati

CdS	Cittadinanza	Studenti
[581] GIURISPRUDENZA	ALBANIA	20
[581] GIURISPRUDENZA	ROMANIA	14
[581] GIURISPRUDENZA	PERU'	5
[581] GIURISPRUDENZA	SVIZZERA	4
[581] GIURISPRUDENZA	UCRAINA	4

Visualizzazione

