**Università della Calabria**

**Dip. di Matematica e Informatica**

**Corso di laurea in informatica**

**a.a. 2019/2020**

***Progetto per il Corso di***

***Basi di dati***

Gruppo 21

Risino Gabriele

Cirimele Davide

Brunetti Domenico

**Prof. P. Rullo**

Esercitatori

**Ing. G. Laboccetta**

**Dott.ssa D. Angilica**

1. Tematica Progettuale

C:\Users\Corrado\AppData\Local\Temp\ksohtml14688\wps3.png

La progettazione del sistema informatico in esame riguarda la gestione di un istituto comprensivo.

|  |  |
| --- | --- |
|  | **Requisiti Richiesti** |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17 | L’istituto comprensivo StudioOnline vuole riorganizzare il proprio database a seguito di alcuni aggiornamenti nella normativa.  Il complesso scolastico include tre gradi: scuola dell’infanzia, scuola primaria e scuola secondaria inferiore. Ciascuna scuola è distribuita in diversi plessi e ciascun plesso ospita anche più di un grado scolastico.  Le iscrizioni avvengono nel mese di gennaio: i genitori iscrivono i propri figli e di ognuno di loro viene registrata l’anagrafica.  Ogni studente, in un anno scolastico (01/09-19/06), è iscritto ad uno dei 3 gradi dell’istituto ma si vuole memorizzare per ciascun ragazzo lo storico delle iscrizioni. All’atto dell’iscrizione, ogni studente è associato ad una classe.  I plessi sono identificati da un indirizzo e sono composti da delle aule. Di ogni aula si conoscono i metri quadri, hanno un numero massimo di studenti ospitabili e la fascia di età degli studenti ospitabili.  Ogni classe, che ogni anno può cambiare aula, è identificata da un numero (1-4 per l’infanzia, 1-5 per la primaria e 1-3 per le medie) e da una lettera dell’alfabeto (aule di gradi diversi possono avere la stessa coppia). Di ciascun insegnante si conosce, ogni anno, il numero di ore lavorative settimanali e le classi a cui è assegnato. |

**Specifica delle Operazioni sui dati previste**

1. Inserire una nuova iscrizione

2.Trovare gli studenti iscritti alla scuola secondaria nell’anno 2020.

3.Trovare gli studenti che hanno frequentato almeno una classe per ogni grado dell’istituto.

4.Trovare i plessi con il numero di studenti ospitabili più alto.

5.Trovare, per ogni anno, le scuole con il maggior numero di iscritti.

6.Controllare che ad un’aula non vengano assegnati più iscritti di quanti ne possa ospitare.

7.Alla scadenza delle iscrizioni annuali (1 febbraio) si memorizzi il numero di nuovi iscritti per grado e classe (1-4 infanzia, 1-5 primaria e 1-3 medie).

C:\Users\Corrado\AppData\Local\Temp\ksohtml14688\wps7.png

2 Raccolta e Analisi dei Requisiti

2.1Analisi dei requisiti

2.2.1Eliminazione delle Ambiguità

|  |  |  |  |
| --- | --- | --- | --- |
| **Linea** | Termine | Sinonimi | Motivazione Correzione |
| 1-3 | istituto | complesso | sinonimi |
| 3 | gradi | scuola | sinonimi |
| 4-15 | Secondaria inferiore | medie | sinonimi |
| 8-9 | studente | ragazzo | sinonimi |

2.2.2Ristrutturazione dei Requisiti Richiesti

|  |  |
| --- | --- |
|  | **Specifiche ristrutturate** |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15 | L’istituto include tre scuole : scuola dell’infanzia, scuola primaria e scuola secondaria inferiore. Ogni scuola è distribuita in diversi plessi e ciascun plesso può ospitare anche più di una scuola.  Le iscrizioni devono avvenire nel mese di gennaio: per ogni studente viene registrata l’anagrafica.  Ogni studente, in un anno scolastico (01/09-19/06), è iscritto ad uno delle 3 scuole dell’istituto ma si vuole memorizzare per ciascun studente lo storico delle iscrizioni. All’atto dell’iscrizione, ogni studente è associato ad una classe.  I plessi sono identificati da un indirizzo e sono composti da delle aule. Di ogni aula si conoscono i metri quadri, il numero massimo di studenti ospitabili e la fascia di età degli studenti ospitabili.  Ogni classe, che ogni anno può cambiare aula, è identificata da un numero (1-4 per l’infanzia, 1-5 per la primaria e 1-3 per le medie) e da una lettera dell’alfabeto (classi di scuole diverse possono avere la stessa coppia). Di ciascun insegnante si conosce, ogni anno, il numero di ore lavorative settimanali e le classi a cui è assegnato. |

* + 1. Raffinamento delle Specifiche e Individuazione dei Concetti di Base

|  |  |
| --- | --- |
|  | **Frasi di carattere generale** |
| 1  2 | L’istituto comprensivo StudioOnline vuole riorganizzare il proprio database a seguito di alcuni aggiornamenti nella normativa. |

|  |  |
| --- | --- |
|  | **Frasi relative alle scuole** |
| 3  4  5 | L’istituto include tre scuole : scuola dell’infanzia, scuola primaria e scuola secondaria inferiore. Ciascuna scuola è distribuita in diversi plessi e ciascun plesso ospita anche più di una scuola. |

|  |  |
| --- | --- |
|  | **Frasi relative alle iscrizioni** |
| 6  7 | Le iscrizioni avvengono nel mese di gennaio: i genitori iscrivono i propri figli e di ognuno di loro viene registrata l’anagrafica. |

|  |  |
| --- | --- |
|  | **Frasi relative agli studenti** |
| 7  8  9 | Ogni studente, in un anno scolastico (01/09-19/06), è iscritto ad uno delle 3 scuole dell’istituto ma si vuole memorizzare per ciascun studente lo storico delle iscrizioni. All’atto dell’iscrizione, ogni studente è associato ad una classe. |

|  |  |
| --- | --- |
|  | **Frasi relative ai plessi** |
| 10 | I plessi sono identificati da un indirizzo e sono composti da delle aule |

|  |  |
| --- | --- |
|  | **Frasi relative alle aule** |
| 11  12 | Di ogni aula si conoscono i metri quadri, hanno un numero massimo di studenti ospitabili e la fascia di età degli studenti ospitabili. |

|  |  |
| --- | --- |
|  | **Frasi relative alle classi** |
| 13  14  15 | Ogni classe, che ogni anno può cambiare aula, è identificata da un numero (1-4 per l’infanzia, 1-5 per la primaria e 1-3 per le medie) e da una lettera dell’alfabeto (aule di scuole diverse possono avere la stessa coppia). |

|  |  |
| --- | --- |
|  | **Frasi relative agli insegnanti** |
| 16  17 | Di ciascun insegnante si conosce, ogni anno, il numero di ore lavorative settimanali e le classi a cui è assegnato |

**Specifica delle Operazioni sui dati previste**

**esplicite:**

1.Inserire una nuova iscrizione.

2. Trovare gli studenti iscritti alla scuola secondaria nell’anno 2020.

3. Trovare gli studenti che hanno frequentato almeno una classe per ogni grado dell’istituto.

4. Trovare i plessi con il numero di studenti ospitabili più alto.

5. Trovare, per ogni anno, le scuole con il maggior numero di iscritti.

6. Controllare che ad un’aula non vengano assegnati più iscritti di quanti ne possa ospitare.

7. Alla scadenza delle iscrizioni annuali (1 febbraio) si memorizzi il numero di nuovi iscritti 8. per grado e classe (1-4 infanzia, 1-5 primaria e 1-3 medie).

**IMPLICITE**:

1.Controllare lo studenti rientri nei limiti d’età della classe

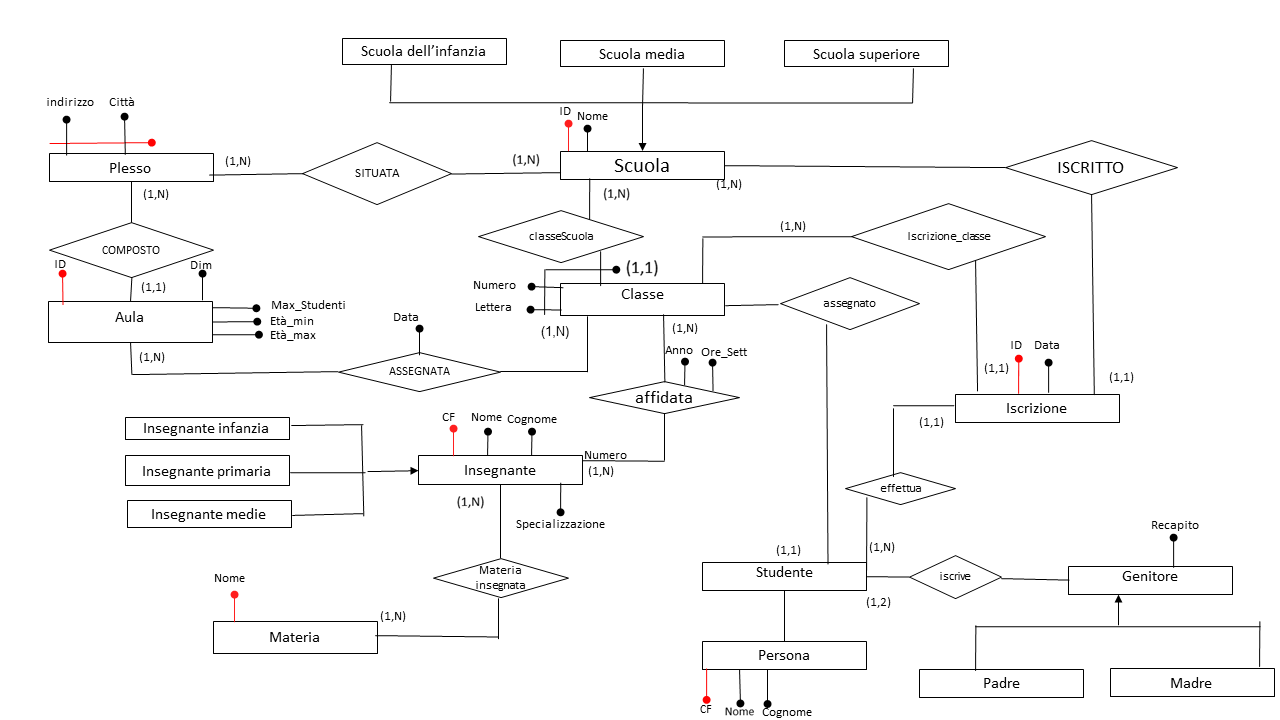
2.Controllare l’iscrizione avvenga nel mese di gennaio

3.Evitare a uno studente di iscriversi a più classi diverse nello stesso anno

4.Gestire i contatori degli iscritti nelle singole classi e nelle scuole

3) Progettazione Concettuale

***3.1 SCHEMI E-R***



***3.2  Documentazione dello schema E-R***

***3.2.1 Dizionario dei dati***

|  |  |  |  |
| --- | --- | --- | --- |
| **Entità** | **Descrizione** | **Attributi** | **Identificatore** |
| Scuola | Tipo di istituzione culturale. | Id, nome | Id |
| Plesso | Struttura fisica di una scuola. | Indirizzo, città | Indirizzo, città |
| Classe | L’insieme di studenti | Numero, lettera | Numero, lettera, scuola |
| Aula | Luogo fisico in cui si svolgono le attività | Id, dim, max\_studenti, età\_min, età\_max | ID |
| Insegnante | Professionista che impartisce le lezioni | CF, nome, cognome, specializzazione | CF |
| Materia | Argomento dell’insegnamento | Nome | Nome |
| Studente | Persona iscritta ad una scuola | CF, nome, cognome | CF |
| Iscrizione | Iscrizione di uno studente ad una classe in una determinata scuola | ID, data | ID |
| Genitore | Genitore dello studente | Recapito | Recapito |

***3.2.2 Descrizione entità***

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  SCUOLA | | | |
| Descrizione | Istituzione culturale | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| ID | Numerico | Obbligatorio | Numero identificativo della scuola |
| Nome | Stringa | Obbligatorio | Nome della scuola |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  PLESSO | | | |
| Descrizione | Struttura fisica di una scuola | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| Indirizzo | Stringa | Obbligatorio | Indirizzo dove è situato il plesso |
| Città | Stringa | Obbligatorio | Città del plesso |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  CLASSE | | | |
| Descrizione | L’insieme di studenti | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| Numero | Numerico | Obbligatorio, > 0 | Grado della classe |
| Lettera | Stringa | Obbligatorio | Sezione della classe |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  AULA | | | |
| Descrizione | Luogo fisico in cui si svolgono le attività | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| ID | Numerico | Obbligatorio | Id dell’aula |
| Dim | Numerico | Obbligatorio, > 0 | Dimensione dell’aula |
| max\_studenti | Numerico | Obbligatorio | Numero massimo di studenti |
| età\_min | Numerico | Obbligatorio | Età minima degli studenti |
| Età\_max | Numerico | Obbligatorio | Età massima degli studenti |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  INSEGNANTE | | | |
| Descrizione | Professionista che impartisce le lezioni | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| CF | Stringa | Obbligatorio | Codice identificativo dell’insegnante |
| Nome | Stringa |  | Nome dell’insegnante |
| Cognome | Stringa | Obbligatorio | Cognome dell’insegnante |
| specializzazione | Stringa | Obbligatorio | Specializzazione dell’insegnante |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  MATERIA | | | |
| Descrizione | Argomento dell’insegnamento | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| Nome | Stringa | Obbligatorio | Nome della materia |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  STUDENTE | | | |
| Descrizione | Persona iscritta ad una scuola | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| CF | Stringa | Obbligatorio | Codice identificativo dello studente |
| Nome | Stringa | Obbligatorio | Nome dello studente |
| Cognome | Stringa | Obbligatorio | Cognome dello studente |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  ISCRIZIONE | | | |
| Descrizione | Iscrizione di uno studente ad una classe in una determinata scuola | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| ID | Numerico | Obbligatorio | Numero identificativo dell’iscrizione |
| Data | Data | Obbligatorio | Data dell’iscrizione |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  GENITORE | | | |
| Descrizione | Genitore dello studente | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| Recapito | Numerico | Obbligatorio | Numero telefonico del genitore |

**3.2.3 DESCRIZIONE RELAZIONI**

|  |  |
| --- | --- |
| Relazione: situata | |
| scuola associato ad una o più plessi | |
| scuola, plesso | |
| Entità | Cardinalità |
| scuola | (1,n) |
| plesso | (1,n) |
| Relazione: composto | |
| aula associata ad un plesso | |
| Iscrizioni, classe | |
| Entità | Cardinalità |
| Plesso | (1,n) |
| Aula | (1,n) |
| Relazione: assegnata | |
| classe associata ad un’aula | |
| Aula, classe | |
| Entità | Cardinalità |
| aula | (1,1) |
| classe | (0,n) |
| relazione: ASSEGNATA | |
| le aule che compongono il plesso | |
| Plesso, aule | |
| Entit | Cardinalità |
| plesso | (1,1) |
| aule | (1,n) |

|  |  |
| --- | --- |
| Relazione: AFFIDATA | |
| insegnante associato ad una o più classi | |
| Insegnante, classe | |
| Entità | Cardinalità |
| insegnante | (1,n) |
| classe | (1,n) |
| relazione: insegna | |
| insegnante associato ad una o piu materie | |
| Insegnante, materia | |
| Entità | Cardinalità |
| Insegnante | (1,n) |
| materia | (1,n) |
| Relazione: assegnato | |
| studente associata ad una classe | |
| Studente, classe | |
| Entità | Cardinalità |
| Studente | (1,1) |
| classe | (1,n) |
| relazione: iscrizione\_classe | |
| iscrizione associata ad una classe | |
| Iscrizione, classe | |
| Entità | Cardinalità |
| iscrizione | (1,1) |
| classe | (1,n) |

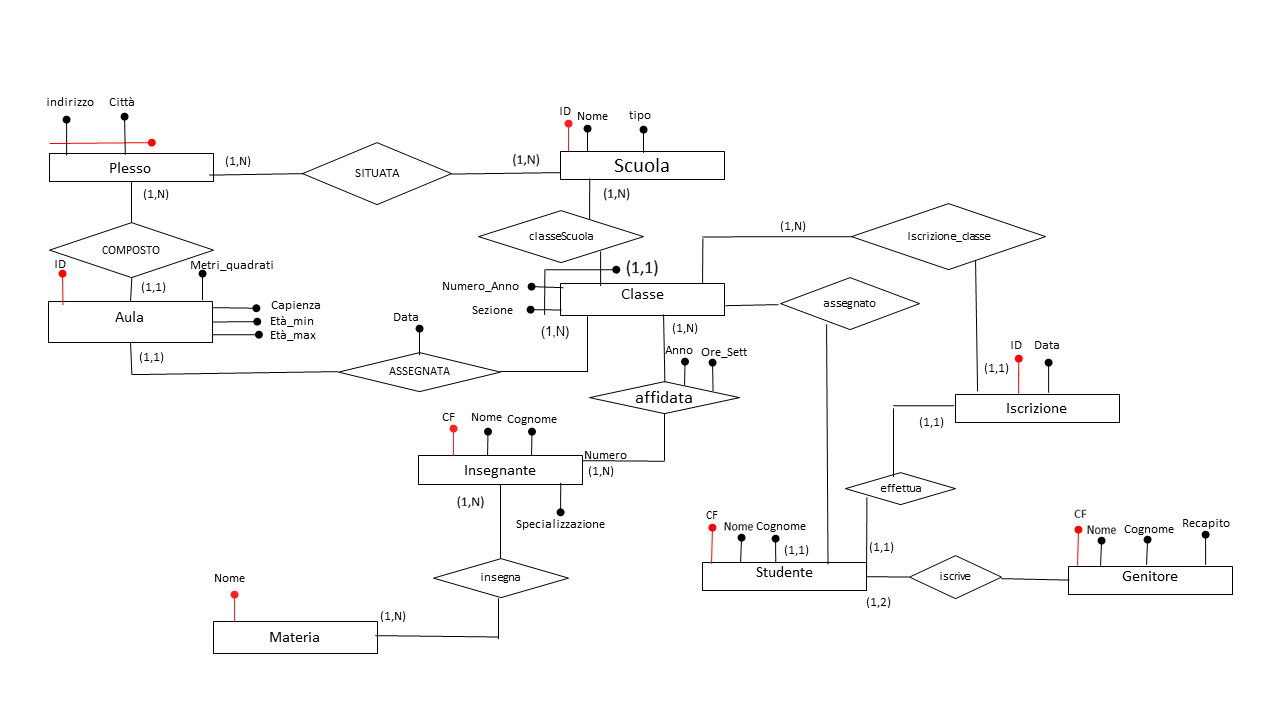
|  |  |
| --- | --- |
| relazione: effettua | |
| iscrizione associata ad uno studente | |
| Iscrizione, studente | |
| Entità | Cardinalità |
| iscrizione | (1,1) |
| studente | (1,1) |

VINCOLI NON ESPRESSI DALLO SCHEMA E/R

|  |
| --- |
| **Regole di Vincolo** |
| * Le iscrizioni avvengono nel mese di gennaio |
| * Ogni classe, ogni anno può cambiare aula. |
| * Le iscrizioni devono rispettare le fasce d’età * Ogni studente può iscriversi a una classe ogni anno |

4) Progettazione Logica

4.1 Ristrutturazione dello schema E-R



|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  SCUOLA | | | |
| Descrizione | Istituzione culturale | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| ID | int | primary key | Numero identificativo della scuola |
| Nome | varchar(50) | not null | Nome della scuola |
| Tipo | varchar(30) | not null | Tipo di scuola |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  PLESSO | | | |
| Descrizione | Struttura fisica di una scuola | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| Indirizzo | varchar(50) | primary key | Indirizzo dove è situato il plesso |
| Città | varchar(30) | primary key | Città del plesso |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  CLASSE | | | |
| Descrizione | L’insieme di studenti | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| Numero\_Anno | tinyint | primary key | Grado della classe |
| Sezione | char(1) | primary key | Sezione della classe |
| Scuola | int | foreign key | ID della scuola |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  AULA | | | |
| Descrizione | Luogo fisico in cui si svolgono le attività | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| ID | int | primary key | Id dell’aula |
| Metri\_quadrati | int unsigned | not null | Dimensione dell’aula |
| Capienza | tinyint | not null | Numero massimo di studenti |
| età\_min | tinyint | not null | Età minima degli studenti |
| Età\_max | tinyint | not null | Età massima degli studenti |
| città\_plesso | varchar(30) | foreign key | Città plesso |
| indirizzo\_plesso | varchar(50) | foreign key | Indirizzo plesso |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  INSEGNANTE | | | |
| Descrizione | Professionista che impartisce le lezioni | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| CF | char(16) | primary key | Codice identificativo dell’insegnante |
| Nome | varchar(30) | not null | Nome dell’insegnante |
| Cognome | varchar(30) | not null | Cognome dell’insegnante |
| specializzazione | varchar(30) | not null | Specializzazione dell’insegnante |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  MATERIA | | | |
| Descrizione | Argomento dell’insegnamento | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| Nome | varchar(30) | primary key | Nome della materia |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  ISCRIZIONE | | | |
| Descrizione | Iscrizione di uno studente ad una classe in una determinata scuola | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| ID | int | primary key | Numero identificativo dell’iscrizione |
| Data | date | not null | Data dell’iscrizione |
| Studente | char(16) | foreign key | Numero identificativo dell’iscrizione |
| Numero\_classe | tinyint unsigned | foreign key | Numero classe |
| Lettera\_classe | char(1) | foreign key | Lettera classe |
| Scuola\_classe | int | foreign key | Scuola classe |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  GENITORE | | | |
| Descrizione | Genitore dello studente | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| Recapito | char(10) | not null | Numero telefonico del genitore |
| CF | char(16) | primary key | Codice identificativo del genitore |
| Nome | varchar(30) | not null | Nome del genitore |
| Cognome | varchar(30) | not null | Cognome del genitore |

|  |  |  |  |
| --- | --- | --- | --- |
| ENTITÀ  STUDENTE | | | |
| Descrizione | Persona iscritta ad una scuola | | |
| Nome Attributo | Tipo di dato | Vincolo | Descrizione |
| CF | char(16) | primary key | Codice identificativo dello studente |
| Nome | varchar(30) | not null | Nome dello studente |
| Cognome | varchar(30) | not null | Cognome dello studente |
| Numero\_classe | tinyint unsigned | foreign key | Numero classe |
| Sezione\_classe | char(1) | foreign key | Lettera classe |
| Scuola\_classe | int | foreign key | Scuola classe |
| Genitore\_1 | char(16) | foreign key | Genitore 1 |
| Genitore\_2 | char(16) |  | Genitore 2 |

**4.3 TRADUZIONE ENTITÀ**

Scuola(ID, nome, tipo)

Plesso(Indirizzo, città)

Aula(ID, metri\_quadrati, capienza, età\_min, età\_max, (città\_plesso,indirizzo\_plesso)\*)

Con vincolo di integrità referenziale tra gli attributi città\_plesso, indirizzo\_plesso e

La relazione Plesso

Classe(Sezione, Numero\_Anno, Scuola)

Iscrizione(ID, data, Studente\*, **(numero\_classe, lettera\_classe, scuola\_classe)\*)**

Con vincolo di integrità referenziale tra gli attributi numero\_classe, lettera classe, scuola\_classe

E la relazione Classe, con vincolo di integrità referenziale tra l’attributo studente e la relazione studente.

Insegnante(CF, nome, cognome)

Materia(Nome)

Studente(CF, nome, cognome, (numero\_classe, lettera\_classe, scuola\_classe)\*, genitore1\*, genitore2)

Con vincolo di integrità referenziale tra gli attributi numero\_classe, lettera classe, scuola\_classe

e la relazione Classe, con vincolo di integrità referenziale tra l’attriburo genitore1\* e la relazione Genitore.

Genitore(CF, nome, cognome, recapito)

ScuolaPlesso( (Plesso\_indirizzo, Plesso\_città)\* , scuola\*)

 Con vincolo di integrità referenziale tra gli attributi plesso\_città, plesso\_indirizzo e la

            Relazione Plesso e tra l’attributo scuola e la relazione Scuola.

ClasseInsegnante (  (numero\_classe, lettera\_classe, scuola\_classe)\*, insegnante\*)

Con vincolo di integrità referenziale tra gli attributi numero\_classe, lettera classe,  scuola\_classe  e la relazione Classe, e l’attributo insegnante e la relazione Insegnante

MateriaInsegnante( materia\*, insegnante\*)

Con vincolo di integrità referenziale tra gli attributi materia e la relazione Materia e l’attributo insegnante e la relazione insegnante

**4.3 TRADUZIONE RELAZIONI**

Situata( (plesso\_città, plesso\_indirizzo)\*, scuola\*)

Con vincolo di integrità referenziale tra gli attributi plesso\_città, plesso\_indirizzo e la

Relazione Plesso e tra l’attributo scuola e la relazione Scuola.

Assegnata ( Aula\*, (Numero\_classe, Lettera\_classe, Scuola\_Classe)\*)

Con vincolo di integrità referenziale tra gli attributi Numero\_classe, Lettera\_classe e la relazione classe e l’attributo scuola\_Classe e la reazione Scuola.

Classe\_insegnante( (Numero\_anno\_classe, Sezione\_classe, scuola\_classe)\*, insegnante\*, anno, ore\_settCon vincolo di integrità referenziale tra gli attributi numero\_anno\_classe, Sezione\_classe, scuola\_classe e la relazione Classe, e l’attributo insegnante e la relazione Insegnante,

Insegna( Materia\*, Insegnante\*)

Con vincolo di integrità referenziale tra gli attributi materia e la relazione Materia e tra l’attributo insegnante e la relazione Insegnante.

1. **PROGETTAZIONE FISICA**
   1. **DEFINIZIONE DELLO SCHEMA DELLA BASE DI DATI**

**TABLE:**

**create table if not exists scuola**

**(**

**id int auto\_increment primary key,**

**nome varchar(50) not null,**

**tipo varchar(30) not null check (tipo = 'infanzia' or tipo='primaria' or tipo= 'secondaria'),**

**totale\_iscritti int**

**);**

**create table if not exists genitore**

**(**

**cf char(16) primary key,**

**nome varchar(30) not null,**

**cognome varchar(30) not null,**

**parentela char(5) not null check (parentela= 'madre' or parentela= 'padre'),**

**recapito char(10) not null**

**);**

**create table if not exists plesso**

**(**

**indirizzo varchar(50),**

**citta varchar(30),**

**primary key(indirizzo,citta)**

**);**

**create table if not exists scuola\_plesso**

**(**

**citta\_plesso varchar(30),**

**indirizzo\_plesso varchar(50),**

**scuola int not null,**

**foreign key(indirizzo\_plesso,citta\_plesso) references plesso(indirizzo,citta),**

**foreign key(scuola) references scuola(id)**

**);**

**create table if not exists aula**

**(**

**id int auto\_increment primary key,**

**metri\_quadrati int unsigned not null,**

**capacita tinyint unsigned not null,**

**min\_eta tinyint unsigned not null,**

**max\_eta tinyint unsigned not null,**

**citta\_plesso varchar(30),**

**indirizzo\_plesso varchar(50),**

**foreign key (indirizzo\_plesso,citta\_plesso) references plesso(indirizzo,citta)**

**);**

**create table if not exists classe**

**(**

**numero\_anno tinyint unsigned,**

**sezione char(1),**

**scuola int,**

**primary key(numero\_anno,sezione,scuola),**

**foreign key(scuola) references scuola(id)**

**);**

**create table if not exists studente**

**(**

**cf char(16) primary key,**

**nome varchar(30) not null,**

**cognome varchar(30) not null,**

**scuola int not null,**

**genitore1 char(16) not null,**

**genitore2 char(16),**

**sezione char(1),**

**scuola\_classe int,**

**numero\_classe tinyint unsigned not null,**

**foreign key (numero\_classe,sezione,scuola\_classe) references classe(numero\_anno,sezione,scuola),**

**foreign key (scuola) references scuola(id),**

**foreign key (genitore1) references genitore(cf)**

**);**

**create table if not exists iscrizione**

**(**

**id int auto\_increment primary key,**

**data\_iscrizione date not null,**

**studente char(16) not null,**

**numero\_classe tinyint unsigned,**

**sezione char(1),**

**scuola int,**

**foreign key (numero\_classe,sezione,scuola) references classe(numero\_anno,sezione,scuola),**

**foreign key (studente) references studente(cf)**

**);**

**create table if not exists aula\_classe**

**(**

**data\_assegnamento date not null,**

**numero\_classe tinyint unsigned not null,**

**sezione char(1) not null,**

**scuola int not null,**

**aula int not null,**

**foreign key (aula) references aula(id),**

**foreign key (numero\_classe, sezione, scuola) references classe(numero\_anno,sezione,scuola)**

**);**

**create table if not exists insegnante**

**(**

**cf char(16) primary key,**

**nome varchar(30) not null,**

**cognome varchar(30) not null,**

**specializzazione varchar(30) not null,**

**check (specializzazione = 'infanzia' or specializzazione = 'primaria' or specializzazione = 'secondaria')**

**);**

**create table if not exists classe\_insegnante**

**(**

**ore\_settimana int unsigned not null,**

**anno date not null,**

**numero\_classe tinyint unsigned not null,**

**sezione char(1) not null,**

**scuola int not null,**

**insegnante char(16) not null,**

**foreign key (insegnante) references insegnante(cf),**

**foreign key (numero\_classe,sezione,scuola) references classe(numero\_anno,sezione,scuola)**

**);**

**create table if not exists materia**

**(**

**nome varchar(30) primary key**

**);**

**create table if not exists materia\_insegnante**

**(**

**materia varchar(30),**

**insegnante char(16),**

**foreign key (materia) references materia(nome),**

**foreign key (insegnante) references insegnante(cf)**

**);**

**TRIGGER:**

**create trigger controllo\_iscrizione\_studente before insert on studente**

**for each row**

**begin**

**if month(current\_date())>=2 then**

**signal sqlstate '45000' set message\_text = "Le iscrizioni sono chiuse";**

**end if;**

**end; //**

**delimiter ;**

**delimiter //**

**create trigger controllo\_iscrizione\_iscrizione before insert on iscrizione**

**for each row**

**begin**

**if month(current\_date())>=2 then**

**signal sqlstate '45000' set message\_text = "Le iscrizioni sono chiuse";**

**end if;**

**end; //**

**delimiter ;**

**delimiter $$**

**create trigger iscrizione\_studente after insert on studente**

**for each row**

**begin**

**insert into iscrizione(data\_iscrizione,studente,numero\_classe,sezione,scuola)**

**values(current\_date(),new.cf,new.numero\_classe,new.sezione,new.scuola);**

**end; $$**

**delimiter ;**

**delimiter $$**

**create trigger check\_max\_classe before insert on studente**

**for each row**

**begin**

**set @studenti\_classe = (select count(\*)**

**from classe**

**inner join studente on studente.numero\_classe=classe.numero\_anno**

**and studente.sezione=classe.sezione**

**where classe.scuola=new.scuola);**

**set @max\_studenti = (select max\_studenti**

**from aula**

**inner join aula\_classe on aula\_classe.aula=aula.id**

**where aula\_classe.sezione=new.sezione**

**and aula\_classe.sezione=new.numero\_classe);**

**if @max\_studenti < @studenti\_classe + 1 then**

**signal sqlstate '45000' set message\_text = 'capacità aula superata';**

**end if;**

**end; $$**

**delimiter ;**

**delimiter !!**

**create trigger check\_prof\_classe before insert on classe\_insegnante**

**for each row**

**begin**

**set @scuola = (select tipo from scuola where id = new.scuola);**

**if exists (select \***

**from insegnante,classe\_insegnante**

**where insegnante.specializzazione=@scuola) then**

**signal sqlstate '45000' set message\_text = 'Non puoi insegnare in questa classe';**

**end if;**

**end; !!**

**delimiter;**

**delimiter $$**

**create trigger check\_num\_classe before insert on classe**

**for each row**

**begin**

**set @grado = (select distinct tipo from scuola inner join classe on scuola.id = classe.scuola where scuola.id=new.scuola);**

**if @grado = 'infanzia' and (new.numero\_anno<1 or new.numero\_anno>4) or @grado = 'primaria' and (new.numero\_anno<1 or new.numero\_anno>5) or @grado = 'secondaria' and (new.numero\_anno<1 or new.numero\_anno>3) then**

**signal sqlstate '45000' set message\_text = 'numero di classe non valido per il tipo di scuola';**

**end if;**

**end; $$**

**delimiter ;**

**EVENT:**

**delimiter //**

**create event if not exists studenti\_iscritti**

**on schedule every 1 year starts '2021-02-01 00:00:00'**

**do**

**begin**

**drop view if exists tot\_iscritti;**

**create view tot\_iscritti as**

**select tipo,numero\_classe,sezione,count(\*) as tot\_iscritti**

**from scuola inner join iscrizione on iscrizione.scuola\_classe = scuola.id**

**group by scuola,numero\_classe,sezione,year(data\_iscrizione);**

**end; //**

**delimiter ;**

* 1. **DEFINIZIONE DELLE INTERROGAZIONI PER LA VISUALIZZAZIONE DEI DATI**

**5.2.1 SCRIPT SQL – DML: ANALISI PUNTUALI**

1. Inserire una nuova iscrizione

insert into iscrizione(data\_iscrizione,studente,numero\_classe,sezione,scuola) values (now(),'DDCGDN80A01A773O',1,'A',6)

2) Trovare gli studenti iscritti alla scuola secondaria nell’anno 2020.

select distinct studente.cf,scuola.tipo

from iscrizione,scuola,studente

where year(iscrizione.data\_iscrizione)= '2020'

and scuola.tipo ='secondaria'

and iscrizione.scuola = scuola.id

3) Trovare gli studenti che hanno frequentato almeno una classe per ogni grado dell’istituto.

select cf,nome,cognome,genitore1 from studente

where cf in

(select cf from (select distinct \*

from scuola

inner join(

select studente.cf, iscrizione.scuola,iscrizione.data\_iscrizione

from studente

inner join iscrizione on iscrizione.studente = studente.cf

)as codici on codici.scuola=scuola.id)

as tab

group by cf

having count(\*)>=3

)

1. Trovare i plessi con il numero di studenti ospitabili più alto

select citta\_plesso,indirizzo\_plesso, max(somma)

from ( select citta\_plesso,indirizzo\_plesso,sum(capienza) as somma

from aula

group by citta\_plesso,indirizzo\_plesso

)

1. Trovare per ogni anno, le scuole con il maggior numero di iscritti

select scuola,max(iscritti)

from(

select scuola,count(studente) as iscritti

from iscrizione

group by year(data\_iscrizione),scuola)