Diagram (UML Class Diagram)

Città
- nome : Stringa {id}

Regione
- nome : Stringa {id}

Nazione
- nome : Stringa {...}

Direttore
- nome : Stringa
- cognome : Stringa
- cf : CodiceFiscale {id}
- anni_servizio : IntGEZ
- data_nascita : Data

Dipartimento
- nome : Stringa {id}
- indirizzo : Indirizzo

StatoOrdine
- Nome : Stringa {id}

Fornitore
- ragione_sociale : Stringa
- partita_iva : PartitaIVA {id}
- indirizzo : Indirizzo
- telefono : Telefono
- email : Email

Ordine
- data_stipula : Data
- imponibile : RealGE...
- aliquota : Real0_1
- descrizione : Stringa

Relationships / associations:
- Città 0..* — reg_cit — 1..1 {id} Regione
- Regione 0..* — reg_naz — 1..1 {id} Nazione
- Città 1..1 — cit_nasc — 0..* Direttore
- Città 1..1 — dip_cit — 0..* Dipartimento
- Direttore 1..1 — dip_dir — 0..* Dipartimento
- Città 1..1 — cit_forn — 0..* Fornitore
- StatoOrdine 1..1 — stat_ord — 0..* Ordine
- Dipartimento 1..1 — dip_ord — 0..* Ordine
- Fornitore 1..1 — forn_ord — 0..* Ordine

Tipi di dato non standard

Tipo Indirizzo; tipo record con i seguenti campi:
- via: Stringa
- civico: Stringa
- CAP: Stringa di 5 caratteri numerici

Tipo CodiceFiscale:
Stringa di 16 caratteri alfanumerici secondo standard

Tipo PartitaIVA:
Stringa di 11 caratteri numerici secondo standard

Tipo Telefono
Stringa secondo standard

Tipo Email
Stringa secondo standard

Tipo StatoOrdine: tipo enumerativo:
{in preparazione, inviato, da saldare,saldato}

```
create domain Stringa as varchar (100);

create domain CodiceFiscale as char (16)
     check (value ~ '^[A-Z0-9]{16}');

create domain PartitaIVA as char(11)
     check (value ~ '^[0-9]{11}$');

create type Indirizzo as (
     via varchar(50),
     numero_civico varchar(10),
     cap Cap
);

create domain Telefono as varchar (15);

create domain Email as varchar (255)
     check (value like '%@%%');

create type StatoOrdine as enum ('In
preparazione', 'Inviato', 'Da saldare',
'Saldato');

create domain Cap as char(5)
     check ( value ~'^\d{5}$');

create domain IntGEZ as Integer
     check (values>=0);

create domain RealGez as real
     check ( value>0);

create domain Real0_1 as real
     check (value>=0 and value<=1);
```