**עבודה שניה – מסדי נתונים**

**שאלה 1:**

להלן השאילתה של השאלה:

select patients.patient\_id,patient\_name,appointment\_time

from patients join appointment

on appointment.doctor\_id="+doctorID+"

and patients.patient\_id=appointment.patient\_id

and appointment.appointment\_time>now()

בוחן יקר אנא שים לב שהוספנו את קובץ הjava הנדרש.

**שאלה 2:**

ה- stored procedure שנכתב ב- workbench:

CREATE PROCEDURE `updating\_actual\_time` (IN idPatient INT)

BEGIN

declare appointmentId int default null;

select appointment\_id from appointment where patient\_id=idPatient

AND abs(TIMEDIFF(now(),appointment\_time))

in (select min(abs(TIMEDIFF(now(),appointment\_time))) from appointment where patient\_id=idPatient) INTO appointmentId;

INSERT INTO queue (appointment\_id,actual\_time) VALUES (appointmentId,now());

END

ה- stored procedure שעשינו עליו Apply:

USE `hmo`;

DROP procedure IF EXISTS `updating\_actual\_time`;

DELIMITER $$

USE `hmo`$$

CREATE PROCEDURE `updating\_actual\_time` (IN idPatient INT)

BEGIN

declare appointmentId int default null;

select appointment\_id from appointment where patient\_id=idPatient

AND abs(TIMEDIFF(now(),appointment\_time))

in (select min(abs(TIMEDIFF(now(),appointment\_time))) from appointment where patient\_id=idPatient) INTO appointmentId;

INSERT INTO queue (appointment\_id,actual\_time) VALUES (appointmentId,now());

END$$

DELIMITER ;

בוחן יקר אנא שים לב שהוספנו את קובץ הjava הנדרש.

**שאלה 3:**

הView- שנכתב ב- workbench:

CREATE VIEW `the\_ten\_patients\_who\_wait\_the\_longest\_time` AS

SELECT appointment.patient\_id

FROM appointment JOIN queue

WHERE appointment.appointment\_id = queue.appointment\_id

ORDER BY ABS(TIMEDIFF(queue.actual\_time,appointment.appointment\_time)) DESC

LIMIT 10

ה- View שעשינו עליו Apply:

USE `hmo`;

CREATE OR REPLACE VIEW `the\_ten\_patients\_who\_wait\_the\_longest\_time` AS

SELECT appointment.patient\_id

FROM appointment JOIN queue

WHERE appointment.appointment\_id = queue.appointment\_id

ORDER BY ABS(TIMEDIFF(queue.actual\_time,appointment.appointment\_time)) DESC

LIMIT 10;

**שאלה 4:**

**הטריגר של ההוספה:**

ה-Trigger שנכתב ב- workbench:

CREATE DEFINER = CURRENT\_USER TRIGGER `hmo`.`queue\_AFTER\_INSERT` AFTER INSERT ON `queue` FOR EACH ROW

BEGIN

declare appointmentDate DATE;

declare doctorId int default null;

select DATE(appointment\_time) from Appointment where appointment\_id=new.appointment\_id into appointmentDate;

select doctor\_id from Appointment where appointment\_id=new.appointment\_id into doctorId;

update Queue\_Summary set num\_of\_patients=num\_of\_patients+1 where date=appointmentDate and doctor\_id=doctorId;

END

ה-Triggerשעשינו עליו Apply :

DELIMITER $$

USE `hmo`$$

CREATE DEFINER = CURRENT\_USER TRIGGER `hmo`.`queue\_AFTER\_INSERT` AFTER INSERT ON `queue` FOR EACH ROW

BEGIN

declare appointmentDate DATE;

declare doctorId int default null;

select DATE(appointment\_time) from Appointment where appointment\_id=new.appointment\_id into appointmentDate;

select doctor\_id from Appointment where appointment\_id=new.appointment\_id into doctorId;

update Queue\_Summary set num\_of\_patients=num\_of\_patients+1 where date=appointmentDate and doctor\_id=doctorId;

END$$

DELIMITER ;

**הטריגר של ההסרה:**

ה-Trigger שנכתב ב- workbench:

CREATE DEFINER = CURRENT\_USER TRIGGER `hmo`.`queue\_BEFORE\_DELETE` BEFORE DELETE ON `queue` FOR EACH ROW

BEGIN

declare appointmentDate DATE;

declare doctorId int default null;

select DATE(appointment\_time) from Appointment where appointment\_id=old.appointment\_id into appointmentDate;

select doctor\_id from Appointment where appointment\_id=old.appointment\_id and DATE(appointment\_time)=appointmentDate into doctorId;

update Queue\_Summary set num\_of\_patients=num\_of\_patients-1 where Queue\_Summary.date=appointmentDate and doctor\_id=doctorId;

END

ה-Triggerשעשינו עליו Apply :

DELIMITER $$

USE `hmo`$$

CREATE DEFINER = CURRENT\_USER TRIGGER `hmo`.`queue\_BEFORE\_DELETE` BEFORE DELETE ON `queue` FOR EACH ROW

BEGIN

declare appointmentDate DATE;

declare doctorId int default null;

select DATE(appointment\_time) from Appointment where appointment\_id=old.appointment\_id into appointmentDate;

select doctor\_id from Appointment where appointment\_id=old.appointment\_id and DATE(appointment\_time)=appointmentDate into doctorId;

update Queue\_Summary set num\_of\_patients=num\_of\_patients-1 where Queue\_Summary.date=appointmentDate and doctor\_id=doctorId;

END$$

DELIMITER ;

**הערה:**

בוחן יקר אנא שים לב שקראנו לטבלה שלנו "hmo" (=קופת חולים באנגלית) ולכן בשאילתות שאישרנו יש אזכורים למילה זו.

**שאלה 5:**

**א.**

select doctor\_name , salary

from Doctors join Queue\_Summary

where data="2020-04-20" and num\_of\_patients>4;

**ב.** השאילתה מחזירה את כל שמות הרופאים והמשכורת שלהם שעבדו בתאריך ה 20.4.2020 וגם היה להם באותו יום יותר מ-4 חולים.

**שאלה 6:**

**א.** המפתחות הם: {B,F} , {B,E,C} , {B,D,C}

**ב.** רמת הנרמול היא NF-3