***5.2.1 & 5.2.2 SCRIPT SQL-DML***

***TRIGGER***

*“Questo trigger controlla che ad un paziente in un determinato ricovero venga assegnato un farmaco prescrivibile.”*

DELIMITER //

CREATE trigger vincolo\_curainricovero

after insert on Cura\_In\_Ricovero

for each row

begin

if(new.Ricovero in(select ID\_Ricovero

from Ricovero)

and new.Farmaco in(select Codice\_Ean

from Farmaco

where Prescrivibile='0') )

then

signal sqlstate '75000'

set message\_text="Stai assegnando ad un paziente di un ricovero un farmaco non prescrivile";

end if;

END ;//

DELIMITER ;

*“Questo trigger controlla più vincoli:*

*- Dopo aver inserito un assegnazione letto su un paziente, tale letto si trova in una stanza di un reparto dello stesso genere del paziente che è stato ricoverato.*

*- Dopo aver inserito un assegnazione letto su un paziente, se il posto letto è libero allora viene definito lo stato di occupazione per tale letto associato a tale ricovero.*

*- Dopo aver inserito un assegnazione letto su un paziente, il cui ricovero sia già presente con tanto di assegnazione stanza, se il posto letto è occupato allora viene settato a libero mentre il nuovo posto letto viene settato a occupato.”*

DELIMITER //

CREATE trigger letto

after insert on Assegnazione\_Letto

for each row

begin

declare Letto\_Vecchio INT default 0;

select L.Posto\_Letto into Letto\_Vecchio

from Assegnazione\_Letto as L

where L.Ricovero=New.Ricovero and New.Posto\_Letto<>L.Posto\_Letto;

if( new.Posto\_Letto in (select ID\_Posto\_Letto

from Posto\_Letto as PL,Stanza as S, Reparto as REP,Persona as P,Ricovero as R

where PL.ID\_Posto\_Letto=new.Posto\_Letto and PL.ID\_Stanza=S.ID\_Stanza and S.ID\_Reparto=REP.ID\_Reparto

and R.ID\_Ricovero=new.Ricovero and R.Paziente=P.Codice\_Fiscale and REP.Tipo<>P.Genere))

then

signal sqlstate '75000'

set message\_text ="Rispettare i generi di reparto";

end if;

if (new.Posto\_Letto is not null)

then

update Posto\_Letto

set Occupato="1"

where ID\_Posto\_Letto=new.Posto\_Letto;

end if;

if(Letto\_Vecchio<>0)

then

update Assegnazione\_Letto

set Data\_Liberamento=New.Data\_Assegnazione

where Ricovero=New.Ricovero and Posto\_Letto=Letto\_Vecchio;

update Posto\_Letto

set Occupato=0

where ID\_Posto\_Letto=Letto\_Vecchio and Occupato=1;

update Posto\_Letto

set Occupato=1

where ID\_Posto\_Letto=New.Posto\_Letto;

end if;

*“Questo trigger controlla che prima di inserire un’assegnazione letto, venga controllato che il letto che si cerca di assegnare non risulti già occupato.”*

DELIMITER //

CREATE trigger Letto\_Occupato

before insert on Assegnazione\_Letto

for each row

begin

if(new.Posto\_Letto IN(select AL2.Posto\_Letto

from Assegnazione\_Letto as AL2

where new.Posto\_Letto=AL2.Posto\_Letto AND new.Data\_Assegnazione>=AL2.Data\_Assegnazione AND (new.Data\_Assegnazione<=AL2.Data\_Liberamento or AL2.Data\_Liberamento is Null)))

then

signal sqlstate '75000'

set message\_text ="Impossibile assegnare un posto occupato";

end if;

END ;//

DELIMITER ;

“*Questo trigger controlla che un monitoraggio sia assegnato per una malattia effettivamente monitorabile.*”

DELIMITER //

CREATE trigger monitor

before insert on Monitoraggio\_Assegnato

for each row

begin

if(0=( select PE.Monitorabile

from Patologia\_Effettiva as PE

where new.Patologia=PE.ID\_Patologia\_Effettiva))

then

signal sqlstate '75000'

set message\_text = "Malattia non monitorabile";

end if;

END ;//

DELIMITER ;

“*Questo trigger verifica che quando ad un paziente con età minore di 12 viene assegnato un medico, tale medico deve essere un medico con una specializzazione in Pediatria*.”

DELIMITER //

CREATE trigger vincolo\_pediatra

before insert on Afferenza\_Medica

for each row

begin

if(12>=( select year(now())-year(P.Data\_Nascita)

from Persona as P

where P.Codice\_Fiscale=new.Paziente)

and (new.Medico NOT IN ( select SM.Medico\_Specialista

from Specializzazione\_Medico as SM

where SM.Specializzazione="Pediatria")))

then

signal sqlstate '75000'

set message\_text = "Obbligatorio un pediatra";

end if;

END ;//

DELIMITER ;

“*Questo trigger garantisce che il medico redattore del referto appartiene effettivamente alla struttura medica in cui è stato stilato il referto.”*

DELIMITER //

CREATE trigger Referto\_Medico\_Struttura

after insert on Referto

for each row

begin

if(new.Medico\_Redattore not in(select SMS.Medico\_Specialista

from Staff\_Medico\_Struttura as SMS

where new.Struttura\_Medica=SMS.Struttura\_Medica AND new.Data>=SMS.Data\_Inizio\_Rapporto\_Lavorativo AND ((new.Data<=SMS.Data\_Fine\_Rapporto\_Lavorativo) or (SMS.Data\_Fine\_Rapporto\_Lavorativo is Null))))

then

signal sqlstate '75000'

set message\_text = "Il medico redattore deve appartente alla struttura che ha rilasciato il referto";

end if;

END ;//

DELIMITER ;

“*Questo trigger verifica che quando si inserisce una patologia effettiva di un dato paziente, se tale patologia compare come una ricaduta deve verificare che la data diagnosticazione sia minore del periodo di ricaduta della patologia.”*

DELIMITER //

CREATE TRIGGER vincolo\_ricaduta

before insert on Patologia\_Effettiva

for each row

begin

if( new.Ricaduta IN(select ID\_Patologia\_Effettiva

from Patologia\_Effettiva as PE, Patologia as P

where P.Codice=new.Patologia and (new.Paziente<>PE.Paziente or new.Patologia<>PE.Patologia or (day(new.Data\_Diagnosticazione-PE.Data\_Diagnosticazione)>P.Periodo\_Ricaduta))))

then

signal sqlstate '75000'

set message\_text = "Cercato di inserire errata ricaduta di una patologia effettiva";

end if;

END ;//

DELIMITER ;

“*Questo trigger verifica che tra le istanze di Gravidanza non venga inserito un Uomo.”*

DELIMITER //

CREATE trigger vincolo\_gravidanza

before insert on Gravidanza

for each row

begin

if("M"=( select P.Genere

from Persona as P

where new.CF\_Donna=P.Codice\_Fiscale))

then

signal sqlstate '75000'

set message\_text ="Un uomo non può avere una gravidanza";

end if;

END ;//

DELIMITER ;

“*Questo trigger verifica che tra le istanze di Gravidanza\_Interrotta non venga inserita una Gravidanza che invece è stata già Portata a termine.”*

DELIMITER //

CREATE trigger vincolo\_gravidanza\_interrotte

after insert on Gravidanza\_Interrotta

for each row

begin

if(new.CF\_Donna in(select CF\_Donna

from Gravidanza as G

where new.Data\_Presunta\_Di\_Parto=G.Data\_Presunta\_Di\_Parto and G.Data\_Effettiva\_Di\_Parto is not null))

then

signal sqlstate '75000'

set message\_text="Inserita una gravidanza interrotta che è già stata portata a termine";

end if;

END ;//

DELIMITER ;

“*Questo trigger verifica che non venga inserito tra le istanze di Esame\_Sostenuto un Esame, effettuato da un Medico che in quella data non afferisce alla struttura Medica.”*

DELIMITER //

CREATE TRIGGER vincolo\_esame\_sostenuto

before insert on Esame\_Sostenuto

for each row

begin

if(new.Medico\_Specialista not in (select SMS.Medico\_Specialista

from Staff\_Medico\_Struttura as SMS

where new.Struttura\_Medica=SMS.Struttura\_Medica and new.Data>=SMS.Data\_Inizio\_Rapporto\_Lavorativo

and ((new.Data<=SMS.Data\_Fine\_Rapporto\_Lavorativo) or SMS.Data\_Fine\_Rapporto\_Lavorativo is Null)))

then

signal sqlstate '75000'

set message\_text = "Cercato di inserire esame sostenuto errato poichè il medico che ha effettuato l'esame non afferisce alla struttura presso la quale si è svolto l'esame";

end if;

END ;//

DELIMITER ;

“*Questo trigger verifica che se lo stato della ricetta è Scaduta, questa non potrà essere esaurita in una Farmacia oppure se lo stato di Validità della Ricetta è Invalido, questa dovrà necessariamente essere stata esaurita in una Farmacia quindi il campo Farmacia non può essere nullo oppure se lo stato è Invalido e la Farmacia è stata specificata, allora il trigger imposterà in automatico la data di esaurimento della ricetta”*

DELIMITER //

CREATE trigger scadenza\_ricetta

after update on Ricetta

for each row

begin

if(new.Stato\_Validità='Scaduta' and new.Farmacia is not null)

then

signal sqlstate '75000'

set message\_text = "Una ricetta scaduta è stata portata ed esaurita in una farmacia";

end if;

if (new.Stato\_Validità="Invalido" and new.Farmacia is null)

then

signal sqlstate '75000'

set message\_text="Specificare la farmacia";

end if;

if (new.Stato\_Validità="Invalido" and new.Farmacia is not null)

then

update Ricetta

set Data\_Esaurimento=now()

where Codice=new.Codice;

end if;

END ;//

DELIMITER ;

“*Questo trigger verifica che un esame può essere prescritto soltanto mediante una ricetta di tipo Rossa.”*

DELIMITER //

CREATE trigger Ricetta\_Per\_Esame

after insert on Prescrizione\_Esame

for each row

begin

if("Bianca"=(select R.Tipo

from Ricetta as R

where new.Ricetta=R.Codice))

then

signal sqlstate '75000'

set message\_text = "Un esame può essere prescritto solo con ricette rosse";

end if;

END ;//

DELIMITER ;

“*Questo trigger verifica che se viene prescritta una Ricetta Bianca essa non potrà contenere farmaci Prescrivibili, ma soltanto non Prescrivibili, invece se la viene Prescritta una Ricetta Rossa essa non potrà contenere farmaci Non\_Prescrivibili, ma soltanto Prescrivibili.”*

DELIMITER //

CREATE trigger vincolo\_prescrizione\_paziente

before insert on Prescrizione\_Paziente

for each row

begin

if("Bianca"=(select R.Tipo

from Ricetta as R

where R.Codice=new.Ricetta

and 1=(select f.Prescrivibile

from Farmaco as f

where f.Codice\_Ean=new.Farmaco)))

then

signal sqlstate '75000'

set message\_text ="Una pescrizione bianca non può contere farmaci prescrivibili";

end if;

if ("Rossa"=(select R.Tipo

from Ricetta as R

where R.Codice=new.Ricetta

and 0=(select f.Prescrivibile

from Farmaco as f

where f.Codice\_Ean=new.Farmaco)))

then

signal sqlstate '75000'

set message\_text ="Una pescrizione rossa non può contere farmaci non-prescrivibili";

end if;

END ;//

DELIMITER ;

“*Questo trigger verifica che un Paziente che sia disabile oppure che abbia un età minore o uguale a 14 anni deve avere specificato necessariamente un Tutore, inoltre se per questi Pazienti viene Specificato il Tutore esso dev'essere necessariamente Maggiorenne.”*

DELIMITER //

CREATE trigger vincolo\_tutore

after insert on Paziente

for each row

begin

if( new.Disabile=1 or new.Codice\_Fiscale in(select p.Codice\_Fiscale

from Persona as p

where year(now()-p.Data\_Nascita)<=14)

and new.Codice\_Fiscale\_Tutore is null )

then

signal sqlstate '75000'

set message\_text ="Specificare il tutore";

elseif(new.Disabile=1 or new.Codice\_Fiscale in(select Codice\_Fiscale

from Persona as p

where year(now()-p.Data\_Nascita)<=14)

and new.Codice\_Fiscale\_Tutore in(select Codice Fiscale

from Persona as p

where year(now()-p.Data\_Nascita)<18))

then

signal sqlstate '75000'

set message\_text ="Specificare un tutore maggiorenne";

end if;

END ;//

DELIMITER ;

“*Questo trigger verifica che il Medico che effettua un Intervento, afferisca realmente in quella data alla struttura medica presso la quale viene sostenuto l'intervento.”*

DELIMITER //

CREATE trigger medico\_intervento

before insert on Intervento\_Sostenuto

for each row

begin

if(new.Codice\_Fiscale\_Medico not in(select sf.Medico\_Specialista

from Staff\_Medico\_Struttura as sf

where sf.Struttura\_Medica=new.Struttura\_Medica and new.Data>=sf.Data\_Inizio\_Rapporto\_Lavorativo and (new.Data<=sf.Data\_Fine\_Rapporto\_Lavorativo or sf.Data\_Fine\_Rapporto\_Lavorativo is null)))

then

signal sqlstate '75000'

set message\_text= "Medico non afferente a struttura medica dell'intervento";

end if;

END ;//

DELIMITER ;

“*Questo trigger verifica che un paziente non abbia come Medico curante sè stesso, evita cioè l’auto-assegnamento”*

delimiter //

create trigger Auto\_Assegnamento

after insert on Afferenza\_Medica

for each row

begin

if(new.Paziente=new.Medico)then

signal sqlstate '75000'

set message\_text = "Auto assegnamento non consentito";

end if;

end;//

delimiter ;

“*Questo trigger verifica che se una Casa Farmaceutica si trova in uno Stato Estero, non può essere specificato nè l’indirizzo nè il Comune di ubicazione della suddetta Casa Farmaceutica”.*

delimiter //

create trigger Casa\_Farmaceutica\_Estera

after insert on Casa\_Farmaceutica

for each row

begin

if(new.Stato="Italia" and (new.Ubicazione is not null or new.Indirizzo is not null))then

signal sqlstate '75000'

set message\_text = "Non è possibile specificare un Comune o un Indirizzo, in quanto il sistema riguarda solo Comuni Italiani";

end if;

end;//

delimiter ;

***PROCESSI SCHEDULATI***

*“Il processo schedulato seguente controlla i vincoli di ogni afferenza medica temporanea. In particolare se manca una settimana esatta alla scadenza, il seguente processo setta a fine la data dell'afferenza. Questo processo è la causa scatenante di un trigger il quale prepara una “comunicazione” al Paziente e al Medico interessato, comunicando loro la fine del periodo temporaneo di afferenza.”*

delimiter //

create event Termina\_Afferenza\_Medica\_Temporanea

on schedule every 1 day

do

begin

DECLARE finito int default 0;

DECLARE P VARCHAR(16);

DECLARE cur1 CURSOR FOR

SELECT Persona

FROM Afferenza\_Medica\_Temporanea as amt

WHERE day(date(now())-amt.Data\_Inizio)=358 and amt.Data\_Fine is null

DECLARE CONTINUE HANDLER FOR SQLSTATE '02000' SET finito = 1;

open cur1;

ciclo: WHILE NOT finito DO

fetch cur1 into P;

update Afferenza\_Medica as am

set Data\_a=date(now)+7

where am.Persona=P;

END WHILE ciclo;

end //

delimiter ;

*“Questo processo schedulato viene innescato ogni giorno: per i monitoraggi assegnati per malattie monitorabili, prepara una comunicazione al paziente possessore della patologia monitorabile, così da ricordargli di effettuare il monitoraggio.”*

delimiter //

create event Notifiche\_Push\_Patologia\_Monitorabili

ON SCHEDULE EVERY 1 DAY

do

begin

DECLARE email VARCHAR(16);

Declare finito int default 0;

declare fine int default 0;

Declare NomePatologia VARCHAR(50);

DECLARE testo varchar(80);

Declare cont int;

DECLARE cur1 CURSOR FOR SELECT p.email,pt.Nome

FROM Monitoraggio\_Assegnato as m,Patologia\_Effettiva as pe,Persona as p,Patologia as pt

WHERE m.Patologia=pe.ID\_Patologia\_Effettiva and p.Codice\_Fiscale=pe.Paziente and pt.Codice=pe.Patologia;

DECLARE CONTINUE HANDLER FOR SQLSTATE '02000'

SET finito = 1;

select count(\*) into fine

from Monitoraggio\_Assegnato;

OPEN cur1;

FETCH cur1 INTO email,NomePatologia;

ciclo: WHILE NOT finito DO

if(cont<fine)

then

set cont=cont+1;

set testo="Ricorda di monitorare il/la"+NomePatologia;

insert into Comunicazione values("SistemaSanitarioNazione@ssn.gov",email,testo,"Da Inviare",date(now()));

FETCH cur1 INTO email,NomePatologia;

else

leave ciclo;

end if;

END WHILE ciclo;

end //

delimiter ;

*“Questo processo schedulato ha il compito di controllare la scadenza della ricette rosse. In particolare se una ricetta di tipo rossa dopo 180 giorni, non è ancora stata utilizzata, essa viene considerata scaduta.”*

delimiter //

create event Scadenza\_Ricetta

ON SCHEDULE EVERY 1 DAY

do

begin

DECLARE cod int;

Declare da date;

Declare finito int default 0;

Declare cont int default 0;

Declare fine int;

DECLARE cur1 CURSOR FOR

SELECT r.Codice,r.Data\_Emissione

FROM Ricetta as r;

DECLARE CONTINUE HANDLER FOR SQLSTATE '02000'

SET finito = 1;

Select count(\*) into fine

From Ricetta;

OPEN cur1;

FETCH cur1 INTO cod,da;

ciclo: WHILE NOT finito DO

if(cont<fine)

then

set cont=cont+1;

if(day(now())-day(da)>180)

then

update Ricetta as r

set r.Stato\_Validita="Scaduta"

where r.Codice=cod;

end if;

FETCH cur1 INTO cod,da;

else

leave ciclo;

end if;

END WHILE ciclo;

end //

delimiter ;