

## 1 – Git

- a. Create a Branch with your name – Devops\_SFCC\_yourname ( Devops\_SFCC\_Rahul)
- b. Create a set of files either .txt or Java and check in the code in the above branch (Push)
- c. Make changes to the file locally
- d. Compare the changes and merge the contents to a new file ( a new version to be checked out)
- e. Create a another branch – DEVOPS\_SFCC\_Release\_yourname
- f. Merge the content of previous branch to newly created branch
- g. Delete the old branch

Note: Don't Operate on Master /Main Branch

2 – Create a sample Maven – Java pipeline in Jenkins and build the Jar from git repository.

## 3 – Maven –

- a. Create a sample java project in eclipse/Intelij.
- b. Convert the java project to maven project
- c. Build Jar from Maven project using maven commands
- d. Refer any maven plugin and perform any one of the maven goals. ( Ex : Jetty server start /stop)

## 4 – RDBMS

Provide Solution for any of the 2 from below questions.

### 4.1

Based on the *employees* table below, select all fields from the *employees* table whose salary is less than or equal to \$52,500 (no sorting is required):

```
CREATE TABLE employees
( employee_number int NOT NULL,
  last_name char(50) NOT NULL,
  first_name char(50) NOT NULL,
```

```
    salary int,  
    dept_id int,  
    CONSTRAINT employees_pk PRIMARY KEY (employee_number)  
);  
  
INSERT INTO employees  
(employee_number, last_name, first_name, salary, dept_id)  
VALUES  
(1001, 'Smith', 'John', 62000, 500);  
  
INSERT INTO employees  
(employee_number, last_name, first_name, salary, dept_id)  
VALUES  
(1002, 'Anderson', 'Jane', 57500, 500);  
  
INSERT INTO employees  
(employee_number, last_name, first_name, salary, dept_id)  
VALUES  
(1003, 'Everest', 'Brad', 71000, 501);  
  
INSERT INTO employees  
(employee_number, last_name, first_name, salary, dept_id)  
VALUES  
(1004, 'Horvath', 'Jack', 42000, 501);
```

## 4.2

Based on the *customers* table and the *orders* table below, select the *customer\_id* and *last\_name* from the *customers* table and select the *order\_date* from the *orders* table where there is a matching *customer\_id* value in both the *customers* and *orders* tables. Order the results by *customer\_id* in descending order.

```
CREATE TABLE customers
( customer_id int NOT NULL,
  last_name char(50) NOT NULL,
  first_name char(50) NOT NULL,
  favorite_website char(50),
  CONSTRAINT customers_pk PRIMARY KEY (customer_id)
);

CREATE TABLE orders
( order_id int NOT NULL,
  customer_id int,
  order_date date,
  CONSTRAINT orders_pk PRIMARY KEY (order_id)
);

INSERT INTO customers
(customer_id, last_name, first_name, favorite_website)
VALUES
(4000, 'Jackson', 'Joe', 'techonthenet.com');

INSERT INTO customers
(customer_id, last_name, first_name, favorite_website)
```

```
VALUES
```

```
(5000, 'Smith', 'Jane', 'digminecraft.com');
```

```
INSERT INTO customers
```

```
(customer_id, last_name, first_name, favorite_website)
```

```
VALUES
```

```
(6000, 'Ferguson', 'Samantha', 'bigactivities.com');
```

```
INSERT INTO customers
```

```
(customer_id, last_name, first_name, favorite_website)
```

```
VALUES
```

```
(7000, 'Reynolds', 'Allen', 'checkyourmath.com');
```

```
INSERT INTO customers
```

```
(customer_id, last_name, first_name, favorite_website)
```

```
VALUES
```

```
(8000, 'Anderson', 'Paige', NULL);
```

```
INSERT INTO customers
```

```
(customer_id, last_name, first_name, favorite_website)
```

```
VALUES
```

```
(9000, 'Johnson', 'Derek', 'techonthenet.com');
```

```
INSERT INTO orders
```

```
(order_id, customer_id, order_date)
```

```
VALUES
```

```
(1,7000, '2016/04/18');
```

```
INSERT INTO orders
(order_id, customer_id, order_date)
VALUES
(2,5000,'2016/04/18');
```

```
INSERT INTO orders
(order_id, customer_id, order_date)
VALUES
(3,8000,'2016/04/19');
```

```
INSERT INTO orders
(order_id, customer_id, order_date)
VALUES
(4,4000,'2016/04/20');
```

```
INSERT INTO orders
(order_id, customer_id, order_date)
VALUES
(5,null,'2016/05/01');
```

#### 4.3

Based on the *suppliers* table below, select the unique *city* values that reside in the *state* of California and order the results in descending order by *city*:

```
CREATE TABLE suppliers
( supplier_id int NOT NULL,
  supplier_name char(50) NOT NULL,
  city char(50),
```

```
state char(25),  
CONSTRAINT suppliers_pk PRIMARY KEY (supplier_id)  
);
```

```
INSERT INTO suppliers  
(supplier_id, supplier_name, city, state)  
VALUES  
(100, 'Microsoft', 'Redmond', 'Washington');
```

```
INSERT INTO suppliers  
(supplier_id, supplier_name, city, state)  
VALUES  
(200, 'Google', 'Mountain View', 'California');
```

```
INSERT INTO suppliers  
(supplier_id, supplier_name, city, state)  
VALUES  
(300, 'Oracle', 'Redwood City', 'California');
```

```
INSERT INTO suppliers  
(supplier_id, supplier_name, city, state)  
VALUES  
(400, 'Kimberly-Clark', 'Irving', 'Texas');
```

```
INSERT INTO suppliers  
(supplier_id, supplier_name, city, state)  
VALUES  
(500, 'Tyson Foods', 'Springdale', 'Arkansas');
```

```
INSERT INTO suppliers
(supplier_id, supplier_name, city, state)
VALUES
(600, 'SC Johnson', 'Racine', 'Wisconsin');
```

```
INSERT INTO suppliers
(supplier_id, supplier_name, city, state)
VALUES
(700, 'Dole Food Company', 'Westlake Village', 'California');
```

```
INSERT INTO suppliers
(supplier_id, supplier_name, city, state)
VALUES
(800, 'Flowers Foods', 'Thomasville', 'Georgia');
```

```
INSERT INTO suppliers
(supplier_id, supplier_name, city, state)
VALUES
(900, 'Electronic Arts', 'Redwood City', 'California');
```

#### 4.4

Based on the *customers* and *orders* table from Practice Exercise #3, select the *customer\_id* and *last\_name* from the *customers* table where there is a record in the *orders* table for that *customer\_id*. Order the results in ascending order by *last\_name* and then descending order by *customer\_id*.

```
CREATE TABLE customers
```

```
( customer_id int NOT NULL,  
  last_name char(50) NOT NULL,  
  first_name char(50) NOT NULL,  
  favorite_website char(50),  
  CONSTRAINT customers_pk PRIMARY KEY (customer_id)  
);
```

CREATE TABLE orders

```
( order_id int NOT NULL,  
  customer_id int,  
  order_date date,  
  CONSTRAINT orders_pk PRIMARY KEY (order_id)  
);
```

using the [SQL EXISTS clause](#)

[ORDER BY clause](#).