

Trabalho 1 de Infraestrutura para Gestão de Dados:

● Etapa 1:

A modelagem foi desenvolvida com base no modelo fornecido pela professora.

```
create table AIR_AIRLINES as select * from arruda.AIR_AIRLINES;

create table AIR_AIRPLANES as select * from arruda.AIR_AIRPLANES;

create table AIR_AIRPLANE_TYPES as select * from
arruda.AIR_AIRPLANE_TYPES; create table AIR_AIRPORTS as select * from
arruda.AIR_AIRPORTS; create table AIR_AIRPORTS_GEO as select * from
arruda.AIR_AIRPORTS_GEO; create table AIR_BOOKINGS as select * from
arruda.AIR_BOOKINGS; create table AIR_FLIGHTS as select * from
arruda.AIR_FLIGHTS;

create table AIR_FLIGHTS_SCHEDULES as select * from
arruda.AIR_FLIGHTS_SCHEDULES; create table AIR_PASSENGERS as
select * from arruda.AIR_PASSENGERS;

create table AIR_PASSENGERS_DETAILS as select * from
arruda.AIR_PASSENGERS_DETAILS;
```

● Etapa 2 e 3:

Listar o nome completo (primeiro nome + último nome), a idade e a cidade de todos os passageiros do sexo feminino (sex='w') com mais de 40 anos, residentes no país 'BRAZIL'. [resposta sugerida = 143 linhas]

```
SELECT pas.firstname, pas.lastname, to_char(sysdate, 'YYYY')
to_char(det.birthdate, 'YYYY') idade, det.city
FROM AIR_PASSENGERS pas, AIR_PASSENGERS_DETAILS det
WHERE pas.passenger_id = det.passenger_id
AND birthdate <= ADD_MONTHS(sysdate, -40*12)
AND det.sex = 'w'
AND det.country = 'BRAZIL';
```

OPERATION	OBJECT_NAME	OPTIONS	CARDINALITY	COST
SELECT STATEMENT				202
HASH JOIN				202
Access Predicates				
PAS.PASSENGER_ID=DET.PASSENGER_ID				
TABLE ACCESS	AIR_PASSENGERS_DETAILS	FULL	35	150
Filter Predicates				
AND				
DET.COUNTRY='BRAZIL'				
DET.SEX='w'				
BIRTHDATE<=ADD_MONTHS(SYSDATE@1,-480)				
TABLE ACCESS	AIR_PASSENGERS	FULL	36095	51

Listar o nome da companhia aérea, o identificador da aeronave, o nome do tipo de aeronave e o número de todos os voos operados por essa companhia aérea (independentemente de a aeronave ser de sua propriedade) que saem e chegam em aeroportos localizados no país 'BRAZIL'.

```

SELECT air.airline_name nome_companhia, pl.airplane_id id_aeronave,
plt.name tipo_aeronave, flg.flightno numero_voo
  FROM AIR_AIRLINES air, AIR_AIRPLANES pl, AIR_AIRPLANE_TYPES plt,
AIR_FLIGHTS flg
   WHERE air.airline_id = pl.airline_id
   AND pl.airplane_type_id = plt.airplane_type_id
   AND pl.airplane_id = flg.airplane_id;

```

OPERATION	OBJECT_NAME	OPTIONS	CARDINALITY	COST
SELECT STATEMENT				
HASH JOIN			1498	19
Access Predicates			1498	19
PL.AIRPLANE_TYPE_ID=PLT.AIRPLANE_TYPE_ID				
TABLE ACCESS	AIR_AIRPLANE_TYPES	FULL	342	3
HASH JOIN			1498	16
Access Predicates				
AIR.AIRLINE_ID=PL.AIRLINE_ID				
TABLE ACCESS	AIR_AIRLINES	FULL	113	3
HASH JOIN			1498	13
Access Predicates				
PL.AIRPLANE_ID=FLG.AIRPLANE_ID				
TABLE ACCESS	AIR_FLIGHTS	FULL	1498	6
TABLE ACCESS	AIR_AIRPLANES	FULL	5583	7

Listar o número do voo, o nome do aeroporto de saída e o nome do aeroporto de destino, o nome completo (primeiro e último nome) e o assento de cada passageiro, para todos os voos que partem no dia do seu aniversário neste ano (caso a consulta não retorne nenhuma linha, faça para o dia subsequente até encontrar uma data que retorne alguma linha).

```

SELECT origem.num_voo, origem.aeroporto_origem, dest.aeroporto_dest,
abk.seat, pas.firstname, pas.lastname
  FROM (SELECT afs.flightno num_voo, apt.name aeroporto_origem
        FROM AIR_FLIGHTS_SCHEDULES afs, AIR_AIRPORTS apt
        WHERE afs.from_airport_id = apt.airport_id) origem,
        (SELECT afs.flightno num_voo, apt.name aeroporto_dest
        FROM AIR_FLIGHTS_SCHEDULES afs, AIR_AIRPORTS apt
        WHERE afs.to_airport_id = apt.airport_id) dest, AIR_FLIGHTS aif,
AIR_BOOKINGS abk, AIR_PASSENGERS pas
   WHERE origem.num_voo = dest.num_voo
   AND dest.num_voo = aif.flightno
   AND aif.flight_id = abk.flight_id
   AND abk.passenger_id = pas.passenger_id
   AND to_char(aif.departure, 'dd/mm/YY') = to_date('15/03/23', 'dd/mm/YY');

```

OPERATION	OBJECT_NAME	OPTIONS	CARDINALITY	COST
HASH JOIN	ABK.PASSENGER_ID=PAS.PASSENGER_ID			1833 193
Access Predicates	AIF.FLIGHT_ID=ABK.FLIGHT_ID			
HASH JOIN			15	46
Access Predicates	AFS.TO_AIRPORT_ID=APT.AIRPORT_ID			
HASH JOIN			15	32
Access Predicates	AFS.FROM_AIRPORT_ID=APT.AIRPORT_ID			
HASH JOIN			15	18
Access Predicates	AFS.FLIGHTNO=AFS.FLIGHTNO			
HASH JOIN			15	12
Access Predicates	AFS.FLIGHTNO=AIF.FLIGHTNO			
TABLE ACCESS	AIR_FLIGHTS	FULL	15	6
Filter Predicates	TO_CHAR(INTERNAL_FUNCTION(AIF.DEPARTURE),'dd/mm/YY')=TO_DATE('15/03/23','dd/mm/YY')			
TABLE ACCESS	AIR_FLIGHTS_SCHEDULES	FULL	1424	6
TABLE ACCESS	AIR_FLIGHTS_SCHEDULES	FULL	1424	6
TABLE ACCESS	AIR_AIRPORTS	FULL	9854	14
TABLE ACCESS	AIR_AIRPORTS	FULL	9854	14
TABLE ACCESS	AIR_BOOKINGS	FULL	122244	147
TABLE ACCESS	AIR_PASSENGERS	FULL	36095	51

Listar o nome da companhia aérea bem como a data e a hora de saída de todos os voos que chegam para a cidade de 'NEW YORK' que partem às terças, quartas ou quintas-feiras, no mês do seu aniversário (caso a consulta não retorna nenhuma linha, faça para o mês subsequente até encontrar um

```
SELECT air.airline_name, afs.departure
FROM AIR_AIRLINES air, AIR_FLIGHTS_SCHEDULES afs, AIR_AIRPORTS apt,
AIR_AIRPORTS_GEO geo
WHERE air.airline_id = afs.airline_id
AND afs.to_airport_id = apt.airport_id
AND apt.airport_id = geo.airport_id
AND geo.city = 'NEW YORK'
AND (afs.tuesday = 1 OR afs.wednesday = 1 OR afs.thursday = 1)
AND to_char(afs.departure, 'mm') = '05';
```

OPERATION	OBJECT_NAME	OPTIONS	CARDINALITY	COST
HASH JOIN	AIR.AIRLINE_ID=AFS.AIRLINE_ID			1 43
Access Predicates				
AND				
AFS.TO_AIRPORT_ID=APT.AIRPORT_ID				
APT.AIRPORT_ID=GEO.AIRPORT_ID				
MERGE JOIN		CARTESIAN	14	29
TABLE ACCESS	AIR_AIRPORTS_GEO	FULL	1	23
Filter Predicates	GEO.CITY='NEW YORK'			
BUFFER				
TABLE ACCESS	AIR_FLIGHTS_SCHEDULES	SORT	12	6
Filter Predicates		FULL	12	6
AND				
TO_CHAR(INTERNAL_FUNCTION(AFS.DEPARTURE),'mm')='05'				
OR				
AFS.TUESDAY=1				
AFS.WEDNESDAY=1				
AFS.THURSDAY=1				
TABLE ACCESS	AIR_AIRPORTS	FULL	9854	14
TABLE ACCESS	AIR_AIRLINES	FULL	113	3

mês que retorne alguma linha).

Crie uma consulta que seja resolvida adequadamente com um acesso hash em um cluster com pelo menos duas tabelas. A consulta deve utilizar todas as tabelas do cluster e pelo menos outra tabela fora dele: Liste o nome completo do passageiro, o nome do aeroporto e a cidade na qual ele está partindo de todos os passageiros com sobrenome iniciado por "L".

```
SELECT pas.FIRSTNAME, pas.LASTNAME, air.NAME, geo.CITY
FROM AIR_PASSENGERS pas
JOIN AIR_BOOKINGS abk ON pas.PASSENGER_ID = abk.PASSENGER_ID
JOIN AIR_FLIGHTS flg ON abk.FLIGHT_ID = flg.FLIGHT_ID
JOIN AIR_AIRPORTS air ON flg.FROM_AIRPORT_ID = air.AIRPORT_ID
JOIN AIR_AIRPORTS_GEO geo ON air.AIRPORT_ID = geo.AIRPORT_ID WHERE
pas.LASTNAME LIKE 'L%';
```

OPERATION	OBJECT_NAME	OPTIONS	CARDINALITY	COST
SELECT STATEMENT				242
HASH JOIN				242
Access Predicates AIR.AIRPORT_ID=Geo.AIRPORT_ID				
HASH JOIN			2255	218
Access Predicates FLG.FROM_AIRPORT_ID=AIR.AIRPORT_ID				
HASH JOIN			2255	204
Access Predicates ABK.FLIGHT_ID=FLG.FLIGHT_ID				
TABLE ACCESS AIR_FLIGHTS		FULL	1498	6
HASH JOIN			2255	198
Access Predicates PAS.PASSENGER_ID=ABK.PASSENGER_ID				
TABLE ACCESS AIR_PASSENGERS		FULL	646	51
Filter Predicates PAS.LASTNAME LIKE 'L%'				
TABLE ACCESS AIR_BOOKINGS		FULL	122244	147
TABLE ACCESS AIR_AIRPORTS		FULL	9854	14
TABLE ACCESS AIR_AIRPORTS_GEO		FULL	9854	23

● Etapa 4:

Nesta etapa ocorre a criação de todas as estruturas de acesso otimizado necessárias para que a consulta seja executada da forma mais otimizada possível, as consultas são refeitas e seu plano de execução listado.

Criação de constraints:

```
-- Tabela AIRLINES
ALTER TABLE AIR_AIRLINES
ADD CONSTRAINT airlines_pk PRIMARY KEY (airline_id);

-- Tabela AIRPLANE_TYPES
ALTER TABLE AIR_AIRPLANE_TYPES
ADD CONSTRAINT airplane_types_pk PRIMARY KEY (airplane_type_id);

-- Tabela AIRPLANES
ALTER TABLE AIR_AIRPLANES
ADD CONSTRAINT airplanes_pk PRIMARY KEY (airplane_id), ADD CONSTRAINT
airplanes_airlines_fk FOREIGN KEY (airline_id) REFERENCES
AIR_AIRLINES (airline_id),
ADD CONSTRAINT airplanes_types_fk FOREIGN KEY (airplane_type_id)
REFERENCES AIR_AIRPLANE_TYPES (airplane_type_id);

-- Tabela PASSENGERS
```

```

ALTER TABLE AIR_PASSENGERS
ADD CONSTRAINT passengers_pk PRIMARY KEY (passenger_id),
ADD CONSTRAINT passengers_passport_ak UNIQUE (passportno);

-- Tabela PASSENGERS_DETAILS
ALTER TABLE AIR_PASSENGERS_DETAILS
ADD CONSTRAINT passengers_details_pk PRIMARY KEY (passenger_id),
ADD CONSTRAINT passengers_fk FOREIGN KEY (passenger_id) REFERENCES
AIR_PASSENGERS (passenger_id);

-- Tabela AIRPORTS
ALTER TABLE AIR_AIRPORTS
ADD CONSTRAINT airports_pk PRIMARY KEY (airport_id),
ADD CONSTRAINT airports_icao_ak UNIQUE (icao);

-- Tabela AIRPORTS_GEO
ALTER TABLE AIR_AIRPORTS_GEO
ADD CONSTRAINT airports_geo_pk PRIMARY KEY (airport_id),
ADD CONSTRAINT airports_geo_fk FOREIGN KEY (airport_id) REFERENCES
AIR_AIRPORTS (airport_id);

-- Tabela FLIGHTS_SCHEDULES
ALTER TABLE AIR_FLIGHTS_SCHEDULES
ADD CONSTRAINT flights_schedules_pk PRIMARY KEY (flightno),
ADD CONSTRAINT flight_schedules_airlines_fk FOREIGN KEY (airline_id)
REFERENCES AIR_AIRLINES (airline_id),
ADD CONSTRAINT from_airports_fk FOREIGN KEY (from_airport_id) REFERENCES
AIR_AIRPORTS (airport_id),
ADD CONSTRAINT to_airport_geo_fk FOREIGN KEY (to_airport_id) REFERENCES
AIR_AIRPORTS (airport_id);

-- Tabela FLIGHTS
ALTER TABLE AIR_FLIGHTS
ADD CONSTRAINT flights_pk PRIMARY KEY (flight_id),
ADD CONSTRAINT flight_schedules_fk FOREIGN KEY (flightno) REFERENCES
AIR_FLIGHTS_SCHEDULES (flightno),
ADD CONSTRAINT flight_airlines_fk FOREIGN KEY (airline_id) REFERENCES
AIR_AIRLINES (airline_id),
ADD CONSTRAINT flights_from_airports_fk FOREIGN KEY (from_airport_id)
REFERENCES AIR_AIRPORTS (airport_id),
ADD CONSTRAINT flights_to_airport_fk FOREIGN KEY (to_airport_id)
REFERENCES AIR_AIRPORTS (airport_id);

-- Tabela BOOKINGS
ALTER TABLE AIR_BOOKINGS
ADD CONSTRAINT bookings_pk PRIMARY KEY (booking_id),
ADD CONSTRAINT booking_passengers_fk FOREIGN KEY (passenger_id)
REFERENCES AIR_PASSENGERS (passenger_id),
ADD CONSTRAINT booking_flights_fk FOREIGN KEY (flight_id) REFERENCES
AIR_FLIGHTS (flight_id),
ADD CONSTRAINT booking_flights_ak UNIQUE (flight_id),
ADD CONSTRAINT booking_seats_ak UNIQUE (seat);

```

PESQUISA 1:

B-Tree+

```
CREATE TABLE AIR_PASSENGERS_BTI (  
    PASSENGER_ID NUMBER(12,0) NOT NULL,  
    PASSPORTNO CHAR(9 BYTE) NOT NULL,  
    FIRSTNAME VARCHAR2(100 BYTE) NOT NULL,  
    LASTNAME VARCHAR2(100 BYTE) NOT NULL,  
    CONSTRAINT air_passengers_bti_pk PRIMARY KEY (PASSENGER_ID),  
    CONSTRAINT passengers_passport_bti_ak UNIQUE (PASSPORTNO)  
)  
  
CLUSTER air_passenger_index(passenger_id);  
  
CREATE TABLE AIR_PASSENGERS_DET_BTI  
(PASSENGER_ID NUMBER(12,0),  
    BIRTHDATE DATE,  
    SEX CHAR(1 BYTE),  
    STREET VARCHAR2(100 BYTE),  
    CITY VARCHAR2(100 BYTE),  
    ZIP NUMBER(5,0),  
    COUNTRY VARCHAR2(100 BYTE),  
    EMAILADDRESS VARCHAR2(120 BYTE),  
    TELEPHONENO VARCHAR2(30 BYTE),  
    CONSTRAINT passengers_det_bti_pk PRIMARY KEY (passenger_id),  
    CONSTRAINT air_passengers_bti_fk FOREIGN KEY (passenger_id) REFERENCES  
    AIR_PASSENGERS_BTI (passenger_id)  
)  
  
INSERT INTO AIR_PASSENGERS_BTI(PASSENGER_ID, PASSPORTNO, FIRSTNAME,  
    LASTNAME)  
SELECT PASSENGER_ID, PASSPORTNO, FIRSTNAME, LASTNAME from air_passengers  
where rownum < 3000;  
  
INSERT INTO AIR_PASSENGERS_DET_BTI(PASSENGER_ID, BIRTHDATE, SEX, STREET,  
    CITY, ZIP, COUNTRY, EMAILADDRESS, TELEPHONENO) select pas.PASSENGER_ID,  
    det.BIRTHDATE, det.SEX, det.STREET, det.CITY, det.ZIP, det.COUNTRY,  
    det.EMAILADDRESS, det.TELEPHONENO from AIR_PASSENGERS_DETAILS det,  
    AIR_PASSENGERS_BTI pas where pas.PASSENGER_ID = det.PASSENGER_ID;  
  
SELECT pas.firstname, pas.lastname, to_char(sysdate, 'YYYY')  
to_char(det.birthdate, 'YYYY') idade, det.city  
    FROM AIR_PASSENGERS_BTI pas, AIR_PASSENGERS_DET_BTI det  
    WHERE pas.passenger_id = det.passenger_id  
    AND birthdate <= ADD_MONTHS(sysdate,-40*12)  
    AND det.sex = 'w'  
    AND det.country = 'BRAZIL';
```

PESQUISA 2:

Clusters, hash:

```
CREATE CLUSTER air_airplanes_hash(airplane_type_id numeric(3))
HASHKEYS 50;

CREATE TABLE air_airplane_types_hash(
AIRPLANE_TYPE_ID NUMBER(3,0),

NAME VARCHAR2(50 BYTE),
CONSTRAINT air_airplane_types_hash_pk PRIMARY KEY (airplane_type_id)
)
CLUSTER air_airplanes_hash(AIRPLANE_TYPE_ID);

INSERT INTO air_airplane_types_hash(AIRPLANE_TYPE_ID,NAME)
SELECT AIRPLANE_TYPE_ID, NAME
FROM air_airplane_types;

CREATE TABLE AIR_AIRPLANES_HASH (
AIRPLANE_ID NUMBER(5,0),
AIRLINE_ID NUMBER(*,0),
AIRPLANE_TYPE_ID NUMBER(3,0),
CAPACITY NUMBER(3,0),
CONSTRAINT air_airplanes_hash_pk PRIMARY KEY (airplane_id),
CONSTRAINT airplanes_airlines_hash_fk FOREIGN KEY (airline_id) REFERENCES
AIR_AIRLINES (airline_id),
CONSTRAINT airplanes_type_hash_fk FOREIGN KEY
(airplane_type_id) REFERENCES air_airplane_types_hash
(airplane_type_id) )
CLUSTER air_airplanes_hash(AIRPLANE_TYPE_ID);

INSERT INTO
AIR_AIRPLANES_HASH(AIRPLANE_ID,AIRLINE_ID,AIRPLANE_TYPE_ID,CAPACITY)
SELECT AIRPLANE_ID,AIRLINE_ID,AIRPLANE_TYPE_ID,CAPACITY
FROM AIR_AIRPLANES;
```

Pesquisa:

```
SELECT air.airline_name nome_companhia, pl.airplane_id id_aeronave,
plt.name tipo_aeronave, flg.flightno numero_voo
FROM AIR_AIRLINES air, AIR_AIRPLANES_HASH pl,
air_airplane_types_hash plt, AIR_FLIGHTS flg
WHERE air.airline_id = pl.airline_id
AND pl.airplane_type_id = plt.airplane_type_id
AND pl.airplane_id = flg.airplane_id;
```

OPERATION	OBJECT_NAME	OPTIONS	CARDINALITY	COST
SELECT STATEMENT			1501	32
HASH JOIN			1501	32
Access Predicates AIR.AIRLINE_ID=PL.AIRLINE_ID				
TABLE ACCESS	AIR_AIRLINES	FULL	113	3
HASH JOIN			1501	29
Access Predicates PL.AIRPLANE_ID=FLG.AIRPLANE_ID				
TABLE ACCESS	AIR_FLIGHTS	FULL	1498	6
NESTED LOOPS			6042	23
TABLE ACCESS	AIR_AIRPLANE_TYPES_HASH	FULL	359	23
TABLE ACCESS	AIR_AIRPLANES_HASH	HASH	17	
Access Predicates PL.AIRPLANE_TYPE_ID=PLT.AIRPLANE_TYPE_ID				

PESQUISA 3:

```
CREATE CLUSTER air_flights_hash(flightno CHAR(8))
HASHKEYS 10;

CREATE TABLE FLIGHTS_SCHEDULES_HASH(
  FLIGHTNO CHAR(8 BYTE),
  AIRLINE_ID NUMBER(5,0),
  FROM_AIRPORT_ID NUMBER(5,0),
  TO_AIRPORT_ID NUMBER(5,0),
  DEPARTURE DATE,
  ARRIVAL DATE,
  MONDAY NUMBER(1,0),
  TUESDAY NUMBER(1,0),
  WEDNESDAY NUMBER(1,0),
  THURSDAY NUMBER(1,0),
  FRIDAY NUMBER(1,0),
  SATURDAY NUMBER(1,0),
  SUNDAY NUMBER(1,0),
  CONSTRAINT flights_sch_hash_pk PRIMARY KEY (flightno),
  CONSTRAINT flight_sch_airline_hash_fk FOREIGN KEY (airline_id) REFERENCES
  AIR_AIRLINES (airline_id),
  CONSTRAINT from_airport_hash_fk FOREIGN KEY (from_airport_id) REFERENCES
  AIR_AIRPORTS (airport_id),
  CONSTRAINT to_aip_geo_hash_fk FOREIGN KEY (to_airport_id) REFERENCES
  AIR_AIRPORTS (airport_id)
)
CLUSTER air_flights_hash(FLIGHTNO);

INSERT INTO FLIGHTS_SCHEDULES_HASH(FLIGHTNO, AIRLINE_ID, FROM_AIRPORT_ID,
TO_AIRPORT_ID, DEPARTURE, ARRIVAL, MONDAY, TUESDAY, WEDNESDAY, THURSDAY,
FRIDAY, SATURDAY, SUNDAY)
SELECT FLIGHTNO, AIRLINE_ID, FROM_AIRPORT_ID, TO_AIRPORT_ID, DEPARTURE,
ARRIVAL, MONDAY, TUESDAY, WEDNESDAY, THURSDAY, FRIDAY, SATURDAY, SUNDAY
FROM AIR_FLIGHTS_SCHEDULES;

CREATE TABLE AIR_FLIGHTS_HASH(
  FLIGHT_ID NUMBER(10,0),
  FLIGHTNO CHAR(8 BYTE),
  AIRLINE_ID NUMBER(5,0),
  FROM_AIRPORT_ID NUMBER(5,0),
  TO_AIRPORT_ID NUMBER(5,0),
  AIRPLANE_ID NUMBER(5,0),
  DEPARTURE TIMESTAMP (6),
  ARRIVAL TIMESTAMP (6),
  CONSTRAINT air_flights_hash_pk PRIMARY KEY (flight_id),
  CONSTRAINT flight_schedule_hash_fk FOREIGN KEY (flightno) REFERENCES
  FLIGHTS_SCHEDULES_HASH (flightno),
  CONSTRAINT flight_airline_hash_fk FOREIGN KEY (airline_id) REFERENCES
  AIR_AIRLINES (airline_id),
  CONSTRAINT flights_from_airport_hash_fk FOREIGN KEY (from_airport_id)
REFERENCES AIR_AIRPORTS (airport_id),
```



```

CONSTRAINT flights_to_airport_hash_fk FOREIGN KEY (to_airport_id)
REFERENCES AIR_AIRPORTS (airport_id)
)
CLUSTER air_flights_hash (FLIGHTNO);

```

```

INSERT INTO
AIR_FLIGHTS_HASH (FLIGHT_ID, FLIGHTNO, AIRLINE_ID, FROM_AIRPORT_ID, TO_AIRPORT_ID, AIRPLANE_ID, DEPARTURE, ARRIVAL)
SELECT
FLIGHT_ID, FLIGHTNO, AIRLINE_ID, FROM_AIRPORT_ID, TO_AIRPORT_ID, AIRPLANE_ID, DEPARTURE, ARRIVAL
FROM air_flights;

```

```

SELECT origem.num_voo, origem.aeroporto_origem, dest.aeroporto_dest,
abk.seat, pas.firstname, pas.lastname

```

```

FROM (SELECT afs.flightno num_voo, apt.name aeroporto_origem
FROM FLIGHTS_SCHEDULES_HASH afs, AIR_AIRPORTS apt
WHERE afs.from_airport_id = apt.airport_id) origem,
(SELECT afs.flightno num_voo, apt.name aeroporto_dest
FROM FLIGHTS_SCHEDULES_HASH afs, AIR_AIRPORTS apt
WHERE afs.to_airport_id = apt.airport_id) dest, AIR_FLIGHTS_HASH aif,
AIR_BOOKINGS abk, AIR_PASSENGERS pas
WHERE origem.num_voo = dest.num_voo
AND dest.num_voo = aif.flightno
AND aif.flight_id = abk.flight_id
AND abk.passenger_id = pas.passenger_id
AND to_char(aif.departure, 'dd/mm/YY') = to_date('15/03/23', 'dd/mm/YY');

```

OPERATION	OBJECT_NAME	OPTIONS	CARDINALITY	COST
SELECT STATEMENT				122 212
HASH JOIN				122 212
Access Predicates				
NESTED LOOPS				122 212
NESTED LOOPS				
STATISTICS COLLECTOR				
HASH JOIN				122 161
Access Predicates				
HASH JOIN				1 14
Access Predicates	TO_AIRPORT_ID=APT.AIRPORT_ID			
NESTED LOOPS				1 14
NESTED LOOPS				1 14
STATISTICS COLLECTOR				
HASH JOIN				1 13
Access Predicates				
NESTED				1 13
STATISTICS COLLECTOR				
FLIGHTS_SCHEDULES_HASH				1 12
AIR_FLIGHTS_HASH		FULL		1424 12
Access Predicates	AIF.FLIGHTNO=AFS.FLIGHTNO			
Filter Predicates	TO_CHAR(INTERNAL_FUNCTION(AIF.DEPARTURE), 'dd/mm/YY')=TO_DATE('15/03/23', 'dd/mm/YY')			
TABLE ACCESS	AIR_AIRPORTS	BY INDEX ROWID		1 1
AIRPORTS_PK		UNIQUE SCAN		1 0
Access Predicates				
TABLE ACCESS	AIR_AIRPORTS	FULL		1 1
INDEX	AIRPORTS_PK	UNIQUE SCAN		1 0
Access Predicates				
TABLE ACCESS	AIR_AIRPORTS	BY INDEX ROWID		1 1
TABLE ACCESS	AIR_AIRPORTS	FULL		1 1
TABLE ACCESS	AIR_BOOKINGS	FULL		122244 147
INDEX	PASSENGERS_PK	UNIQUE SCAN		
Access Predicates				
TABLE ACCESS	AIR_PASSENGERS	BY INDEX ROWID		1 51
TABLE ACCESS	AIR_PASSENGERS	FULL		36095 51

PESQUISA 4:

```
CREATE CLUSTER air_airports_index(airport_id NUMBER(5))
INDEX;
CREATE INDEX idx_air_airports_index ON CLUSTER air_airports_index;
```

```
CREATE TABLE AIR_AIRPORTS_BTI(
AIRPORT_ID NUMBER(5,0),
IATA CHAR(3 BYTE),
ICAO CHAR(4 BYTE),
NAME VARCHAR2(50 BYTE),
CONSTRAINT air_airport_bti_pk PRIMARY KEY (airport_id),
CONSTRAINT airport_icao_bti_ak UNIQUE (icao)
)
CLUSTER air_airports_index(airport_id);
```

```
INSERT INTO AIR_AIRPORTS_BTI(AIRPORT_ID,IATA,ICAO,NAME)
SELECT aa.AIRPORT_ID,aa.IATA,aa.ICAO,aa.NAME
FROM AIR_AIRPORTS aa
```

```
where rownum < 3000;
```

```
CREATE TABLE AIRPORTS_GEO_BTI(
AIRPORT_ID NUMBER(5,0),
NAME VARCHAR2(50 BYTE),
CITY VARCHAR2(50 BYTE),
COUNTRY VARCHAR2(50 BYTE),
LATITUDE NUMBER(11,8),
LONGITUDE NUMBER(11,8),
CONSTRAINT airport_geo_bti_pk PRIMARY KEY (airport_id),
CONSTRAINT airport_geo_bti_fk FOREIGN KEY (airport_id) REFERENCES
AIR_AIRPORTS_BTI(airport_id)
)
CLUSTER air_airports_index(airport_id);
```

```
INSERT INTO
AIRPORTS_GEO_BTI(AIRPORT_ID,NAME,CITY,COUNTRY,LATITUDE,LONGITUDE)
SELECT
geo.AIRPORT_ID,geo.NAME,geo.CITY,geo.COUNTRY,geo.LATITUDE,geo.LONGITUDE
FROM AIR_AIRPORTS_GEO geo, AIR_AIRPORTS_BTI aab
where geo.AIRPORT_ID = aab.AIRPORT_ID;
```

```
SELECT air.airline_name, afs.departure
FROM AIR_AIRLINES air, AIR_FLIGHTS_SCHEDULES afs, AIR_AIRPORTS_BTI pas,
AIRPORTS_GEO_BTI aag
WHERE air.airline_id = afs.airline_id
AND afs.to_airport_id = pas.airport_id
AND pas.airport_id = aag.airport_id
AND aag.city = 'NEW YORK'
AND (afs.tuesday = 1 OR afs.wednesday = 1 OR afs.thursday = 1)
AND to_char(afs.departure, 'mm') = '05';
```

PESQUISA 5:

```
CREATE CLUSTER air_airport_hash(airport_id numeric(5))
HASHKEYS 50;
```

```
CREATE TABLE AIR_AIRPORT_HASH (AIRPORT_ID NUMBER(5,0),
IATA CHAR(3 BYTE),
ICAO CHAR(4 BYTE),
NAME VARCHAR2(50 BYTE),
CONSTRAINT air_airport_bti_pk PRIMARY KEY (airport_id),
CONSTRAINT airport_icao_bti_ak UNIQUE (icao)
)
CLUSTER air_airport_hash(airport_id);
```

```
INSERT INTO AIR_AIRPORT_HASH(AIRPORT_ID,IATA,ICAO,NAME)
SELECT aa.AIRPORT_ID,aa.IATA,aa.ICAO,aa.NAME
FROM AIR_AIRPORTS aa
where rownum < 3000;
```

```
CREATE TABLE AIRPORTS_GEO_HASH(
AIRPORT_ID NUMBER(5,0),
NAME VARCHAR2(50 BYTE),
CITY VARCHAR2(50 BYTE),
COUNTRY VARCHAR2(50 BYTE),
```

```
LATITUDE NUMBER(11,8),
LONGITUDE NUMBER(11,8),
CONSTRAINT airport_geo_hash_pk PRIMARY KEY (airport_id),
CONSTRAINT airport_geo_hash_fk FOREIGN KEY (airport_id) REFERENCES
AIR_AIRPORT_HASH(airport_id)
)
CLUSTER air_airport_hash(airport_id);
```

```
INSERT INTO
AIRPORTS_GEO_HASH(AIRPORT_ID,NAME,CITY,COUNTRY,LATITUDE,LONGITUDE)
SELECT
aag.AIRPORT_ID,aag.NAME,aag.CITY,aag.COUNTRY,aag.LATITUDE,aag.LONGITUDE
FROM AIR_AIRPORTS_GEO aag, AIR_AIRPORT_HASH aab
where aag.AIRPORT_ID = aab.AIRPORT_ID;
```

```
SELECT pas.firstname, pas.lastname, air.name, aag.city
FROM air_passengers pas, air_bookings abk, air_flights aif,
AIR_AIRPORT_HASH air, AIRPORTS_GEO_HASH aag
WHERE pas.passenger_id = abk.passenger_id
AND aif.flight_id = abk.flight_id
AND aif.from_airport_id = air.airport_id
AND air.airport_id = aag.airport_id
AND pas.lastname LIKE 'L%';
```