**Screenshots of Results for SQL Queries, Stored Procedures and Triggers**

1. **List of the details of all the Male Donors.**

Query: select \* from Donor where DonorGender = 'Male';

Output: A screenshot of a computer

Description automatically generated

1. **Provide the details of ‘O Negative’ Blood group stem cell Donors along with the laboratory details where the cells are stored.**

Query: select Donor.DonorID, concat(DonorFirstName, ' ', DonorLastName) as Donor\_Name, DonorGender, DonorRace, DonorBloodGroup, DonorDOB, DonorAddress, StemCell.CellId, CellType, CellOrigin, CellPrice, laboratory.LabId, LabName, LabRoomNum from Donor inner join stemcell on Donor.DonorID = StemCell.DonorID inner join laboratory stemcell.labid = laboratory.labid where Donor.DonorBloodGroup = 'O-';

Output: Graphical user interface, text, application, email

Description automatically generated

1. **What is the available number of different types of stem cells?**

Query: select CellType, count(cellid) Number\_of\_cells from Stemcell group by CellType;

Output: Graphical user interface, text, application, Word

Description automatically generated

1. **Provide the list of stem cell laboratories along with their in-charge contact details.**

Query: select laboratory.LabID, laboratory.LabName, laboratory.LabRoomNum, laboratory.LabAddress, laboratory.InchargeID,

labincharge.InchFstName, labincharge.InchLstName, labincharge.InchAddress, labincharge.InchPhNum, labincharge.InchGender

from laboratory inner join labincharge on laboratory.InchargeID = labincharge.InchargeID;

Output: A screenshot of a computer

Description automatically generated with medium confidence

1. **List of all the cell types and price of the stem cells along with the donor details whose price is more than or equal to $350.**

Query: select Donor.DonorID, concat(DonorFirstName, ' ', DonorLastName) as Donor\_Name, DonorGender, DonorRace, DonorBloodGroup, DonorDOB, DonorAddress, StemCell.CellId, CellType, CellOrigin, CellPrice from Donor inner join stemcell on Donor.DonorID = StemCell.DonorID inner join laboratory on stemcell.labid = laboratory.labid where CellPrice >= '350';

Output: A screenshot of a computer

Description automatically generated

1. **Provide the number of stem cells collected by each laboratory along with the laboratory name.**

Query: select LabName, count(StemCell.CellId) Number\_of\_Cells from stemcell inner join laboratory on stemcell.labid = laboratory.labid group by laboratory.labname;

Output: Graphical user interface, text, application

Description automatically generated

1. **Generate a list of different types of stem cells with their origin.**

Query: select CellOrigin, CellType, count(StemCell.CellId) as Number\_of\_Cells from stemcell group by CellOrigin, CellType order by CellOrigin;

Output: Graphical user interface, text, application

Description automatically generated

1. **Generate a report that would show the number of invoices generated and total invoice amount in each year.**

Query: select year(invoice.InvoiceDate) as Year, count(invoice.InvoiceId) as Number\_of\_Invoices\_Generated, sum(invoice.InvoiceAmt) as Total\_Invoice\_Amount from invoice inner join appointment on invoice.donorid = appointment.donorid group by year(invoice.InvoiceDate);

Output: Graphical user interface, text, application, Word

Description automatically generated

1. **Create a stored procedure to display the number of stem cells collected by each laboratory along with the laboratory name.**

Code: use stemcelldatabase;

DELIMITER //

CREATE PROCEDURE LabAndCells()

BEGIN

select count(stemcelldatabase.stemcell.CellID) as Number\_of\_Cells, stemcelldatabase.laboratory.LabName

from stemcelldatabase.stemcell join stemcelldatabase.laboratory

on stemcelldatabase.stemcell.LabID = stemcelldatabase.laboratory.LabID

group by stemcelldatabase.laboratory.LabName order by count(stemcelldatabase.stemcell.CellID) DESC;

END //

call LabAndCells();

Output: Graphical user interface, text, application

Description automatically generated

1. **Create after insertion trigger and after deletion trigger for the appointment table.**

Code: use stemcelldatabase;

DROP TABLE IF EXISTS appointment\_audit;

CREATE TABLE appointment\_audit

(

ApmntID INT NOT NULL,

ApmntDate DATE NOT NULL,

DonorID INT NOT NULL,

action\_type VARCHAR(50) NOT NULL,

action\_date DATETIME NOT NULL

)

DELIMITER //

CREATE TRIGGER after\_insertion

AFTER INSERT ON stemcelldatabase.appointment

FOR EACH ROW

BEGIN

INSERT INTO appointment\_audit VALUES

(New.ApmntID, New.ApmntDate,New.DonorID,'INSERTED',now());

END//

DELIMITER //

CREATE TRIGGER after\_deletion

AFTER DELETE ON stemcelldatabase.appointment

FOR EACH ROW

BEGIN

INSERT INTO appointment\_audit VALUES

(OLD.ApmntID, OLD.ApmntDate, OLD.DonorID,'DELETED',now());

END//

insert into stemcelldatabase.appointment values(NULL,'2022-02-01',1010);

delete from stemcelldatabase.appointment where ApmntID = 41;

select distinct \* from appointment\_audit;

Output: Graphical user interface, text, application

Description automatically generated