

SOLUZIONE ESAME DI STATO 2015/2016

Indirizzo: ITSI - AMMINISTRAZIONE, FINANZA E MARKETING ARTICOLAZIONE “SISTEMI INFORMATIVI AZIENDALI”

©Giselda De Vita 2016

La traccia di esame, pubblicata dal Ministero, si trova al seguente link: [esame 2016](#)

1. identifichi le principali aree del sistema informativo della compagnia City2City e le soluzioni tecnologiche necessarie alla sua implementazione;

La traccia chiede di progettare un Sistema Informativo per la società City2City.

Leggendo la traccia, la società City2City:

1. *“offre collegamenti di andata e di ritorno che partono dalla propria sede operativa, situata in una città di medie dimensioni del centro Italia, e sono diretti ad alcune città europee, raggiungibili in 24 ore di viaggio”*
2. *“Si è quindi dotata di un parco di autobus GT (“Gran Turismo”)”*
3. *“La manutenzione degli autobus GT è affidata ad una serie di ditte esterne”*
4. *“il personale viaggiante sia costituito da un autista principale ed un secondo*
5. *autista che si alternano alla guida, oltre che da un assistente di viaggio”*
6. *“fidelizzare i clienti e prevede quindi sistemi di registrazione degli utenti, con tessera virtuale a punti e indagini di mercato di customer satisfaction. Infine, essa intende offrire sistemi di prenotazione on-line.”*

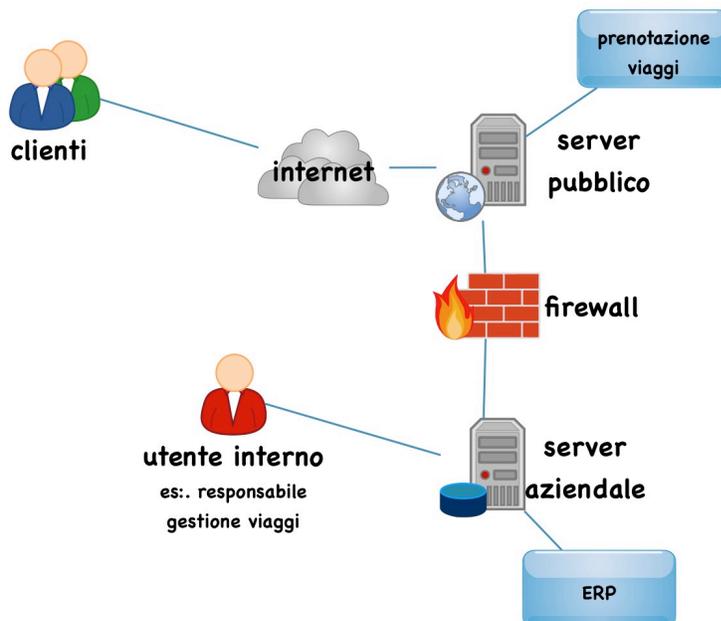
Dalla traccia sono stati selezionati i punti salienti per poter agevolmente procedere alla risposta del primo quesito, ovvero come organizzare il Sistema Informativo.

Le parti centriche del Sistema Informativo verranno riportate nell'ordine in cui sono apparse nella traccia per facilità di comprensione:

1. Essendo la City2City una società di servizi, la parte centrale del Sistema Informativo è proprio la parte che gestirà il **servizio di trasporto**, che verrà chiesto nello specifico nel secondo punto.
2. Logistica e archiviazione dei beni (bus).
3. Gestione dei fornitori (ditte esterne per la manutenzione dei bus) e di conseguenza contabilità (fatture per fornitori esterni).
4. Gestione del personale (autisti, assistenti di viaggio, responsabili cura sito e prenotazioni viaggi e ovviamente la parte di direzione operativa e strategica).
5. Customer care, ovvero tutta la parte CRM (Customer Relationship Management) dedicata agli utenti finali per la fidelizzazione, scheda punti e profilazione.

Tutto il sistema informativo, per la pianificazione del servizio offerto specifico dell'azienda City2City, sarà realizzato con un'architettura web, in modo da poter essere utilizzato online sia per le prenotazioni dei clienti che dagli operatori per fissare e definire i viaggi, gli autisti coinvolti e le tratte interessate. Per le parti comuni ad un'azienda (personale, acquisti, marketing, di servizi potrà essere affidato ad un ERP (Enterprise Resource Planning) di tipo opensource (tipo odoo openerp).

Per disegnare le parti principali del sistema informativo, ricordiamo la differenza tra utenti interni ed esterni all'azienda.



Per quanto riguarda l'attività di trasporto (punto 1 del Sistema informativo che verrà richiesto nel punto 2 di dettagliare) per il software si ipotizza di utilizzare :

- MySql come Database
- Apache come Web Server
- Php come linguaggio lato Server
- HTML, CSS, Java Script per implementare le pagine Web.

2. Concentrandosi poi sulla porzione del sistema informativo che gestisce l'attività di trasporto, schema concettuale della relativa base di dati

Dalla traccia, selezioniamo in giallo le parole che probabilmente diventeranno le entità nel nostro modello concettuale e sottolineiamo i probabili attributi delle entità:

- *i collegamenti offerti, registrando per ciascuno la città collegata, il giorno della settimana, l'orario previsto di partenza e di arrivo, se il collegamento è di andata o di ritorno*
- *il personale viaggiante, del quale oltre ai dati anagrafici interessa il ruolo ricoperto come sopra descritto*
- *il parco di autobus GT, con i dati identificativi di ciascun mezzo, le relative caratteristiche e i dati necessari a gestirne la manutenzione*

- i **viaggi** effettuati, registrando per ciascuno la data in cui è stato svolto, l'orario effettivo di partenza e di arrivo, il numero dei passeggeri effettivi, l'autobus GT utilizzato e il personale viaggiante impiegato;

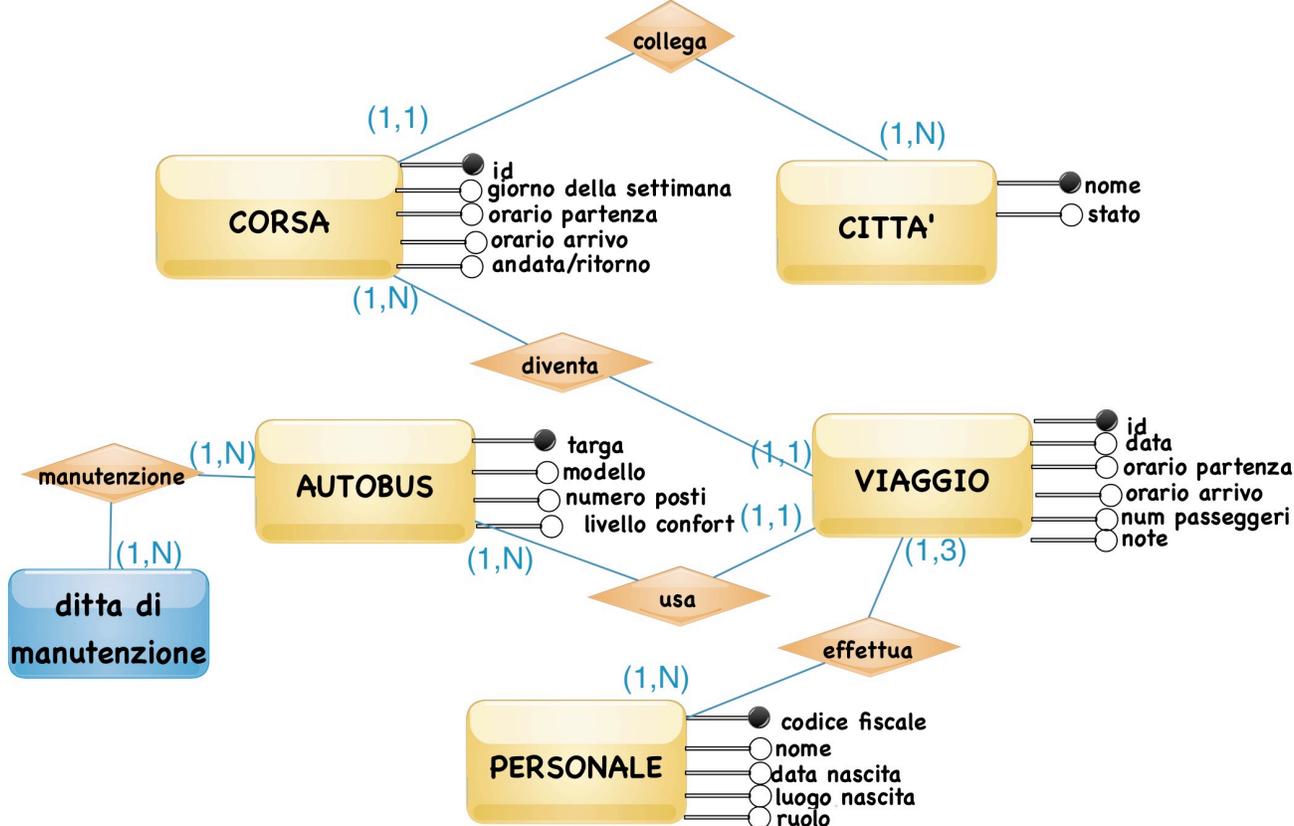
Il modello concettuale richiesto appare molto chiaro sia nelle entità che nelle relazioni e viene anche chiaramente indicato di non confondere i VIAGGI effettuati con i COLLEGAMENTI previsti, che per evitare ambiguità, chiameremo CORSA l'entità che rappresenta il generico collegamento in un determinato giorno della settimana.

Per la normalizzazione della base dati e soprattutto, leggendo le query che poi verranno richieste, l'entità CITTA' deve esistere e non essere un mero attributo dei viaggi o dei collegamenti, credo che questo possa essere l'unico errore in cui gli studenti potevano cadere, per il resto ci sono pochi dubbi anche sulle relazioni. Utilizzando lo stesso ragionamento potrebbe anche essere inserita l'entità NAZIONI.

Un altro ragionevole dubbio poteva essere rappresentato dalla città di partenza, ma la traccia ci riporta una notevole semplificazione specificando che: *“offrire collegamenti di andata e di ritorno che partono dalla propria sede operativa, situata in una città di medie dimensioni del centro Italia”*

Quindi la città di partenza è sempre la stessa e non andrà indicata nel Sistema informativo e di conseguenza nei viaggi e nei collegamenti.

Il modello concettuale che ne consegue è il seguente:



Le entità sono: CORSA(collegamento), CITTA', VIAGGIO, PERSONALE, AUTOBUS, DITTA DI MANUTENZIONE (potrebbe anche essere omessa inserendo qualche dato sulla manutenzione nell'entità autobus).

Verrebbe naturale pensare alle entità CLIENTE e PRENOTAZIONE ma nella seconda parte della traccia viene espressamente richiesto di ampliare il nostro modello concettuale, per cui rimandiamo l’inserimento di queste due entità alla risoluzione del secondo quesito del compito.

Le relazioni sono:

CORSA – CITTA’: uno a molti, un singolo collegamento è verso una singola città, una città è meta di diversi collegamenti.

CORSA - VIAGGIO: uno a molti, la singola corsa diventerà uno o più viaggi, a seconda della data, un viaggio rappresenta una sola corsa in un determinato giorno.

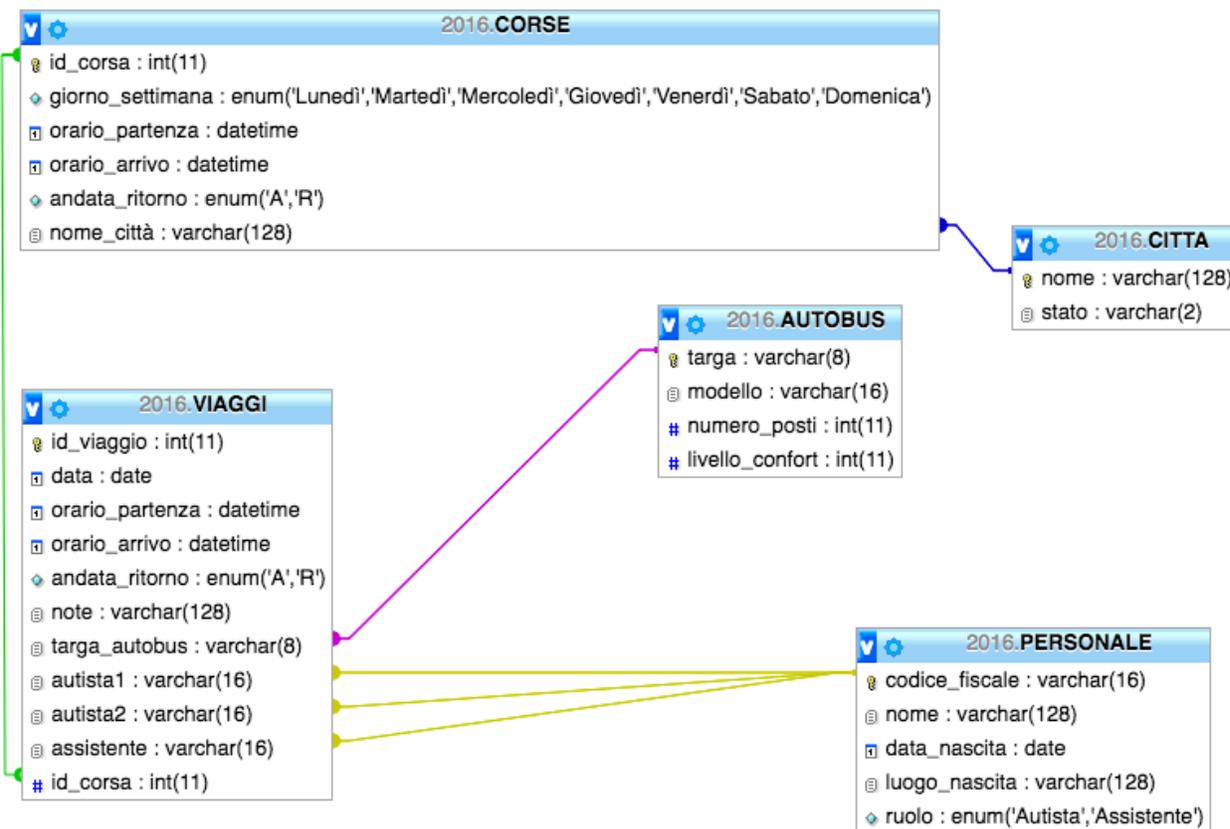
VIAGGIO-PERSONALE: 3 a molti o meglio molti a molti, ma nello specifico son proprio 3 addetti per ogni viaggio per cui ne terremo conto nel passaggio al modello logico.

VIAGGIO - AUTOBUS: uno a molti, un singolo viaggio viene effettuato su un singolo autobus, ogni autobus può effettuare più viaggi.

AUTOBUS – DITTA: molti a molti, un autobus viene mantenuto da una o più ditte mentre una ditta mantiene più autobus.

3. derivi il corrispondente schema logico relazionale;

Dal modello concettuale descritto deriviamo il modello logico:



La decisione di creare la tabella viaggi con 3 foreign key, al posto di avere una altra tabella che leghi viaggi al personale, è perché non ci saranno più di tre persone coinvolte per viaggio. Conviene dal punto di vista delle query e degli inserimenti ed anche inserire la chiave primaria della città con il suo nome conviene per le query che vengono richieste nel passo successivo.

DDL MySql per la creazione del database e delle tabelle

```
-----  
-- Table structure for table 'AUTOBUS'  
  
CREATE TABLE IF NOT EXISTS AUTOBUS (  
  targa varchar(8) NOT NULL,  
  modello varchar(16) NOT NULL,  
  numero_posti int(11) NOT NULL,  
  livello_confort int(11) NOT NULL,  
  PRIMARY KEY (targa)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
  
-----  
-- Table structure for table 'CITTA'  
  
CREATE TABLE IF NOT EXISTS CITTA (  
  nome varchar(128) NOT NULL,  
  stato varchar(2) NOT NULL,  
  PRIMARY KEY (nome)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1;  
  
-----  
-- Table structure for table 'CORSE'  
  
CREATE TABLE IF NOT EXISTS CORSE (  
  id_corsa int(11) NOT NULL AUTO_INCREMENT,  
  giorno_settimana  
enum('Lunedì', 'Martedì', 'Mercoledì', 'Giovedì', 'Venerdì', 'Sabato', 'Domenica') NOT  
NULL,  
  orario_partenza datetime NOT NULL,  
  orario_arrivo datetime NOT NULL,  
  andata_ritorno enum('A', 'R') NOT NULL,  
  nome_città varchar(128) NOT NULL,  
  PRIMARY KEY (id_corsa),  
  KEY nome_città (nome_città),  
  KEY nome_città_2 (nome_città)  
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=1 ;  
  
--  
-- RELATIONS FOR TABLE CORSE:  
--   id_corsa  
--     CORSE -> id_corsa  
--   nome_città  
--     CITTA -> nome  
  
-----  
-- Table structure for table 'PERSONALE'
```

```

CREATE TABLE IF NOT EXISTS PERSONALE (
  codice_fiscale varchar(16) NOT NULL,
  nome varchar(128) NOT NULL,
  data_nascita date NOT NULL,
  luogo_nascita varchar(128) NOT NULL,
  ruolo enum('Autista','Assistente') NOT NULL,
  PRIMARY KEY (codice_fiscale)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

-----
-- Table structure for table 'VIAGGI'

```

```

CREATE TABLE IF NOT EXISTS VIAGGI (
  id_viaggio int(11) NOT NULL AUTO_INCREMENT,
  `data` date NOT NULL,
  orario_partenza datetime NOT NULL,
  orario_arrivo datetime NOT NULL,
  andata_ritorno enum('A','R') NOT NULL,
  note varchar(128) NOT NULL,
  targa_autobus varchar(8) NOT NULL,
  autistal varchar(16) NOT NULL,
  autista2 varchar(16) NOT NULL,
  assistente varchar(16) NOT NULL,
  id_corsa int(11) NOT NULL,
  PRIMARY KEY (id_viaggio),
  KEY autistal (autistal),
  KEY autista2 (autista2),
  KEY assistente (assistente),
  KEY autistal_2 (autistal),
  KEY targa_autobus (targa_autobus),
  KEY nome_città (note),
  KEY id_corsa (id_corsa)
) ENGINE=InnoDB DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

```

```

--
-- RELATIONS FOR TABLE VIAGGI:
--   assistente
--       PERSONALE -> codice_fiscale
--   autistal
--       PERSONALE -> codice_fiscale
--   autista2
--       PERSONALE -> codice_fiscale
--   id_corsa
--       CORSE -> id_corsa
--   targa_autobus
--       AUTOBUS -> targa
--

```

```

-- Constraints for table `CORSE`
ALTER TABLE `CORSE`
  ADD CONSTRAINT corse_ibfk_1 FOREIGN KEY (nome_città) REFERENCES CITTA (nome);

```

```

-- Constraints for table `VIAGGI`
ALTER TABLE `VIAGGI`
  ADD CONSTRAINT viaggi_ibfk_1 FOREIGN KEY (targa_autobus) REFERENCES AUTOBUS
(targa),
  ADD CONSTRAINT viaggi_ibfk_2 FOREIGN KEY (autistal) REFERENCES PERSONALE
(codice_fiscale),
  ADD CONSTRAINT viaggi_ibfk_3 FOREIGN KEY (autista2) REFERENCES PERSONALE
(codice_fiscale),

```

```
ADD CONSTRAINT viaggi_ibfk_4 FOREIGN KEY (assistente) REFERENCES PERSONALE
(codice_fiscale);
```

4. sviluppi in linguaggio SQL le query per ottenere le seguenti informazioni:

a) *elenco dei viaggi di andata svolti verso una determinata città, in un intervallo di date fornite in input;*

```
SELECT *
FROM viaggi, corse
WHERE viaggi.id_corsa = corse.id_corsa
AND nome_città = 'PARIGI'
AND data
BETWEEN '01/01/2016'
AND '10/10/2016'
AND corse.andata_ritorno = 'A'
```

b) *per ciascuna città collegata, calcolare la media del tempo di percorrenza dei viaggi di andata effettuati nel mese di agosto dell'anno corrente.*

```
SELECT nome_città, AVG( TIMESTAMPDIFF(
MINUTE, viaggi.orario_partenza, viaggi.orario_arrivo ) ) AS media_percorrenza
FROM viaggi, corse
WHERE viaggi.id_corsa = corse.id_corsa
AND MONTH( data ) = 8
AND corse.andata_ritorno = 'A'
GROUP BY nome_città
```

SECONDA PARTE DEL COMPITO

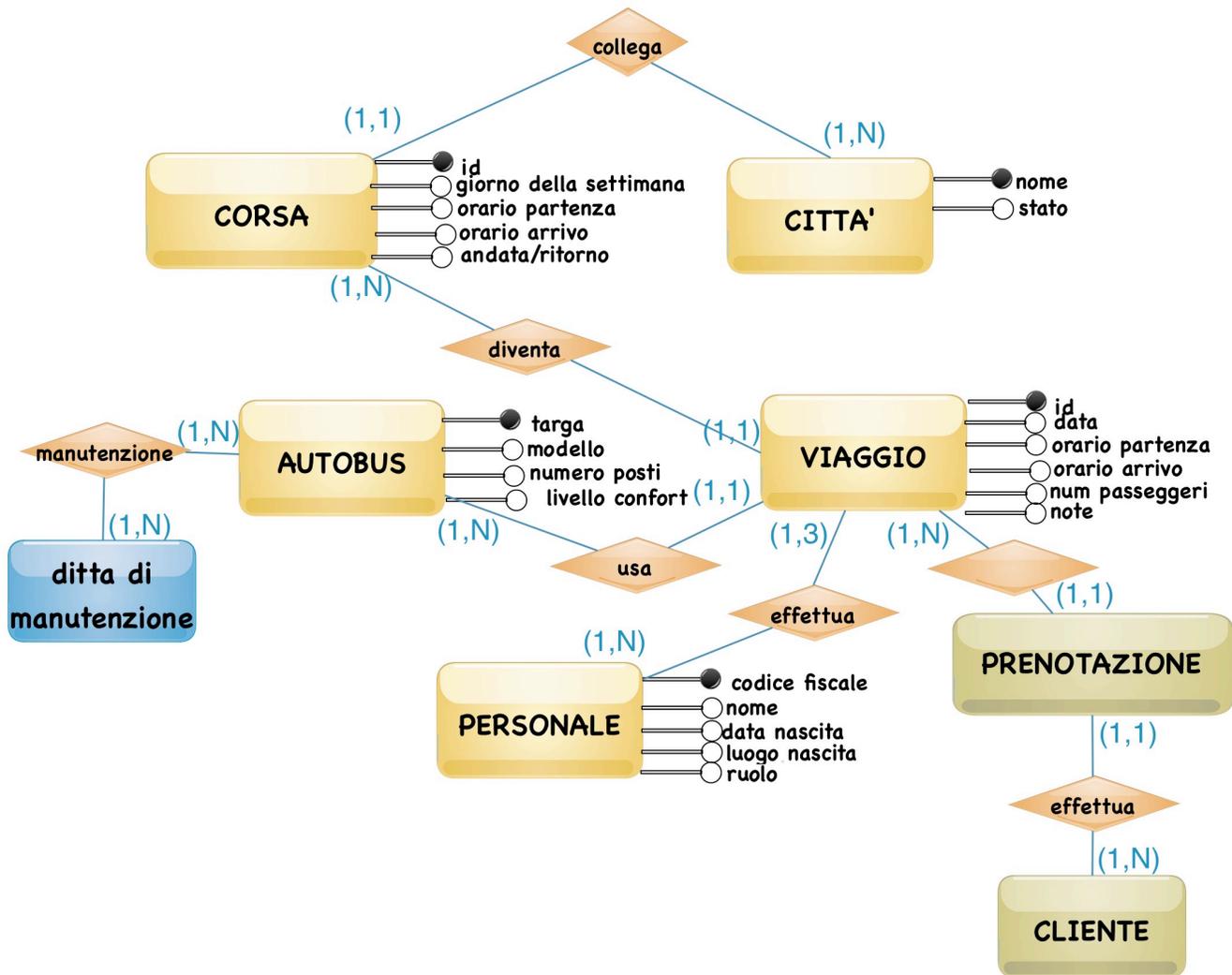
Il primo quesito chiedeva di ampliare il modello concettuale inserendo i Clienti focalizzandosi sul marketing.

Il Cliente rientra nel nostro modello attraverso una altra entità che è la PRENOTAZIONE. La cosa che richiede la traccia è di focalizzarsi sugli attributi del cliente utili alla profilazione e alla memorizzazioni di alcuni dati che ci vengono forniti:

stile di vita, interessi, realtà familiare, eccetera.

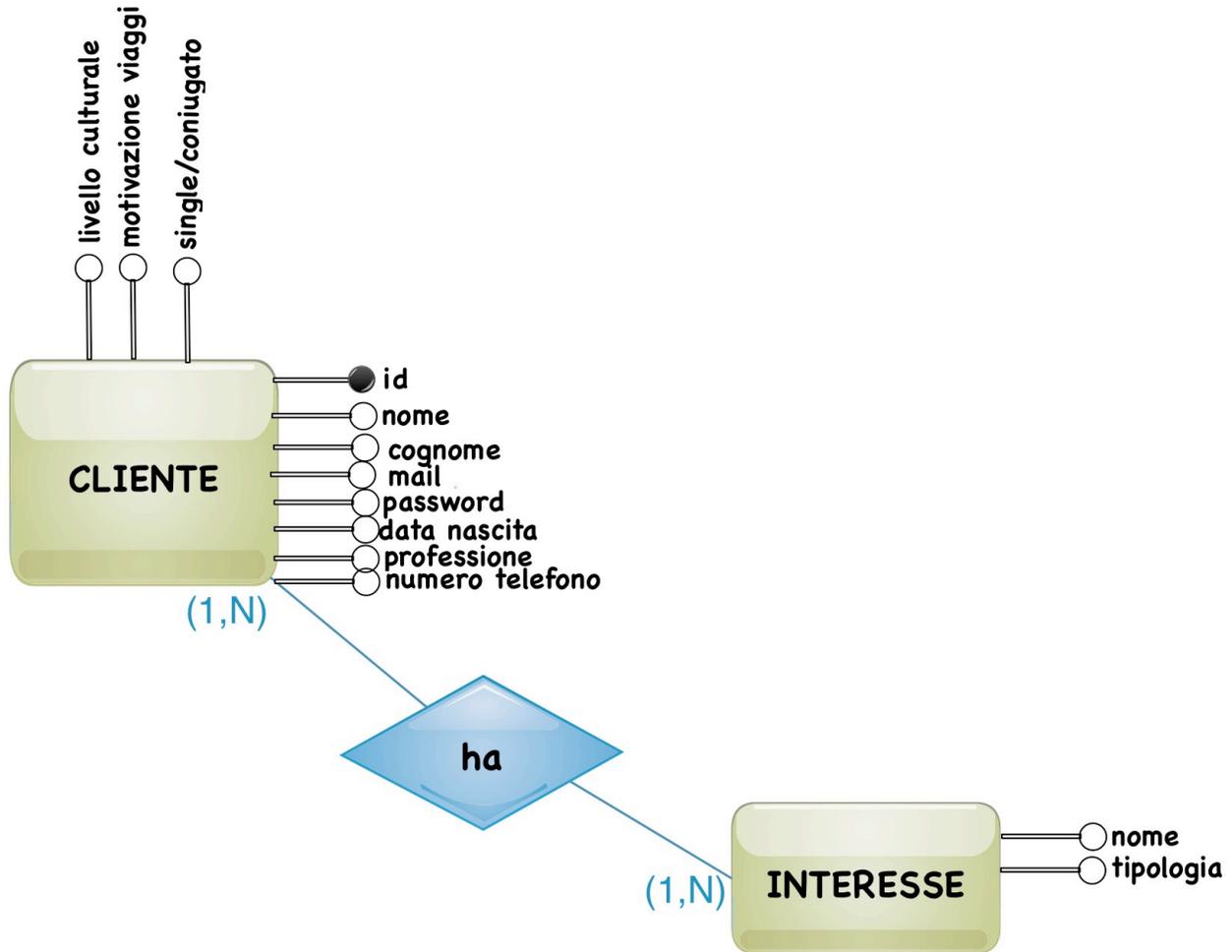
Punto a) integri la base di dati sviluppata nella prima parte inserendo opportunamente un'entità "Cliente";

Abbiamo aggiunto l'entità Cliente e prenotazione al nostro modello concettuale:



Non abbiamo definito gli attributi per l'entità CLIENTE, perché viene espressamente richiesto nel punto b:

Punto b) ne definisca gli attributi utili alle operazioni di marketing ipotizzate, motivando le scelte fatte, e ne derivi lo schema logico;



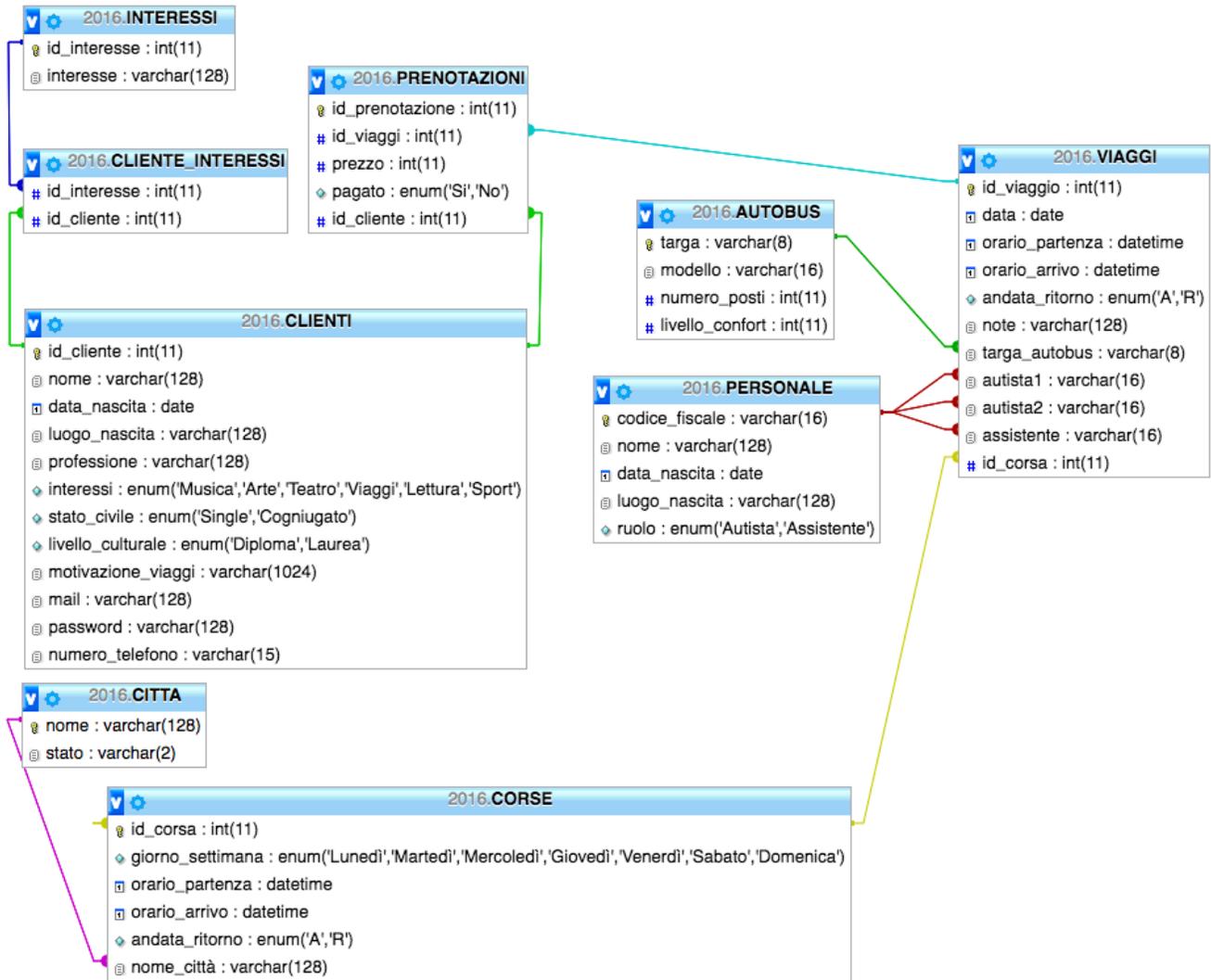
Gli attributi ipotizzati dovrebbe essere il cliente stesso a fornirli nel momento in cui si registra al sito della City2City per effettuare prenotazioni online. Dovrà inserire tutti i dati anagrafici, relativi alla identificazione, e saranno obbligatori un indirizzo e-mail per l'attivazione del profilo, una password per accessi successivi e anche, in maniera opzionale, un numero di telefono nel caso si volessero attivare le notifiche di prenotazione viaggio via SMS. La chiave primaria già viene ipotizzata come un id interno al sistema informativo. Potrebbe essere anche l'email, che per chiari motivi deve essere univoca, ma è preferibile un numero.

Le azioni di marketing dovrebbero essere rivolte a fidelizzare i clienti e a spingerli ad effettuare ulteriori viaggi rispetto ai necessari.

Gli interessi potrebbero essere Musica, Teatro, Arte, Lettura, Viaggi, Sport. Visto che un cliente può avere più di un interesse, appare opportuno creare l'entità INTERESSE in relazione **multi-a-multi** con CLIENTE.

Sicuramente, dopo aver profilato gli interessi, se essi fossero di natura artistica, si provvederà ad inviare delle offerte via newsletter con degli eventi che potrebbero interessare il cliente. Se ad esempio il cliente è un appassionato di sport, si può inviare un'offerta per una determinata competizione sportiva, oppure per una mostra d'arte se fosse interessato all'arte.

Il modello logico che ne consegue:



c. infine sviluppi le pagine web necessarie alla registrazione di un nuovo cliente, in un linguaggio di programmazione a propria scelta.

Di seguito riportiamo una form realizzata in HTML, CSS per la registrazione ipotetica di un cliente:

Registrazione Cliente

Nome	<input type="text"/>
Cognome	<input type="text"/>
email	<input type="text" value="example@sample.com"/>
Password	<input type="password"/>
Telefono	<input type="text"/>
Studi	<input type="text" value="Diploma"/>
Professione	<input type="text"/>
Stato Civile	<input type="checkbox"/> Single <input type="checkbox"/> Coniugato
Interessi	<input type="checkbox"/> Arte <input type="checkbox"/> Musica <input type="checkbox"/> Letteratura <input type="checkbox"/> Cinema <input type="checkbox"/> Sport <input type="checkbox"/> Viaggi
<input type="button" value="Registra"/>	

Il codice HTML è il seguente. Utilizza il CSS predefinito di Bootstrap, a parte impostare uno sfondo lithgrey. Durante il compito, scrivendo con carta e penna, il sorgente HTML poteva essere molto più minimale ma per completezza allego quello che effettivamente ho realizzato per la soluzione del compito:

```
<html>
  <head>
    <!-- Latest compiled and minified CSS -->
    <link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.6/css/bootstrap.min.css">

    <style>
    div {
      background-color lightgrey
    }
    </style>
  </head>

  <body>
    <div class="container">
      <form class="form-horizontal" method="POST" action="registracliente.php">
      <fieldset>

        <!-- Form Name -->
        <h1>Registrazione Cliente</h1>

        <!-- Text input-->
        <div class="form-group">
          <label class="col-md-4 control-label" for="nome">Nome</label>
          <div class="col-md-4">
            <input id="nome" name="nome" type="text" placeholder="" class="form-control input-md" required="">
```

```

    </div>
</div>

<!-- Text input-->
<div class="form-group">
  <label class="col-md-4 control-label" for="cognome">Cognome</label>
  <div class="col-md-4">
    <input id="cognome" name="cognome" type="text" placeholder="" class="form-control input-md" required="">

  </div>
</div>

<!-- Text input-->
<div class="form-group">
  <label class="col-md-4 control-label" for="email">email</label>
  <div class="col-md-4">
    <input id="email" name="email" type="text" placeholder="example@sample.com" class="form-control input-md"
required="">

  </div>
</div>

<!-- Password input-->
<div class="form-group">
  <label class="col-md-4 control-label" for="password">Password</label>
  <div class="col-md-4">
    <input id="password" name="password" type="password" placeholder="" class="form-control input-md" required="">

  </div>
</div>

<!-- Text input-->
<div class="form-group">
  <label class="col-md-4 control-label" for="telefono">Telefono</label>
  <div class="col-md-4">
    <input id="telefono" name="telefono" type="text" placeholder="" class="form-control input-md">

  </div>
</div>

<!-- Select Basic -->
<div class="form-group">
  <label class="col-md-4 control-label" for="Studi">Studi</label>
  <div class="col-md-4">
    <select id="Studi" name="Studi" class="form-control">
      <option value="1">Diploma</option>
      <option value="2">Laurea</option>
    </select>
  </div>
</div>

<!-- Text input-->
<div class="form-group">
  <label class="col-md-4 control-label" for="professione">Professione</label>
  <div class="col-md-4">
    <input id="professione" name="professione" type="text" placeholder="" class="form-control input-md">

  </div>
</div>

<!-- Multiple Checkboxes (inline) -->
<div class="form-group">
  <label class="col-md-4 control-label" for="statocivile">Stato Civile</label>
  <div class="col-md-4">
    <label class="checkbox-inline" for="statocivile-0">
      <input type="checkbox" name="statocivile" id="statocivile-0" value="1">
      Single
    </label>
    <label class="checkbox-inline" for="statocivile-1">
      <input type="checkbox" name="statocivile" id="statocivile-1" value="2">
      Coniugato
    </label>
  </div>
</div>

```

```

        </label>
    </div>
</div>

<!-- Multiple Checkboxes -->
<div class="form-group">
    <label class="col-md-4 control-label" for="interessi">Interessi</label>
    <div class="col-md-4">
        <div class="checkbox">
            <label for="interessi-0">
                <input type="checkbox" name="interessi" id="interessi-0" value="1">
                Arte
            </label>
        </div>
        <div class="checkbox">
            <label for="interessi-1">
                <input type="checkbox" name="interessi" id="interessi-1" value="2">
                Musica
            </label>
        </div>
        <div class="checkbox">
            <label for="interessi-2">
                <input type="checkbox" name="interessi" id="interessi-2" value="">
                Letteratura
            </label>
        </div>
        <div class="checkbox">
            <label for="interessi-3">
                <input type="checkbox" name="interessi" id="interessi-3" value="">
                Cinema
            </label>
        </div>
        <div class="checkbox">
            <label for="interessi-4">
                <input type="checkbox" name="interessi" id="interessi-4" value="">
                Sport
            </label>
        </div>
        <div class="checkbox">
            <label for="interessi-5">
                <input type="checkbox" name="interessi" id="interessi-5" value="">
                Viaggi
            </label>
        </div>
    </div>
</div>

<!-- Button -->
<div class="form-group">
    <label class="col-md-4 control-label" for="salva"></label>
    <div class="col-md-4">
        <button id="salva" name="salva" class="btn btn-primary">Registra</button>
    </div>
</div>

</fieldset>
</form>
</div>

</body>
</html>

```

Per l'inserimento del cliente all'interno di un Database, si può portare un breve esempio di PHP. Infatti la form, al submit del button, richiamerà l'action registracliente.php. che potrà essere illustrata, in maniera molto sintetica, dal seguente pezzetto di codice che riporta le fasi minimali di registrazione: connessione al DB e inserimento in tabella dei parametri inviati in POST.

```

<?php
$conn=mysql_connect("localhost","root","root"); // Connessione al Database
if(!$conn){
die ('Non riesco a connettermi: errore '.mysql_error()); // errore
}

$db_selected=mysql_select_db("nomedatabase",$conn);

if(!$db_selected){
die ('Errore nella selezione del database: errore '.mysql_error());
}

if($_POST["nome"] != "" && $_POST["cognome"] != "" && $_POST["password"]!= "" &&
$_POST["email"] != ""){
    // se i parametri obbligatori non sono vuoti
$query_registrazione = mysql_query("INSERT INTO CLIENTI(nome, data_nascita, luogo_nascita,
professione, stato_civile, livello_culturale, motivazione_viaggi, mail, password,
numero_telefono)
VALUES (".$_POST['nome'].", ".$_POST['data_nascita'].", ".$_POST['luogo_nascita'].", ".
$_POST['professione'].", ".$_POST['stato_civile'].", ".$_POST['livello_culturale'].", ".
$_POST['motivazione_viaggi'].", ".$_POST['mail'].", ".
$_POST['password'].", ".$_POST['numero_telefono'].");" or die ("query di registrazione
non riuscita".mysql_error());
// Inserisci dati o se la query fallisce ritorna errore
}else{
header('location:index.php?action=registration&errore=Non hai compilato tutti i campi
obbligatori'); // se le prime condizioni non vanno bene entra in questo ramo else
}
if(isset($query_registrazione)){ //se la registrazione è andata a buon fine
}else{
echo "non ti sei registrato con successo";
}
?>

```

Per semplificare, sono stati omessi gli inserimenti degli interessi che andavano sulla tabella CLIENTE_INTERESSI.