1. For each user view from milestone # 1 formulate a few requests for information (use English sentences) which require browsing the business data from your Oracle database implemented in milestone # 2. You need to formulate 28 requests (group #4).

1. Airline Employee who works at Air Canada selects all the company's flights that have landed, with the time of arrival and the aircraft model who operated the flight.
2. Airport Employee who works at Toronto Airport selects all cities that are connected by flights to/from Toronto.
3. Airport Employee select all airlines that have a Boeing aircraft, and the model of the Boeing airplane.
4. Airport Employee selects the boarding time of all the domestic and international flights departing from Canadian airports, and the arrival destination (city and country).
5. Airport Employee select all Boeing aircrafts that have flown more than 5,000 hours.
6. Airport Employee selects international flights which departed from Canadian airports after November 1, 2023, 23:59 GMT -5 and already arrived at the destination.
7. Airport Employee gets the number of seats of each flight departing from Toronto and operated by a Boeing aircraft.

8. Display the flight number and departure city for flights operated by airlines with the airline ID 0538.
9. Show the person ID, first and last names, and time zone for all employees working at the airport.
10. Organise passengers based on their passport expiry date, and present their ID and full name, excluding the passenger with ID 2009.
11. Provide a list that includes the names, IDs, phone numbers, and job positions of all airline employees, along with the names of the respective airlines they work for.
12. From a list of passengers, present the expiration date and identification number for the individual named Mouralii Ki, and include the corresponding name in the display as well.
13. From the departure and Canadian flight records, show the city associated with the domestic flight departure having the flight number AC1450.
14. Display details about passengers whose passport expiration date is on December 7, 2032

15. Select the checkInStatus of all the passengertour that fly to Honolulu and group them by checkinstatus.
16. Select the AirlineEmployees' fullName and gender that run the Aircraft whose aircraftID is AE3665.
17. Select the BoardingGate information at Los Angeles.
18. Select passportNumber and IssueCountry of passengers who take flights that delay for over 0 minute.
19. Select passengertours that take a seat with L04 from all the passengertour.
20. Select the information of the flight with a flightid as AC1
21. Select the bookingcode of tours whose flights depart from Canada and arrive to city London

22. Check flightID, Flight number, Status of the flights where flights are delayed more than 10 minutes.
23. Get passenger’s first and last name, date of birth, where passenger’s id is '1005' and group by phone gender and salutation.
24. Find the check-in status of the passenger where the booking code is 'T1232’.
25. Select seat code, check-in status, booking code where passengers able to select seat or not.
26. Select all passengers' information.
27. Sort the document information ordered by passport number along with expiry date and issue country.
28. Get the passenger’s date of birth where passenger’s id =1005 and name is 'Mouralii Ki’.
29. Retrieves the names and phone numbers of passengers whose name contains the word “Dey”
30. Returns the seatcode and check-in status where passengers bookingcode is 'T1231'.

2. For 12 requests (group #4) formulate SQL statements with Oracle XML functions returning XML data such that:
a. All SQL queries access object data from more than one object table using references;
b. All XML documents contain more than two levels of element hierarchy;
c. Four (group # 4) of your documents are to contain XML tags with attributes;
d. Four (group # 4) queries must contain GROUP BY clause;
e. Demonstrate application of XMLFOREST, XMLAGG and XMLROOT functions;
f. All the documents must be well-formed: contain XML declaration and the root element.

```
set echo on
set long 32000
set pagesize 100

/* Question 2) For 12 requests (group #4) formulate SQL statements with Oracle XML
functions returning XML data... */

--  queries (2, 3, 4)

/* Query 2) Airport Employee who works at Toronto Airport selects all cities that
are connected by flights to/from Toronto. */

with aux as (
select f.arrivalairport.airportcode AS AirportCode, f.arrivalairport.city AS City,
f.arrivalairport.country AS Country, f.arrivalairport.name AS Name from
flight f where f.departureairport.city = 'Toronto' UNION
select f.departureairport.airportcode AS AirportCode, f.departureairport.city AS
City, f.departureairport.country AS Country, f.departureairport.name AS
Name from flight f where f.arrivalairport.city = 'Toronto')
select xmlroot(xmlelement("connections", xmlagg(xmlelement("airport",
xmlattributes(a.airportcode as "code"), xmlelement("city", a.city),
xmlelement("country", a.country), xmlelement("name", a.name)))), version '1.0') as
doc from aux a;

/* Query 3) Airport Employee select all airlines that have a Boeing airplane, and
the model of the Boeing airplane. */

select xmlroot(xmlelement("aircraftslist", xmlagg(xmlelement("aircraft",
xmlattributes(a.aircraftid as "id"), xmlelement("make", a.make ),
xmlelement("airline", a.airlineoid.airline_name), xmlelement("model", a.model),
xmlelement("numberofseats", a.numberofseats), xmlelement("hoursflown",
a.hoursflown)))), version '1.0') as doc from aircraft a where a.make = 'Boeing';

/* Query 4) Airport Employee selects the boarding time of all the domestic and
international flights departing from canadian airports, and the arrival
destination (city and country).
Follow: item C, D (Group BY), E
*/

select xmlroot(xmlelement("flightlist", xmlagg(xmlelement("category",
xmlattributes(f.isinternational as "isinternational"), xmlagg(xmlelement("flight",
xmlattributes(f.flightID as "id", f.flightnumber as "number"),
xmlelement("departure", xmlforest(f.departureairport.city as "city",
f.departureairport.country as "country")),  xmlelement("arrival",
xmlforest(f.arrivalairport.city as "city", f.arrivalairport.country as "country"
)),
xmlelement("boardingtime", f.calculateboardingtime()), xmlelement("departuretime",
f.calculatedelayeddeptime()), xmlelement("duration",
f.flightdurationinminutes), xmlelement("status", f.status), xmlelement("airline",
f.airlineoid.airline_name), xmlelement("aircraftid",
```

```sql
f.aircraftoid.aircraftid) ) )))), version '1.0') as doc from flight f where
f.departureairport.country = 'Canada' group by f.isinternational;


--  queries (8, 9, 10)

/* Query 8) Display the flight number and departure city for flights operated by
airlines with the airline ID 0538. */

select xmlroot(xmlelement("departures", xmlagg (xmlelement ("flightnm",
xmlattributes(f.flightnumber as "nm"), xmlelement("city",
f.departureAirport.city)))), version '1.0') as doc from flight f where
f.airlineoid.airline_id= '0538';

/* Query 9) Show the person ID, first and last names, and time zone for all
employees working at the airport.
Follow: item C, E
*/

select xmlroot(xmlelement("employees", xmlagg(xmlelement("employee",
xmlattributes(ar.pid as "ID"), xmlforest(ar.pname.firstn as "given_nm",
ar.pname.lastn as "surname", ar.oairportcode.timezone as "timeZone")))), version
'1.0') as doc2 from Airpemployee ar;

/* Query 10) Organise passengers based on their passport expiry date, and present
their ID and full name, excluding the passenger with ID 2009.
Follow: item C, D (Group BY), E
*/

select xmlroot(xmlelement("passengerlist", xmlagg( xmlelement("passport_exp",
xmlattributes(p.aoid.expiry_date as "exp"), xmlagg(xmlelement("client",
xmlattributes(p.pid as "ID"), xmlelement("name", p.fullname()))))))), version '1.0')
as doc from passenger p where p.pid<> '2009' group by
p.aoid.expiry_date;

-- queries (15, 16, 17)

/* Query 15) Select the checkInStatus of all the passengertour that fly to Toronto
and group them by checkinstatus.
This meet the requirement of using ROOT + ATTRIBUTES + GROUP BY
*/

SELECT XMLROOT(
XMLELEMENT("result",XMLAGG(XMLELEMENT("PassengerStatus",XMLATTRIBUTES(pt.checkinsta
tus), XMLAGG(XMLELEMENT("PassengerInfo",
XMLELEMENT("name",pt.poid.pname),XMLELEMENT("seat",pt.seatcode),XMLELEMENT("flighti
d",pt.foid.flightid)))))),version '1.0') AS document FROM passengertour
pt WHERE pt.foid.arrivalairport.city = 'Toronto' GROUP BY pt.checkinstatus;

/* Query 16) Select the AirlineEmployees' fullName and gender that run the Aircraft
```

```
whose aircraftID is AE3665.
This meet the requirement of using ROOT + FOREST.
*/

SELECT XMLROOT(XMLELEMENT("AirlineEmployee",XMLFOREST(ae.pname as "name", ae.gender
as "gender", ae.role_al as "role")),version '1.0', standalone yes) AS
document FROM airlEmployee ae, Aircraft ac WHERE ac.AircraftID = 'AE3665' AND
ae.airlineoid.airline_id = ac.airlineoid.airline_id;

/* Query 17) Select the BoardingGate information at Los Angeles.
This meet the requirement of using ROOT + AGG + ATTRIBUTE
*/

SELECT
XMLROOT(XMLELEMENT("RESULT",XMLAGG(XMLELEMENT("BoardingGate",XMLATTRIBUTES(bg.bgid)
,XMLELEMENT("GateNumber",bg.gatenumber)))),version
'1.0',standalone yes) AS document FROM BoardingGate bg WHERE bg.aoid.city = 'Los
Angeles';


-- queries (22, 23, 24, 25, 26)

/* Query 22) Check flightID, Flight number, Status of the flights where flights are
delayed more than 10 minutes
*/

select XMLROOT(XMLELEMENT("flights", XMLAGG(XMLELEMENT("flights",
XMLFOREST(f.flightid AS "flightID",f.flightNumber AS
"flightNumber",f.scheduleddeparturetime AS
"scheduleddeperaturetime",f.flightdurationinminutes AS
"flightdurationInMinutes",f.STATUS  AS
"status",f.gnumber AS "gnumber")))), VERSION '1.0') AS document FROM flight f WHERE
f.delayinminutes > 10;

/* Query 23) Get passenger’s first and last name, date of birth, where passenger’s
id is '1005' and group by phone gender and salutation.
*/

Select XMLROOT(XMLElement("PassengerInformation",XMLElement("Document",
XMLATTRIBUTES(p.pid AS
"passportid"),XMLElement("PassengerDetails",XMLAgg(XMLElement("Passenger",XMLFOREST
(p.pname AS "name",p.dob AS "dob")))),XMLElement("passenger",XMLFOREST(
p.phone AS "phone",p.gender AS "gender", p.salutation AS "salutation")))),VERSION
'1.0') FROM passenger p WHERE p.pid = '1005' GROUP BY p.pid, p.phone,
p.gender, p.salutation;

/* Query 24) Find the check-in status of the passenger where the booking code  is
'T1232'.
*/
```

```
select XMLROOT(XMLElement("passengertour", XMLAgg(XMLElement("BookingCode",
XMLAttributes(t.bookingcode AS "BookingCode"), XMLElement("CheckinStatus",
XMLAttributes(t.checkinstatus AS "CheckinStatus"))))), VERSION '1.0') AS document
FROM passengertour t WHERE bookingcode= 'T1232';

/* Query 25) Select seat code, check-in status , booking code where passengers able
to select seat or not
*/

select xmlroot(xmlelement("passengertourlist",
xmlagg(xmlelement("isabletoselectseat", xmlattributes(b.isabletoselectseat as
"abletoselect"),
xmlagg(xmlelement("seatcode",xmlattributes(b.seatcode as "seatcode"),
xmlelement("checkinstatus", b.checkinstatus),xmlelement("bookingcode",
b.bookingcode)))))), version'1.0') as doc from passengertour b group by
b.isabletoselectseat;

/* Query 26) Select all passengers information
*/

SELECT
XMLROOT(XMLELEMENT("Passenger",XMLAGG(XMLELEMENT("PassengerList",XMLFOREST(p.pid AS
"PassengerID", p.pname AS "PassengerName", p.dob AS "dob",
p.phone AS "phone", p.gender AS "Gender", p.salutation AS "Salutation",
p.ishelprequired AS "ishelprequired")))),VERSION '1.0') AS document FROM passenger
p;
```

2.1 Spool file of SQL

```
SQL> @ milestone3-parte2.sql
SQL> set echo on
SQL> set long 32000
SQL> set pagesize 100
SQL>
SQL> /* Question 2) For 12 requests (group #4) formulate SQL statements with Oracle
XML functions returning XML data... */
SQL>
SQL> -- queries (2, 3, 4)
SQL>
SQL> /* Query 2) Airport Employee who works at Toronto Airport selects all cities
that are connected by flights to/from Toronto. */
SQL>
SQL> with aux as (
  2  select f.arrivalairport.airportcode AS AirportCode, f.arrivalairport.city AS
City, f.arrivalairport.country AS Country, f.arrivalairport.name AS Name from
  3  flight f where f.departureairport.city = 'Toronto' UNION
  4  select f.departureairport.airportcode AS AirportCode, f.departureairport.city
AS City, f.departureairport.country AS Country, f.departureairport.name AS
  5  Name from flight f where f.arrivalairport.city = 'Toronto')
  6  select xmlroot(xmlelement("connections", xmlagg(xmlelement("airport",
xmlattributes(a.airportcode as "code"), xmlelement("city", a.city),
  7  xmlelement("country", a.country), xmlelement("name", a.name)))), version
'1.0') as doc from aux a;

DOC
--------------------------------------------------------------------------------
<?xml version="1.0"?>
<connections>
  <airport code="LHR">
    <city>London</city>
    <country>United Kingdom</country>
    <name>London Heathrow Airport</name>
  </airport>
  <airport code="YVR">
    <city>Vancouver</city>
    <country>Canada</country>
    <name>Vancouver International Airport</name>
  </airport>
</connections>


SQL>
SQL> /* Query 3) Airport Employee select all airlines that have a Boeing airplane,
and the model of the Boeing airplane. */
SQL>
SQL> select xmlroot(xmlelement("aircraftslist", xmlagg(xmlelement("aircraft",
xmlattributes(a.aircraftid as "id"), xmlelement("make", a.make ),
  2  xmlelement("airline", a.airlineoid.airline_name), xmlelement("model",
a.model), xmlelement("numberofseats", a.numberofseats), xmlelement("hoursflown",
  3  a.hoursflown)))), version '1.0') as doc from aircraft a where a.make =
'Boeing';

DOC
--------------------------------------------------------------------------------
<?xml version="1.0"?>
<aircraftslist>
  <aircraft id="AB1310">
    <make>Boeing</make>
    <airline>United Airlines</airline>
    <model>B777</model>
    <numberofseats>350</numberofseats>
    <hoursflown>500</hoursflown>
  </aircraft>
  <aircraft id="AB2310">
    <make>Boeing</make>
    <airline>Turkish Airlines</airline>
    <model>B777</model>
    <numberofseats>330</numberofseats>
    <hoursflown>6000</hoursflown>
  </aircraft>
  <aircraft id="AB3310">
    <make>Boeing</make>
    <airline>Air Canada</airline>
    <model>B787</model>
    <numberofseats>250</numberofseats>
    <hoursflown>6700</hoursflown>
  </aircraft>
</aircraftslist>


SQL>
SQL> /* Query 4) Airport Employee selects the boarding time of all the domestic and
international flights departing from canadian airports, and the arrival
SQL> destination (city and country).
SQL> Follow: item C, D (Group BY), E
SQL> */
SQL>
SQL> select xmlroot(xmlelement("flightlist", xmlagg(xmlelement("category",
xmlattributes(f.isinternational as "isinternational"), xmlagg(xmlelement("flight",
  2  xmlattributes(f.flightID as "id", f.flightnumber as "number"),
xmlelement("departure", xmlforest(f.departureairport.city as "city",
  3  f.departureairport.country as "country")),  xmlelement("arrival",
xmlforest(f.arrivalairport.city as "city", f.arrivalairport.country as "country"
)),
  4  xmlelement("boardingtime", f.calculateboardingtime()),
xmlelement("departuretime", f.calculatedelayeddeptime()), xmlelement("duration",
  5  f.flightdurationinminutes), xmlelement("status", f.status),
xmlelement("airline", f.airlineoid.airline_name), xmlelement("aircraftid",
  6  f.aircraftoid.aircraftid) ) )))), version '1.0') as doc from flight f where
```

```
f.departureairport.country = 'Canada' group by f.isinternational;

DOC
--------------------------------------------------------------------------------
<?xml version="1.0"?>
<flightlist>
  <category isinternational="0">
    <flight id="AC1" number="AC1450">
      <departure>
        <city>Vancouver</city>
        <country>Canada</country>
      </departure>
      <arrival>
        <city>Toronto</city>
        <country>Canada</country>
      </arrival>
      <boardingtime>2023-08-21T12:15:00.000000000-07:00</boardingtime>
      <departuretime>2023-08-21T13:00:00.000000000-07:00</departuretime>
      <duration>281</duration>
      <status>arrived</status>
      <airline>Air Canada</airline>
      <aircraftid>AB3310</aircraftid>
    </flight>
  </category>
  <category isinternational="1">
    <flight id="AC2" number="AC3450">
      <departure>
        <city>Toronto</city>
        <country>Canada</country>
      </departure>
      <arrival>
        <city>London</city>
        <country>United Kingdom</country>
      </arrival>
      <boardingtime>2023-11-14T11:30:00.000000000-04:00</boardingtime>
      <departuretime>2023-11-14T13:00:00.000000000-04:00</departuretime>
      <duration>457</duration>
      <status>arrived</status>
      <airline>Air Canada</airline>
      <aircraftid>AB3310</aircraftid>
    </flight>
  </category>
</flightlist>


SQL>
SQL>
SQL> --  queries (8, 9, 10)
SQL>
SQL> /* Query 8) Display the flight number and departure city for flights operated
```

```
by airlines with the airline ID 0538. */
SQL>
SQL> select xmlroot(xmlelement("departures", xmlagg (xmlelement ("flightnm",
xmlattributes(f.flightnumber as "nm"), xmlelement("city",
  2  f.departureAirport.city)))), version '1.0') as doc from flight f where
f.airlineoid.airline_id= '0538';

DOC
--------------------------------------------------------------------------------
<?xml version="1.0"?>
<departures>
  <flightnm nm="AC1450">
    <city>Vancouver</city>
  </flightnm>
  <flightnm nm="AC3450">
    <city>Toronto</city>
  </flightnm>
</departures>


SQL>
SQL> /* Query 9) Show the person ID, first and last names, and time zone for all
employees working at the airport.
SQL> Follow: item C, E
SQL> */
SQL>
SQL> select xmlroot(xmlelement("employees", xmlagg(xmlelement("employee",
xmlattributes(ar.pid as "ID"), xmlforest(ar.pname.firstn as "given_nm",
  2  ar.pname.lastn as "surname", ar.oairportcode.timezone as "timeZone")))),
version '1.0') as doc2 from Airpemployee ar;

DOC2
--------------------------------------------------------------------------------
<?xml version="1.0"?>
<employees>
  <employee ID="12345678">
    <given_nm>Dave</given_nm>
    <surname>Smith</surname>
    <timeZone>Etc/GMT+7</timeZone>
  </employee>
  <employee ID="33345678">
    <given_nm>Many</given_nm>
    <surname>Smith</surname>
    <timeZone>Etc/GMT+7</timeZone>
  </employee>
  <employee ID="12346789">
    <given_nm>Seth</given_nm>
    <surname>Jones</surname>
    <timeZone>Etc/GMT+10</timeZone>
  </employee>
  <employee ID="33245339">
    <given_nm>Lily</given_nm>
    <surname>Taylor</surname>
    <timeZone>Etc/GMT+4</timeZone>
  </employee>
  <employee ID="33245434">
    <given_nm>Riley</given_nm>
    <surname>MacDonald</surname>
    <timeZone>Etc/GMT+10</timeZone>
  </employee>
  <employee ID="33245675">
    <given_nm>Leo</given_nm>
    <surname>De Cat</surname>
    <timeZone>Etc/GMT+3</timeZone>
  </employee>
</employees>


SQL>
SQL> /* Query 10) Organise passengers based on their passport expiry date, and
present their ID and full name, excluding the passenger with ID 2009.
SQL> Follow: item C, D (Group BY), E
SQL> */
SQL>
SQL> select xmlroot(xmlelement("passengerlist", xmlagg( xmlelement("passport_exp",
xmlattributes(p.aoid.expiry_date as "exp"), xmlagg(xmlelement("client",
  2  xmlattributes(p.pid as "ID"), xmlelement("name", p.fullname()))))))), version
'1.0') as doc from passenger p where p.pid<> '2009' group by
  3  p.aoid.expiry_date;

DOC
--------------------------------------------------------------------------------
<?xml version="1.0"?>
<passengerlist>
  <passport_exp exp="2025-12-17">
    <client ID="1005    ">
      <name>Mouralii Ki</name>
    </client>
  </passport_exp>
  <passport_exp exp="2029-02-17">
    <client ID="1009    ">
      <name>Marina Li</name>
    </client>
  </passport_exp>
  <passport_exp exp="2032-12-07">
    <client ID="1008    ">
      <name>Hazel Hu</name>
    </client>
    <client ID="2008    ">
      <name>Suzanne Schroeder</name>
    </client>
    <client ID="2007    ">
      <name>Monique Schroeder</name>
    </client>
  </passport_exp>
  <passport_exp exp="2035-12-17">
    <client ID="1234    ">
      <name>Johnny Depp</name>
    </client>
    <client ID="1007    ">
      <name>Mukta Dey</name>
    </client>
  </passport_exp>
</passengerlist>


SQL>
SQL> -- queries (15, 16, 17)
SQL>
SQL> /* Query 15) Select the checkInStatus of all the passengertour that fly to
Toronto and group them by checkinstatus.
SQL> This meet the requirement of using ROOT + ATTRIBUTES + GROUP BY
SQL> */
SQL>
SQL> SELECT XMLROOT(
XMLELEMENT("result",XMLAGG(XMLELEMENT("PassengerStatus",XMLATTRIBUTES(pt.checkinsta
tus), XMLAGG(XMLELEMENT("PassengerInfo",
  2
XMLELEMENT("name",pt.poid.pname),XMLELEMENT("seat",pt.seatcode),XMLELEMENT("flighti
d",pt.foid.flightid)))))),version '1.0') AS document FROM passengertour
  3  pt WHERE pt.foid.arrivalairport.city = 'Toronto' GROUP BY pt.checkinstatus;

DOCUMENT
--------------------------------------------------------------------------------
<?xml version="1.0"?>
<result>
  <PassengerStatus CHECKINSTATUS="CHECKED">
    <PassengerInfo>
      <name>
        <NAME_P>
          <FIRSTN>Hazel</FIRSTN>
          <LASTN>Hu</LASTN>
        </NAME_P>
      </name>
      <seat>L04</seat>
      <flightid>AC1</flightid>
    </PassengerInfo>
  </PassengerStatus>
  <PassengerStatus CHECKINSTATUS="NOTCHECKED">
    <PassengerInfo>
```

```
      <name>
        <NAME_P>
          <FIRSTN>Mouralii</FIRSTN>
          <LASTN>Ki</LASTN>
        </NAME_P>
      </name>
      <seat>L06</seat>
      <flightid>AC1</flightid>
    </PassengerInfo>
    <PassengerInfo>
      <name>
        <NAME_P>
          <FIRSTN>Mouralii</FIRSTN>
          <LASTN>Ki</LASTN>
        </NAME_P>
      </name>
      <seat>L06</seat>
      <flightid>AC1</flightid>
    </PassengerInfo>
  </PassengerStatus>
</result>


SQL>
SQL> /* Query 16) Select the AirlineEmployees' fullName and gender that run the
Aircraft whose aircraftID is AE3665.
SQL> This meet the requirement of using ROOT + FOREST.
SQL> */
SQL>
SQL> SELECT XMLROOT(XMLELEMENT("AirlineEmployee",XMLFOREST(ae.pname as "name",
ae.gender as "gender", ae.role_al as "role")),version '1.0', standalone yes) AS
  2  document FROM airlEmployee ae, Aircraft ac WHERE ac.AircraftID = 'AE3665' AND
ae.airlineoid.airline_id = ac.airlineoid.airline_id;

DOCUMENT
--------------------------------------------------------------------------------
<?xml version="1.0" standalone="yes"?>
<AirlineEmployee>
  <name>
    <FIRSTN>Gilbert</FIRSTN>
    <LASTN>Cote</LASTN>
  </name>
  <gender>M</gender>
  <role>flight attendant</role>
</AirlineEmployee>


SQL>
SQL> /* Query 17) Select the BoardingGate information at Los Angeles.
SQL> This meet the requirement of using ROOT + AGG + ATTRIBUTE
SQL> */
SQL>
SQL> SELECT
XMLROOT(XMLELEMENT("RESULT",XMLAGG(XMLELEMENT("BoardingGate",XMLATTRIBUTES(bg.bgid)
,XMLELEMENT("GateNumber",bg.gatenumber)))),version
  2  '1.0',standalone yes) AS document FROM BoardingGate bg WHERE bg.aoid.city =
'Los Angeles';

DOCUMENT
--------------------------------------------------------------------------------
<?xml version="1.0" standalone="yes"?>
<RESULT>
  <BoardingGate BGID="LAXA20">
    <GateNumber>A20</GateNumber>
  </BoardingGate>
  <BoardingGate BGID="LAXA15">
    <GateNumber>A15</GateNumber>
  </BoardingGate>
</RESULT>


SQL>
SQL>
SQL> -- queries (22, 23, 24, 25, 26)
SQL>
SQL> /* Query 22) Check flightID, Flight number, Status of the flights where
flights are delayed more than 10 minutes
SQL> */
SQL>
SQL> select XMLROOT(XMLELEMENT("flights", XMLAGG(XMLELEMENT("flights",
XMLFOREST(f.flightid AS "flightID",f.flightNumber AS
  2  "flightNumber",f.scheduleddeparturetime AS
"scheduleddeperaturetime",f.flightdurationinminutes AS
"flightdurationInMinutes",f.STATUS  AS
  3  "status",f.gnumber AS "gnumber")))), VERSION '1.0') AS document FROM flight f
WHERE f.delayinminutes > 10;

DOCUMENT
--------------------------------------------------------------------------------
<?xml version="1.0"?>
<flights>
  <flights>
    <flightID>U1</flightID>
    <flightNumber>UA30</flightNumber>
    <scheduleddeperaturetime>2023-11-29T07:00:00.000000-07:00</scheduleddeperatu
retime>
    <flightdurationInMinutes>317</flightdurationInMinutes>
    <status>delayed</status>
    <gnumber>00002202080924694CBAADEF76E0603F82E14E7A4E091B44FD6667F1D5E0603F82E
14E3C8E</gnumber>
```

```
  </flights>
  <flights>
    <flightID>U2</flightID>
    <flightNumber>UA30</flightNumber>
    <scheduleddeperaturetime>2023-08-24T10:00:00.000000-07:00</scheduleddeperatu
retime>
    <flightdurationInMinutes>317</flightdurationInMinutes>
    <status>arrived</status>
    <gnumber>00002202080924694CBAA7EF76E0603F82E14E7A4E091B44FD6667F1D5E0603F82E
14E3C8E</gnumber>
  </flights>
</flights>


SQL>
SQL> /* Query 23) Get passenger’s first and last name, date of birth, where
passenger’s id is '1005' and group by phone gender and salutation.
SQL> */
SQL>
SQL> Select XMLROOT(XMLElement("PassengerInformation",XMLElement("Document",
XMLATTRIBUTES(p.pid AS
  2
"passportid"),XMLElement("PassengerDetails",XMLAgg(XMLElement("Passenger",XMLFOREST
(p.pname AS "name",p.dob AS "dob")))),XMLElement("passenger",XMLFOREST(
  3  p.phone AS "phone",p.gender AS "gender", p.salutation AS
"salutation")))),VERSION '1.0') FROM passenger p WHERE p.pid = '1005' GROUP BY
p.pid, p.phone,
  4  p.gender, p.salutation;

XMLROOT(XMLELEMENT("PASSENGERINFORMATION",XMLELEMENT("DOCUMENT",XMLATTRIBUTES(P.
--------------------------------------------------------------------------------
<?xml version="1.0"?>
<PassengerInformation>
  <Document passportid="1005    ">
    <PassengerDetails>
      <Passenger>
        <name>
          <FIRSTN>Mouralii</FIRSTN>
          <LASTN>Ki</LASTN>
        </name>
        <dob>2005-01-10</dob>
      </Passenger>
    </PassengerDetails>
    <passenger>
      <phone>649690775</phone>
      <gender>F</gender>
      <salutation>MS</salutation>
    </passenger>
  </Document>
</PassengerInformation>
SQL>
SQL> /* Query 24) Find the check-in status of the passenger where the booking code
is 'T1232'.
SQL> */
SQL>
SQL> select XMLROOT(XMLElement("passengertour", XMLAgg(XMLElement("BookingCode",
XMLAttributes(t.bookingcode AS "BookingCode"), XMLElement("CheckinStatus",
  2  XMLAttributes(t.checkinstatus AS "CheckinStatus"))))), VERSION '1.0') AS
document FROM passengertour t WHERE bookingcode= 'T1232';

DOCUMENT
--------------------------------------------------------------------------------
<?xml version="1.0"?>
<passengertour>
  <BookingCode BookingCode="T1232">
    <CheckinStatus CheckinStatus="CHECKED"/>
  </BookingCode>
</passengertour>


SQL>
SQL> /* Query 25) Select seat code, check-in status , booking code where passengers
able to select seat or not
SQL> */
SQL>
SQL> select xmlroot(xmlelement("passengertourlist",
xmlagg(xmlelement("isabletoselectseat", xmlattributes(b.isabletoselectseat as
"abletoselect"),
  2  xmlagg(xmlelement("seatcode",xmlattributes(b.seatcode as "seatcode"),
xmlelement("checkinstatus", b.checkinstatus),xmlelement("bookingcode",
  3  b.bookingcode)))))), version'1.0') as doc from passengertour b group by
b.isabletoselectseat;

DOC
--------------------------------------------------------------------------------
<?xml version="1.0"?>
<passengertourlist>
  <isabletoselectseat abletoselect="NO">
    <seatcode seatcode="L04">
      <checkinstatus>CHECKED</checkinstatus>
      <bookingcode>T1245</bookingcode>
    </seatcode>
    <seatcode seatcode="L06">
      <checkinstatus>NOTCHECKED</checkinstatus>
      <bookingcode>T8883</bookingcode>
    </seatcode>
    <seatcode seatcode="L06">
      <checkinstatus>NOTCHECKED</checkinstatus>
      <bookingcode>T8886</bookingcode>
    </seatcode>
    <seatcode seatcode="L05">
      <checkinstatus>CHECKED</checkinstatus>
      <bookingcode>T1345</bookingcode>
    </seatcode>
  </isabletoselectseat>
  <isabletoselectseat abletoselect="YES">
    <seatcode seatcode="L01">
      <checkinstatus>CHECKED</checkinstatus>
      <bookingcode>T1231</bookingcode>
    </seatcode>
    <seatcode seatcode="L03">
      <checkinstatus>CHECKED</checkinstatus>
      <bookingcode>T1234</bookingcode>
    </seatcode>
    <seatcode seatcode="L03">
      <checkinstatus>CHECKED</checkinstatus>
      <bookingcode>T1235</bookingcode>
    </seatcode>
    <seatcode seatcode="L03">
      <checkinstatus>CHECKED</checkinstatus>
      <bookingcode>T1232</bookingcode>
    </seatcode>
  </isabletoselectseat>
</passengertourlist>


SQL>
SQL> /* Query 26) Select all passengers information
SQL> */
SQL>
SQL> SELECT
XMLROOT(XMLELEMENT("Passenger",XMLAGG(XMLELEMENT("PassengerList",XMLFOREST(p.pid AS
"PassengerID", p.pname AS "PassengerName", p.dob AS "dob",
  2  p.phone AS "phone", p.gender AS "Gender", p.salutation AS "Salutation",
p.ishelprequired AS "ishelprequired")))),VERSION '1.0') AS document FROM passenger
  3  p;

DOCUMENT
--------------------------------------------------------------------------------
<?xml version="1.0"?>
<Passenger>
  <PassengerList>
    <PassengerID>1234    </PassengerID>
    <PassengerName>
      <FIRSTN>Johnny</FIRSTN>
      <LASTN>Depp</LASTN>
    </PassengerName>
    <dob>1963-06-09</dob>
    <phone>647780775</phone>
    <Gender>M</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </PassengerList>
  <PassengerList>
    <PassengerID>1005    </PassengerID>
    <PassengerName>
      <FIRSTN>Mouralii</FIRSTN>
      <LASTN>Ki</LASTN>
    </PassengerName>
    <dob>2005-01-10</dob>
    <phone>649690775</phone>
    <Gender>F</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </PassengerList>
  <PassengerList>
    <PassengerID>1008    </PassengerID>
    <PassengerName>
      <FIRSTN>Hazel</FIRSTN>
      <LASTN>Hu</LASTN>
    </PassengerName>
    <dob>2005-12-17</dob>
    <phone>649680775</phone>
    <Gender>F</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </PassengerList>
  <PassengerList>
    <PassengerID>1009    </PassengerID>
    <PassengerName>
      <FIRSTN>Marina</FIRSTN>
      <LASTN>Li</LASTN>
    </PassengerName>
    <dob>2005-11-18</dob>
    <phone>649690775</phone>
    <Gender>F</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </PassengerList>
  <PassengerList>
    <PassengerID>1007    </PassengerID>
    <PassengerName>
      <FIRSTN>Mukta</FIRSTN>
      <LASTN>Dey</LASTN>
    </PassengerName>
    <dob>1995-12-17</dob>
    <phone>647680775</phone>
    <Gender>F</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </PassengerList>
  <PassengerList>
    <PassengerID>2007    </PassengerID>
    <PassengerName>
      <FIRSTN>Monique</FIRSTN>
      <LASTN>Schroeder</LASTN>
    </PassengerName>
    <dob>2022-12-31</dob>
    <phone>647999999</phone>
    <Gender>F</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </PassengerList>
  <PassengerList>
    <PassengerID>2008    </PassengerID>
    <PassengerName>
      <FIRSTN>Suzanne</FIRSTN>
      <LASTN>Schroeder</LASTN>
    </PassengerName>
    <dob>2018-12-31</dob>
    <phone>647888888</phone>
    <Gender>F</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </PassengerList>
</Passenger>


SQL>
SQL>
SQL> spool off
```

3. For four other requests (group # 4) create XML documents using Oracle XSU:
a. Formulate SQL queries accessing data from more than one object table;
b. Provide user-defined tag names.

Query 1) Airline Employee who works at Air Canada selects all the company's flights that have landed, with the time of arrival and the aircraft model who operated the flight.

```
OracleXML getXML -user "grp4/grp42here" -conn
"jdbc:oracle:thin:@sit.itec.yorku.ca:1521:studb10g" -rowsettag flightlist
-rowtag flight "select f.flightnumber as no, f.status as situation,
DBMS_XMLGEN.CONVERT(f.landedtime) as timelanding,
f.departureairport.airportcode  as departure, f.arrivalairport.airportcode
as arrival, f.aircraftoid.model as airplane from flight f where f.status =
'arrived' and f.airlineoid.airline_name = 'Air Canada'";
```

```
<?xml version = '1.0'?>
<flightlist>
   <flight num="1">
      <NO>AC1450</NO>
      <SITUATION>arrived</SITUATION>
      <TIMELANDING>21-AUG-23 08.41.00.000000 PM -04:00</TIMELANDING>
      <DEPARTURE>YVR</DEPARTURE>
      <ARRIVAL>YYZ</ARRIVAL>
      <AIRPLANE>B787</AIRPLANE>
   </flight>
   <flight num="2">
      <NO>AC3450</NO>
      <SITUATION>arrived</SITUATION>
      <TIMELANDING>15-NOV-23 01.37.00.000000 AM +01:00</TIMELANDING>
      <DEPARTURE>YYZ</DEPARTURE>
      <ARRIVAL>LHR</ARRIVAL>
      <AIRPLANE>B787</AIRPLANE>
   </flight>
</flightlist>
```

Query 11) Provide a list that includes the names, IDs, phone numbers, and job positions of all airline employees, along with the names of the respective airlines they work for.

```
OracleXML getXML -user "grp4/grp42here" -conn
"jdbc:oracle:thin:@sit.itec.yorku.ca:1521:studb10g" -rowsettag
employee_info -rowtag employee "select al.pid as ID, al.pname.firstn as
first_name, al.pname.lastn as last_name, al.role_al as position,
al.airlineoid.airline_name as company, al.phone as phone_n from
airlemployee al";
```

```
<?xml version = '1.0'?>
<employee_info>
   <employee num="1">
      <ID>43245675</ID>
      <FIRST_NAME>Gilbert</FIRST_NAME>
      <LAST_NAME>Cote</LAST_NAME>
      <POSITION>flight attendant</POSITION>
      <COMPANY>Delta Air Lines</COMPANY>
      <PHONE_N>34-563209</PHONE_N>
   </employee>
   <employee num="2">
      <ID>46545675</ID>
      <FIRST_NAME>Martha</FIRST_NAME>
      <LAST_NAME>Lewis</LAST_NAME>
      <POSITION>flight attendant</POSITION>
      <COMPANY>Turkish Airlines</COMPANY>
      <PHONE_N>34-563222</PHONE_N>
   </employee>
   <employee num="3">
      <ID>46545347</ID>
      <FIRST_NAME>Jym</FIRST_NAME>
      <LAST_NAME>Lewis</LAST_NAME>
      <POSITION>pilot</POSITION>
      <COMPANY>Aeromexico</COMPANY>
      <PHONE_N>34-563278</PHONE_N>
   </employee>
   <employee num="4">
      <ID>46986347</ID>
      <FIRST_NAME>Mary</FIRST_NAME>
      <LAST_NAME>Wilson</LAST_NAME>
      <POSITION>co-pilot</POSITION>
      <COMPANY>Air France-KLM</COMPANY>
      <PHONE_N>34-993278</PHONE_N>
   </employee>
</employee_info>
```

Query 18) Select passportNumber and IssueCountry of passengers who take flights that delay for over 0 minute.

```
OracleXML getXML -user "grp4/grp42here" -conn
"jdbc:oracle:thin:@sit.itec.yorku.ca:1521:studb10g" -rowsettag
delayedpassengers -rowtag passengerinfo "select pt.poid.aoid.passport_no
as passportnumber, pt.poid.aoid.issue_country as issuecountry FROM
passengertour pt WHERE pt.foid.delayinminutes > 0 ";
```

```
<?xml version = '1.0'?>
<delayedpassengers>
   <passengerinfo num="1">
      <PASSPORTNUMBER>A002244</PASSPORTNUMBER>
      <ISSUECOUNTRY>USA</ISSUECOUNTRY>
   </passengerinfo>
   <passengerinfo num="2">
      <PASSPORTNUMBER>A001122</PASSPORTNUMBER>
      <ISSUECOUNTRY>Canada</ISSUECOUNTRY>
   </passengerinfo>
</delayedpassengers>
```

Query 27) Sort the document information ordered by passport number along with expiry date and issue country.

```
OracleXML getXML -user "grp4/grp42here" -conn
"jdbc:oracle:thin:@sit.itec.yorku.ca:1521:studb10g" -rowsetTag
"DocumentInfo"  -rowTag "Documents" "select p.passport_no as
"PassportNumber", p.expiry_date as "ExpiryDate", p.issue_country as
"IssueCountry" from documentinfo p order by p.passport_no";
```

```
<?xml version = '1.0'?>
<DocumentInfo>
   <Documents num="1">
      <PASSPORTNUMBER>A001122</PASSPORTNUMBER>
      <EXPIRYDATE>2025-12-17 00:00:00</EXPIRYDATE>
      <ISSUECOUNTRY>Canada</ISSUECOUNTRY>
   </Documents>
   <Documents num="2">
      <PASSPORTNUMBER>A001166</PASSPORTNUMBER>
      <EXPIRYDATE>2029-02-17 00:00:00</EXPIRYDATE>
      <ISSUECOUNTRY>Japan</ISSUECOUNTRY>
   </Documents>
   <Documents num="3">
      <PASSPORTNUMBER>A002244</PASSPORTNUMBER>
      <EXPIRYDATE>2035-12-17 00:00:00</EXPIRYDATE>
      <ISSUECOUNTRY>USA</ISSUECOUNTRY>
   </Documents>
   <Documents num="4">
      <PASSPORTNUMBER>A002288</PASSPORTNUMBER>
      <EXPIRYDATE>2032-12-07 00:00:00</EXPIRYDATE>
      <ISSUECOUNTRY>Brazil</ISSUECOUNTRY>
   </Documents>
   <Documents num="5">
      <PASSPORTNUMBER>A009998</PASSPORTNUMBER>
      <EXPIRYDATE>2032-12-07 00:00:00</EXPIRYDATE>
      <ISSUECOUNTRY>Germany</ISSUECOUNTRY>
   </Documents>
   <Documents num="6">
      <PASSPORTNUMBER>A009999</PASSPORTNUMBER>
      <EXPIRYDATE>2032-12-07 00:00:00</EXPIRYDATE>
      <ISSUECOUNTRY>Germany</ISSUECOUNTRY>
   </Documents>
</DocumentInfo>
```

4. Store some of your XML files in Oracle XML DB repository using DBMS_XDB PL/SQL package. Choose XML documents which allow you to implement the specifications from bullet 5.

File path: **'/public/itec4220airport/aircraftlist.xml' (Query 3)**

```
<aircraftslist>



  <aircraft id="AB1310">
    <make>Boeing</make>
    <airline>United Airlines</airline>
    <model>B777</model>
    <numberofseats>350</numberofseats>
    <hoursflown>500</hoursflown>
  </aircraft>
  <aircraft id="AB2310">
    <make>Boeing</make>
    <airline>Turkish Airlines</airline>
    <model>B777</model>
    <numberofseats>330</numberofseats>
    <hoursflown>6000</hoursflown>
  </aircraft>
  <aircraft id="AB3310">
    <make>Boeing</make>
    <airline>Air Canada</airline>
    <model>B787</model>
    <numberofseats>250</numberofseats>
    <hoursflown>6700</hoursflown>
  </aircraft>
</aircraftslist>
```

File path: **'/public/itec4220airport/canadianflightlist.xml' (Query 4)**

```
<flightlist>
  <category isinternational="0">
    <flight id="AC1" number="AC1450">
      <departure>
        <city>Vancouver</city>
        <country>Canada</country>
      </departure>
      <arrival>
        <city>Toronto</city>
        <country>Canada</country>
      </arrival>
      <boardingtime>2023-08-21T12:15:00.000000000-07:00</boardingtime>
      <departuretime>2023-08-21T13:00:00.000000000-07:00</departuretime>
      <duration>281</duration>
      <status>arrived</status>
      <airline>Air Canada</airline>
      <aircraftid>AB3310</aircraftid>
    </flight>
  </category>
  <category isinternational="1">
    <flight id="AC2" number="AC3450">
      <departure>
        <city>Toronto</city>
        <country>Canada</country>
      </departure>
      <arrival>
        <city>London</city>
        <country>United Kingdom</country>
      </arrival>
      <boardingtime>2023-11-14T11:30:00.000000000-04:00</boardingtime>
      <departuretime>2023-11-14T13:00:00.000000000-04:00</departuretime>
      <duration>457</duration>
      <status>arrived</status>
      <airline>Air Canada</airline>
      <aircraftid>AB3310</aircraftid>
    </flight>
  </category>
</flightlist>
```

File path: **'/public/itec4220airport/departures_info.xml' (Query 8)**

```
<departures>
  <flightnm nm="AC1450">
    <city>Vancouver</city>
  </flightnm>
  <flightnm nm="AC3450">
    <city>Toronto</city>
  </flightnm>
</departures>
```

File path: **'/public/itec4220airport/client_info.xml' (Query 10)**

```
  <passengerlist>
      <passport_exp exp="2025-12-17">
        <client ID="1005    ">
          <name>Mouralii Ki</name>
      </client>
    </passport_exp>
    <passport_exp exp="2029-02-17">
      <client ID="1009    ">
        <name>Marina Li</name>
      </client>
    </passport_exp>
    <passport_exp exp="2032-12-07">
      <client ID="1008    ">
        <name>Hazel Hu</name>
      </client>
      <client ID="2008    ">
        <name>Suzanne Schroeder</name>
      </client>
      <client ID="2007    ">
        <name>Monique Schroeder</name>
      </client>
    </passport_exp>
    <passport_exp exp="2035-12-17">
      <client ID="1234    ">
        <name>Johnny Depp</name>
       </client>
       <client ID="1007    ">
         <name>Mukta Dey</name>
       </client>
     </passport_exp>
   </passengerlist>
```

File path: **'/public/itec4220airport/tourstatuslist.xml' (Query 15)**

```
<result>
  <PassengerStatus CHECKINSTATUS="CHECKED">
    <PassengerInfo>
      <name>
        <NAME_P>
          <FIRSTN>Hazel</FIRSTN>
          <LASTN>Hu</LASTN>
        </NAME_P>
      </name>
      <seat>L04</seat>
      <flightid>AC1</flightid>
    </PassengerInfo>
  </PassengerStatus>
  <PassengerStatus CHECKINSTATUS="NOTCHECKED">
    <PassengerInfo>
      <name>
        <NAME_P>
          <FIRSTN>Mouralii</FIRSTN>
          <LASTN>Ki</LASTN>
        </NAME_P>
      </name>
      <seat>L06</seat>
      <flightid>AC1</flightid>
    </PassengerInfo>
    <PassengerInfo>
      <name>
        <NAME_P>
          <FIRSTN>Mouralii</FIRSTN>
          <LASTN>Ki</LASTN>
        </NAME_P>
      </name>
      <seat>L06</seat>
      <flightid>AC1</flightid>
    </PassengerInfo>
  </PassengerStatus>
</result>
```

File path: **'/public/itec4220airport/passengerlist.xml' (Query 26)**

```
<PassengerList>
  <Passenger>
    <PassengerID>1234    </PassengerID>
    <PassengerName>
      <FIRSTN>Johnny</FIRSTN>
      <LASTN>Depp</LASTN>
    </PassengerName>
    <dob>1963-06-09</dob>
    <phone>647780775</phone>
    <Gender>M</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </Passenger>
  <Passenger>
    <PassengerID>1005    </PassengerID>
    <PassengerName>
      <FIRSTN>Mouralii</FIRSTN>
      <LASTN>Ki</LASTN>
    </PassengerName>
    <dob>2005-01-10</dob>
    <phone>649690775</phone>
    <Gender>F</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </Passenger>
  <Passenger>
    <PassengerID>1008    </PassengerID>
    <PassengerName>
      <FIRSTN>Hazel</FIRSTN>
      <LASTN>Hu</LASTN>
    </PassengerName>
    <dob>2005-12-17</dob>
    <phone>649680775</phone>
    <Gender>F</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </Passenger>
  <Passenger>
    <PassengerID>1009    </PassengerID>
    <PassengerName>
      <FIRSTN>Marina</FIRSTN>
      <LASTN>Li</LASTN>
    </PassengerName>
    <dob>2005-11-18</dob>
    <phone>649690775</phone>
    <Gender>F</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </Passenger>
  <Passenger>
    <PassengerID>1007    </PassengerID>
    <PassengerName>
      <FIRSTN>Mukta</FIRSTN>
      <LASTN>Dey</LASTN>
    </PassengerName>
    <dob>1995-12-17</dob>
    <phone>647680775</phone>
    <Gender>F</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </Passenger>
  <Passenger>
    <PassengerID>2007    </PassengerID>
    <PassengerName>
      <FIRSTN>Monique</FIRSTN>
      <LASTN>Schroeder</LASTN>
    </PassengerName>
    <dob>2022-12-31</dob>
    <phone>647999999</phone>
    <Gender>F</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </Passenger>
  <Passenger>
    <PassengerID>2008    </PassengerID>
    <PassengerName>
      <FIRSTN>Suzanne</FIRSTN>
      <LASTN>Schroeder</LASTN>
    </PassengerName>
    <dob>2018-12-31</dob>
    <phone>647888888</phone>
    <Gender>F</Gender>
    <Salutation>MS</Salutation>
    <ishelprequired>YES</ishelprequired>
  </Passenger>
</PassengerList>
```

File path: **'/public/itec4220airport/selectseatlist.xml' (Query 25)**

```
<passengertourlist>
  <isabletoselectseat abletoselect="NO">
    <seatcode seatcode="L04">
      <checkinstatus>CHECKED</checkinstatus>
      <bookingcode>T1245</bookingcode>
    </seatcode>
    <seatcode seatcode="L05">
      <checkinstatus>CHECKED</checkinstatus>
      <bookingcode>T1345</bookingcode>
    </seatcode>
  </isabletoselectseat>
  <isabletoselectseat abletoselect="YES">
    <seatcode seatcode="L01">
      <checkinstatus>CHECKED</checkinstatus>
      <bookingcode>T1231</bookingcode>
    </seatcode>
    <seatcode seatcode="L03">
      <checkinstatus>CHECKED</checkinstatus>
      <bookingcode>T1234</bookingcode>
    </seatcode>
    <seatcode seatcode="L03">
      <checkinstatus>CHECKED</checkinstatus>
      <bookingcode>T1235</bookingcode>
    </seatcode>
    <seatcode seatcode="L03">
      <checkinstatus>CHECKED</checkinstatus>
      <bookingcode>T1232</bookingcode>
    </seatcode>
  </isabletoselectseat>
</passengertourlist>
```

5. For remaining 12 requests (group # 4) formulate queries in XQuery language and execute the queries against your files in XML DB repository

```
set echo on
set long 32000
set pagesize 100

/* Question 5) For remaining 12 requests (group # 4) formulate queries in XQuery
language and execute the queries against your files in XML DB repository
*/

--  queries (5, 6, 7)

/* Query 5) Airport Employee select all Boeing aircrafts that have flown more than
5,000 hours
Follows: item C
*/

xquery
let $d:= doc('/public/itec4220airport/aircraftlist.xml')
for $a in $d/aircraftslist/aircraft
where $a/hoursflown > 5000 and $a/make = 'Boeing'
return $a
/

/* Query 6) Airport Employee selects international flights which departed from
Canadian airports after November 1, 2023, 23:59 GMT -5 and already arrived
at the destination (item d/f)
Follows: items D, F
*/

xquery
let $c:= doc('/public/itec4220airport/canadianflightlist.xml')
for $i in $c/flightlist/category
where $i/@isinternational = 1 and $i/flight/departure/country = 'Canada' and
$i/flight/status = 'arrived' and xs:dateTime($i/flight/departuretime) >
xs:dateTime('2023-11-01T23:59:00.000000000-05:00')
return $i/flight/@id
/

/* Query 7) Airport Employee gets the number of seats of each flight departing from
Toronto and operated by a Boeing aircraft.
Follows: items C, D, E, F
*/

xquery
let $c:= doc('/public/itec4220airport/canadianflightlist.xml')
for $f in $c/flightlist/category/flight
let $d:= doc('/public/itec4220airport/aircraftlist.xml')
for $a in $d/aircraftslist/aircraft
where $f/departure/city = 'Toronto' and $f/aircraftid = $a/@id and $a/make =
'Boeing'
return $a/numberofseats/text()
```

```
/

--  queries (12, 13, 14)

/* Query 12) From a list of passengers, present the expiration date and
identification number for the individual named Mouralii Ki, and include the
corresponding name in the display as well.
Follows: item C
*/

xquery
let $p:= doc('/public/itec4220airport/client_info.xml')
for $c in $p/passengerlist/passport_exp
where $c/client/name = 'Mouralii Ki'
return element passenger{attribute name{$c/client/name},
element id{$c/client/@ID}, element expdate{$c/@exp}}
/

/* Query 13) From the departure and Canadian flight records, show the city
associated with the domestic flight departure having the flight number AC1450.
Follows: item D, E, F
*/

xquery
let $d:= doc('/public/itec4220airport/departures_info.xml')
for $h in $d/departures/flightnm
let $f:= doc('/public/itec4220airport/canadianflightlist.xml')
for $c in $f/flightlist/category
where $h/@nm='AC1450' and $c/@isinternational='0'
return $h/city/text()
/

/* Query 14) Display details about passengers whose passport expiration date is on
December 7, 2032
Follows: item D
*/

xquery
let $p:= doc('/public/itec4220airport/client_info.xml')
for $c in $p/passengerlist/passport_exp
where $c/@exp='2032-12-07'
return $c
/

-- queries (19, 20, 21)

/* Query 19) Select passengertours that take a seat with L04 from all the
passengertour.
Follows: item C
*/
```

```
xquery
let $tourlist := doc("/public/itec4220airport/tourstatuslist.xml")
for $tour in $tourlist/result/PassengerStatus/PassengerInfo
where $tour/seat = 'L04'
return $tour
/

/* Query 20) Select the information of the flight with a flightid as AC1
Follows: item D
*/

xquery
let $flightlist:= doc('/public/itec4220airport/canadianflightlist.xml')
for $flight in $flightlist/flightlist/category/flight
where $flight/@id = 'AC1'
return $flight
/

/* Query 21) Select the bookingcode of tours whose flights depart from Canada and
arrive to city London
Follows: items E, F
*/

xquery
let $tourlist := doc("/public/itec4220airport/tourstatuslist.xml")
for $tour in $tourlist/result/PassengerStatus/PassengerInfo
let $flightlist:= doc('/public/itec4220airport/canadianflightlist.xml')
for $flight in $flightlist/flightlist/category/flight
where $flight/@id = $tour/flightid and $flight/arrival/city = 'Toronto' and
$flight/departure/country = 'Canada'
return $tour/seat/text()
/


--  queries (28, 29, 30)

/* Query 28) Get the passenger’s date of birth where passenger’s id =1005 and name
is 'Mouralii Ki'.
Follows: items C, E, F
*/

xquery
let $d:= doc('/public/itec4220airport/passengerlist.xml')
for $g in $d/PassengerList/Passenger
let $p:= doc('/public/itec4220airport/client_info.xml')
for $c in $p/passengerlist/passport_exp
where  $g/PassengerID=1005  and $c/client/name/text()='Mouralii Ki'
return $g/dob/text()
/
```

```
/* Query 29) Retrieves the names and phone numbers of passengers whose name
contains the word "Dey"
Follows: item C
*/

xquery
let $d:=doc('/public/itec4220airport/passengerlist.xml')
for $c in $d/PassengerList/Passenger
where contains($c/PassengerName, 'Dey')
return concat($c/PassengerName, ', ',
$c/phone)
/


/* Query 30) Returns the seatcode and check-in status where passengers bookingcode
is 'T1231'.
Follows: item D
*/

xquery
let $t:=doc('/public/itec4220airport/passengertourlist.xml')
for $c in $t/passengertourlist/passengertour/bookingcode
where $c/@bookingcode = 'T1231' return element
passenger{attribute bookingcode {$c/@bookingcode}, element seatcode
{$c/seatcode/text()}, element checkinstatus {$c/checkinstatus/text()}}
/
```

5.1 Spool file of XQueries

```
SQL> @ milestone3-parte5.sql
SQL> set echo on
SQL> set long 32000
SQL> set pagesize 100
SQL>
SQL> /* Question 5) For remaining 12 requests (group # 4) formulate queries in
XQuery language and execute the queries against your files in XML DB repository
SQL> */
SQL>
SQL> -- queries (5, 6, 7)
SQL>
SQL> /* Query 5) Airport Employee select all Boeing aircrafts that have flown more
than 5,000 hours
SQL> Follows: item C
SQL> */
SQL>
SQL> xquery
  2  let $d:= doc('/public/itec4220airport/aircraftlist.xml')
  3  for $a in $d/aircraftslist/aircraft
  4  where $a/hoursflown > 5000 and $a/make = 'Boeing'
  5  return $a
  6  /

Result Sequence
--------------------------------------------------------------------------------
<aircraft id="AB2310">
  <make>Boeing</make>
  <airline>Turkish Airlines</airline>
  <model>B777</model>
  <numberofseats>330</numberofseats>
  <hoursflown>6000</hoursflown>
</aircraft>

<aircraft id="AB3310">
  <make>Boeing</make>
  <airline>Air Canada</airline>
  <model>B787</model>
  <numberofseats>250</numberofseats>
  <hoursflown>6700</hoursflown>
</aircraft>


SQL>
SQL> /* Query 6) Airport Employee selects international flights which departed from
Canadian airports after November 1, 2023, 23:59 GMT -5 and already arrived
SQL> at the destination (item d/f)
SQL> Follows: items D, F
SQL> */
SQL>
SQL> xquery
```

```
  2  let $c:= doc('/public/itec4220airport/canadianflightlist.xml')
  3  for $i in $c/flightlist/category
  4  where $i/@isinternational = 1 and $i/flight/departure/country = 'Canada' and
$i/flight/status = 'arrived' and xs:dateTime($i/flight/departuretime) >
  5  xs:dateTime('2023-11-01T23:59:00.000000000-05:00')
  6  return $i/flight/@id
  7  /

Result Sequence
--------------------------------------------------------------------------------
AC2

SQL>
SQL> /* Query 7) Airport Employee gets the number of seats of each flight departing
from Toronto and operated by a Boeing aircraft.
SQL> Follows: items C, D, E, F
SQL> */
SQL>
SQL> xquery
  2  let $c:= doc('/public/itec4220airport/canadianflightlist.xml')
  3  for $f in $c/flightlist/category/flight
  4  let $d:= doc('/public/itec4220airport/aircraftlist.xml')
  5  for $a in $d/aircraftslist/aircraft
  6  where $f/departure/city = 'Toronto' and $f/aircraftid = $a/@id and $a/make =
'Boeing'
  7  return $a/numberofseats/text()
  8  /

Result Sequence
--------------------------------------------------------------------------------
250

SQL>
SQL> -- queries (12, 13, 14)
SQL>
SQL> /* Query 12) From a list of passengers, present the expiration date and
identification number for the individual named Mouralii Ki, and include the
SQL> corresponding name in the display as well.
SQL> Follows: item C
SQL> */
SQL>
SQL> xquery
  2  let $p:= doc('/public/itec4220airport/client_info.xml')
  3  for $c in $p/passengerlist/passport_exp
  4  where $c/client/name = 'Mouralii Ki'
  5  return element passenger{attribute name{$c/client/name},
  6  element id{$c/client/@ID}, element expdate{$c/@exp}}
  7  /

Result Sequence
```

```
--------------------------------------------------------------------------------
<passenger name="Mouralii Ki"><id ID="1005    "></id><expdate exp="2025-12-17"><
/expdate></passenger>


SQL>
SQL> /* Query 13) From the departure and Canadian flight records, show the city
associated with the domestic flight departure having the flight number AC1450.
SQL> Follows: item D, E, F
SQL> */
SQL>
SQL> xquery
  2  let $d:= doc('/public/itec4220airport/departures_info.xml')
  3  for $h in $d/departures/flightnm
  4  let $f:= doc('/public/itec4220airport/canadianflightlist.xml')
  5  for $c in $f/flightlist/category
  6  where $h/@nm='AC1450' and $c/@isinternational='0'
  7  return $h/city/text()
  8  /

Result Sequence
--------------------------------------------------------------------------------
Vancouver

SQL>
SQL> /* Query 14) Display details about passengers whose passport expiration date
is on December 7, 2032
SQL> Follows: item D
SQL> */
SQL>
SQL> xquery
  2  let $p:= doc('/public/itec4220airport/client_info.xml')
  3  for $c in $p/passengerlist/passport_exp
  4  where $c/@exp='2032-12-07'
  5  return $c
  6  /

Result Sequence
--------------------------------------------------------------------------------
<passport_exp exp="2032-12-07">
  <client ID="1008    ">
    <name>Hazel Hu</name>
  </client>
  <client ID="2008    ">
    <name>Suzanne Schroeder</name>
  </client>
  <client ID="2007    ">
    <name>Monique Schroeder</name>
  </client>
</passport_exp>
```

```
SQL>
SQL> -- queries (19, 20, 21)
SQL>
SQL> /* Query 19) Select passengertours that take a seat with L04 from all the
passengertour.
SQL> Follows: item C
SQL> */
SQL>
SQL> xquery
  2  let $tourlist := doc("/public/itec4220airport/tourstatuslist.xml")
  3  for $tour in $tourlist/result/PassengerStatus/PassengerInfo
  4  where $tour/seat = 'L04'
  5  return $tour
  6  /

Result Sequence
--------------------------------------------------------------------------------
<PassengerInfo>
  <name>
    <NAME_P>
      <FIRSTN>Hazel</FIRSTN>
      <LASTN>Hu</LASTN>
    </NAME_P>
  </name>
  <seat>L04</seat>
  <flightid>AC1</flightid>
</PassengerInfo>


SQL>
SQL> /* Query 20) Select the information of the flight with a flightid as AC1
SQL> Follows: item D
SQL> */
SQL>
SQL> xquery
  2  let $flightlist:= doc('/public/itec4220airport/canadianflightlist.xml')
  3  for $flight in $flightlist/flightlist/category/flight
  4  where $flight/@id = 'AC1'
  5  return $flight
  6  /

Result Sequence
--------------------------------------------------------------------------------
<flight id="AC1" number="AC1450">
  <departure>
    <city>Vancouver</city>
    <country>Canada</country>
  </departure>
```

```
    <arrival>
      <city>Toronto</city>
      <country>Canada</country>
    </arrival>
    <boardingtime>2023-08-21T12:15:00.000000000-07:00</boardingtime>
    <departuretime>2023-08-21T13:00:00.000000000-07:00</departuretime>
    <duration>281</duration>
    <status>arrived</status>
    <airline>Air Canada</airline>
    <aircraftid>AB3310</aircraftid>
  </flight>


SQL>
SQL> /* Query 21) Select the bookingcode of tours whose flights depart from Canada
and arrive to city London
SQL> Follows: items E, F
SQL> */
SQL>
SQL> xquery
  2  let $tourlist := doc("/public/itec4220airport/tourstatuslist.xml")
  3  for $tour in $tourlist/result/PassengerStatus/PassengerInfo
  4  let $flightlist:= doc('/public/itec4220airport/canadianflightlist.xml')
  5  for $flight in $flightlist/flightlist/category/flight
  6  where $flight/@id = $tour/flightid and $flight/arrival/city = 'Toronto' and
$flight/departure/country = 'Canada'
  7  return $tour/seat/text()
  8  /

Result Sequence
--------------------------------------------------------------------------------
L04
L06
L06

SQL>
SQL>
SQL> -- queries (28, 29, 30)
SQL>
SQL> /* Query 28) Get the passenger’s date of birth where passenger’s id =1005 and
name is 'Mouralii Ki'.
SQL> Follows: items C, E, F
SQL> */
SQL>
SQL> xquery
  2  let $d:= doc('/public/itec4220airport/passengerlist.xml')
  3  for $g in $d/PassengerList/Passenger
  4  let $p:= doc('/public/itec4220airport/client_info.xml')
  5  for $c in $p/passengerlist/passport_exp
  6  where  $g/PassengerID=1005  and $c/client/name/text()='Mouralii Ki'
  7  return $g/dob/text()
  8  /

Result Sequence
--------------------------------------------------------------------------------
2005-01-10

SQL>
SQL> /* Query 29) Retrieves the names and phone numbers of passengers whose name
contains the word "Dey"
SQL> Follows: item C
SQL> */
SQL>
SQL> xquery
  2  let $d:=doc('/public/itec4220airport/passengerlist.xml')
  3  for $c in $d/PassengerList/Passenger
  4  where contains($c/PassengerName, 'Dey')
  5  return concat($c/PassengerName, ', ',
  6  $c/phone)
  7  /

Result Sequence
--------------------------------------------------------------------------------
MuktaDey, 647680775

SQL>
SQL>
SQL> /* Query 30) Returns the seatcode and check-in status where passengers
bookingcode is 'T1231'.
SQL> Follows: item D
SQL> */
SQL>
SQL> xquery
  2  let $t:=doc('/public/itec4220airport/passengertourlist.xml')
  3  for $c in $t/passengertourlist/passengertour/bookingcode
  4  where $c/@bookingcode = 'T1231' return element
  5  passenger{attribute bookingcode {$c/@bookingcode}, element seatcode
{$c/seatcode/text()}, element checkinstatus {$c/checkinstatus/text()}}
  6  /

Result Sequence
--------------------------------------------------------------------------------
<passenger bookingcode="T1231"><seatcode>L01</seatcode><checkinstatus>CHECKED</c
heckinstatus></passenger>


SQL>
SQL> spool off
```