ALTER TABLE

RENAME nomecostraint TO nomenuovo

ALTER TABLE Coda

ADD CONSTRAINT Coda\_pk PRIMARY KEY (TipoCoda)

CONSTRAINT AeroportiDiversi CHECK (AeroportoArrivo <> AeoportoPartenza)

TIMESTAMP tiene sia data che ora

SYSDATE = CURRENTTIMESTAMP (perché utilizziamo il timestamp)

**ENUMERAZIONE**

CREATE TABLE Coda(

TipoCoda VARCHAR(40),

CONSTRAINT DominioCoda CHECK (TipoCoda = ‘Famiglia’ OR TipoCoda = ‘Business’ OR TipoCoda = ‘Diversamente abili’ OR TipoCoda = ‘Priority’ OR TipoCoda = ‘Economy’ OR TipoCoda = ‘Prima Classe’))

**CREAZIONE TRIGGER CON PROCEDURA**

create trigger datacorretta before insert

on volo

for each row

execute function validificadata()

CREATE function validificadata() RETURNS trigger AS $BODY$

begin

if (NEW.datavolo < current\_timestamp) then

raise exception 'Non è possibile inserire un volo precedente ad ora';

end if;

end

$BODY$ LANGUAGE 'plpgsql';

REGEX EXPRESSION

Mannaggia al cazzo

alter table aeroporto

add constraint codiceaeroporto check (codaeroporto::text ~ '^[a-z]{1}-[0-9]{1}$') ***la stringa regex va messa tra parentesi!!!!!!***

Aggiunta regex per quasi tutti i codice delle tabelle

alter table aeroporto

add constraint CodiceCorretto check (codaeroporto like '/^[b-df-hj-np-tv-z]{1,3}[0-9999]$/')

alter table volo

add constraint codicevolo check (codvolo like '/^[[b-df-hj-np-tv-z]]{1,3}-[[b-df-hj-np-tv-z]]{1,3}-[0-9999]$/')

alter table tratta

add constraint codicetratta check (codtratta like '/^[[b-df-hj-np-tv-z]]{1,3}-[[b-df-hj-np-tv-z]]{1,3}$/')

**try** {

conn = DriverManager.*getConnection*("jdbc:postgresql://localhost:5432/Gestione Scalo Aeroportuale", "postgres", "progettooobd");

Statement st = conn.createStatement();

ResultSet rs = st.executeQuery("Select \* from Aeroporto Where CodAeroporto = '"+ Codice+ "'");

**while**(rs.next()) {

System.***out***.println("Codice aereo: " + rs.getString("CodAeroporto") + " "+ "Nome Aeroporto: " + rs.getString("NomeAeroporto")+ " " + "Città: " + rs.getString("Città"));

}

} **catch** (SQLException e) {

// **TODO** Auto-generated catch block

e.printStackTrace();

}

Date data = (Date) DateSpn.getValue();

SimpleDateFormat format = new SimpleDateFormat("YYYY:mm:dd HH:mm:ss");

String time = format.format(data);

System.***out***.println(time);

sum(extract(hour from(tempomax-datainizio))) as tempostimato, sum(extract(hour from(datafine - datainizio)))as tempoeffettivo