



UNIVERSITÀ DI PISA

**CORSO DI LAUREA IN INGEGNERIA INFORMATICA
FONDAMENTI DI INFORMATICA II
MODULO DI BASI DI DATI**

STUDENTI:
PAONE ROBERTO
SEMERARO DANIELE

PROGETTO

Indice

Introduzione	pag. 3
Gestione degli indici	pag. 4
Cardinalità Associazioni	pag. 5
Traduzione delle generalizzazioni	pag. 8
Operazioni.	pag. 9
Ridondanze	pag. 10
Tavola dei volumi	pag. 11
Tavole degli accessi	pag. 13
Analisi costo delle ridondanze	pag. 18
Tabelle	pag. 19
Vincoli di integrità referenziale	pag. 20
Vincoli di integrità generici	pag. 22
Normalizzazione	pag. 23
Script SQL	pag. 27
Popolamento	pag. 38
Operazioni	pag. 51
Ridondanze e vincoli di integrità generici	pag. 54
Area Analytics	pag. 58

Introduzione

Il progetto si pone come scopo la realizzazione di un database relazionale per la gestione dei dati di un'impresa di giardinaggio distribuita sul territorio nazionale. La progettazione è volta a organizzare i dati e memorizzare le informazioni relative alla manutenzione, alla coltivazione delle piante e alla gestione di ogni attività svolta dall'azienda. Tutte le piante presenti nell'azienda sono inserite in un contenitore, il quale possiede un codice univoco. Per mezzo della strumentazione in possesso, può essere garantita una perfetta gestione delle piante coltivate. Tramite le frequenti rilevazioni su ogni pianta, è possibile rintracciare quelle affette da patologie, registrarle nel database e spostarle temporaneamente in quarantena.

Il database permette la creazione di un proprio account, per mezzo del quale è possibile interagire con gli altri utenti accedendo al forum e ordinare online le piante. Ad ogni acquisto online, viene creata una scheda, collegata all'utente che ha effettuato l'ordine, in cui vengono memorizzate le informazioni della pianta acquistata.

Gli utenti hanno la possibilità di richiedere interventi di manutenzione, che vengono affidati al personale competente. Il sistema gestirà automaticamente la tempistica d'intervento.

Infine, un utente può far uso della funzione di Garden Design per tracciare uno spazio, corrispondente al proprio giardino, in cui inserire piante o vasi. Tutti i settori creati, con le relative piante e/o vasi inseriti, verranno salvati nel sistema, così da permettere all'utente di accedere nuovamente al suo spazio in un secondo momento.

Gestione degli indici

Vincolo di conflitto per le piante

Data una pianta, bisogna rispettare una distanza minima da un'altra pianta in base all'indice di accrescimento della stessa, dal tipo di terreno di cui necessita, dal fabbisogno idrico, dal tipo di luce necessaria e dalla specie della pianta.

Un esempio è il Philadelphus o Fior D'angelo, arbusto spogliante, alto 1,80 m e largo 1,50m, pianta a crescita rapida, con apparato radicale vasto, che vive all'aperto. Al sole fiorisce di più che non a mezz'ombra. Pianta che richiede condizioni di temperatura prevalentemente basse e preferisce terreno fertile e ben drenato con irrigazione media durante la stagione estiva. Tale pianta non potrà essere coltivata vicino al Ranuncolo, che richiede un terreno estremamente argilloso (basso livello di drenaggio) e che ha una crescita radicale altrettanto vasta, con estensione radicale di circa un metro.

Indice di irrigazione

Gli attributi riguardanti l'irrigazione possono assumere il valore “Basso”, “Medio”, “Alto”, a seconda che:

- Il valore assunto sarà Alto.
- Se è necessario irrigare la pianta una volta alla settimana, il valore assunto sarà Medio.
- Se è necessario irrigare la pianta una volta l'anno, il valore assunto sarà Basso.
- Se la pianta non ha bisogno di essere irrigata il valore assunto sarà Nullo.

Indice di accrescimento

L'indice di accrescimento di una pianta è un attributo che riguarda la velocità con la quale la pianta raggiunge la sua dimensione massima. Può assumere i valori “molto veloce”, “veloce”, “medio”, “lento” o “molto lento” secondo il seguente criterio:

- “molto veloce”: Tempo di raggiungimento della dimensione massima meno di 3 anni.
- “veloce”: Tempo di raggiungimento della dimensione massima fra 3 e 10 anni.
- “medio”: Tempo di raggiungimento della dimensione massima fra 10 e 30 anni.
- “lento”: Tempo di raggiungimento della dimensione massima fra 30 e 50 anni.
- “molto lento”: Tempo di raggiungimento della dimensione massima oltre i 50 anni.

Indice manutenzione

L'indice di manutenzione è un indice che rappresenta il livello di cure che una pianta richiede nel tempo e può assumere i valori “Difficile”, “Medio” e “Semplice”. Per assegnare l'indice ad una pianta si prendono in considerazione quattro fattori principali:

- Malattie che possono colpire la pianta.
- Interventi richiesti.
- Periodicità di irrigazione richiesta.
- Indice di accrescimento.

Cardinalità associazioni

Associazione Acquisto

- Cardinalità (0,N) con Utente, in quanto un utente può fare o zero o più ordini.
- Cardinalità (1,1) con Ordine, in quanto un ordine appartiene ad un solo utente.

Associazione Appartenenza_1

- Cardinalità (1,N) con Ripiano, in quanto in ogni ripiano possono esserci uno o più contenitori.
- Cardinalità (1,1) con Contenitore, in quanto ogni contenitore è presente in uno e un solo ripiano.

Associazione Appartenenza_2

- Cardinalità (1,N) con Sezione, in quanto ogni sezione è suddivisa in uno o più ripiani .
- Cardinalità (1,1) con Ripiano, in quanto ogni ripiano appartiene ad una e una sola sezione.

Associazione Appartenenza_3

- Cardinalità (1,N) con Serra, in quanto ogni serra è suddivisa in una o più sezioni.
- Cardinalità (1,1) con Sezione, in quanto ogni sezione appartiene ad una e una sola serra.

Associazione Appartenenza_4

- Cardinalità (1,N) con Sede, in quanto ad una sede appartengono una o più serre.
- Cardinalità (1,1) con Serra, in quanto ciascuna serra appartiene a una sola sede.

Associazione Autore

- Cardinalità (0,N) con Utente, in quanto un utente può aver scritto zero o più post.
- Cardinalità (0,1) con Post, in quanto un post ha un solo autore o può non averne nel caso in cui l'autore si sia disiscritto dal forum.

Associazione Confine

- Cardinalità (1,N) con Vertice, in quanto ogni vertice può appartenere ad uno o a più settori.
- Cardinalità (3,N) con Settore, in quanto ogni settore è composto da un minimo di tre vertici e può averne di più.

Associazione Corrispondenza

- Cardinalità (1,1) con Intervento, in quanto un intervento effettivo di manutenzione corrisponde ad una sola tipologia di intervento.
- Cardinalità (0,N) con Esigenza_intervento, in quanto un tipo di intervento può essere richiesto nessuna o più volte.

Associazione Cura

- Cardinalità (0,N) con Patologia, in quanto per una determinata patologia ci possono essere più prodotti indicati. Inoltre, per una patologia, può non esistere un prodotto indicato.
- Cardinalità (1,N) con Prodotto, in quanto ogni prodotto può curare una o più patologie.

Associazione Diagnosi

- Cardinalità (0,1) con Rilevazione, in quanto ad ogni rilevazione può essere presente o meno una patologia.
- Cardinalità (0,N) con Patologia, in quanto ogni patologia può essere presente in zero o più rilevazioni.

Associazione Divisione

- Cardinalità (1,1) con Settore, in quanto ogni settore appartiene ad un solo spazio.
- Cardinalità (1,N) con Spazio, in quanto in quanto ogni spazio è composto da almeno un settore e può essere composto da più settori.

Associazione Esigenza_nutrizionale

- Cardinalità (1,N) con Elemento_nutrizionale, in quanto ogni pianta ha bisogno di uno o più elementi disciolti.
- Cardinalità (1,N) con Pianta, in quanto ogni elemento disciolto è necessario ad una o più piante.

Associazione Esigenza_terreno

- Cardinalità (1,N) con Terreno, in quanto ogni tipologia di terreno è adatta per una o più tipi di piante.
- Cardinalità (1,1) con Pianta, in quanto ogni tipo di pianta esige una sola composizione di terreno.

Associazione Farmacologia

- Cardinalità (1,1) con Trattamento, in quanto ad ogni trattamento viene associato un solo farmaco.
- Cardinalità (0,N) con Prodotto, in quanto ogni prodotto può essere utilizzato in nessuno o più trattamenti.

Associazione Genere

- Cardinalità (1,1) con Contenitore, in quanto ogni contenitore ospita un solo tipo di pianta.
- Cardinalità (0,N) con Pianta, in quanto per ogni genere di pianta possono essere presenti nessuno o più esemplari in azienda.

Associazione Inserimento_1

- Cardinalità (1,1) con Componente_giardino, in quanto ogni componente del giardino può appartenere ad un solo settore.
- Cardinalità (0,N) con Settore, in quanto ad ogni settore possono appartenere zero o più piante.

Associazione Inserimento_2

- Cardinalità (1,1) con Vaso, in quanto ogni vaso appartiene ad un solo settore.
- Cardinalità (0,N) con Settore, poiché in ogni settore possono esserci o zero o più vasi.

Associazione Istanza

- Cardinalità (0,N) con Pianta, in quanto ogni tipo di pianta può essere inserita nessuna o più volte in un giardino.
- Cardinalità (1,1) con Componente_giardino, in quanto ogni componente del giardino rappresenta una sola pianta.

Associazione Manutenzione

- Cardinalità (0,N) con Scheda, in quanto per ogni scheda possono essere richiesti zero o più interventi.
- Cardinalità (1,1) con Intervento, in quanto un intervento riguarda una sola scheda.

Associazione Monitoraggio

- Cardinalità (0,N) con Contenitore, in quanto per ogni contenitore sono salvate zero o più rilevazioni.

- Cardinalità (1,1) con Rilevazione, in quanto ogni rilevazione riguarda un solo contenitore.

Associazione Necessità_1

- Cardinalità (1,N) con Pianta, in quanto ogni genere di pianta ha bisogno di un tipo di luminosità e di irrigazione per ogni periodo dell'anno.
- Cardinalità (1,N) con Esigenza, in quanto un tipo di esigenza può soddisfare una o più piante.

Associazione Necessità_2

- Cardinalità (0,N) con Pianta, in quanto ogni pianta può richiedere più interventi, oppure nessuno.
- Cardinalità (1,N) con Esigenza_intervento, in quanto si suppone che un intervento di manutenzione sia necessario ad almeno una pianta oppure può essere richiesto da più piante.

Associazione Preferenza

- Cardinalità (0,N) con Pianta, in quanto ogni genere di pianta può piacere a nessuno o a più utenti.
- Cardinalità (0,N) con Utente, in quanto ogni utente può avere più preferenze oppure nessuna.

Associazione Progettazione

- Cardinalità (1,1) con Spazio, in quanto ogni spazio è progettato da un solo utente.
- Cardinalità (0,N) con Utente, in quanto un utente può progettare o zero o più spazi.

Associazione Proprietà

- Cardinalità (0,N) con Utente, in quanto un utente può aver acquistato nessuna o più piante.
- Cardinalità (1,1) con Scheda, in quanto ogni scheda appartiene ad un solo utente.

Associazione Quarantena

- Cardinalità (0,N) con Contenitore, in quanto ogni pianta può aver subito o zero o più trattamenti.
- Cardinalità (1,N) con Trattamento, in quanto ogni trattamento viene fatto ad una o a più piante.

Associazione Rappresentazione

- Cardinalità (1,1) con Scheda, in quanto una scheda rappresenta una sola pianta.
- Cardinalità (0,N) con Pianta, in quanto per una pianta possono esserci più schede, oppure nessuna.

Associazione Riferimento_1

- Cardinalità (0,1) con Contenitore, in quanto un contenitore può essere ordinato o meno.
- Cardinalità (0,1) con Ordine, in quanto un ordine può riferirsi ad un contenitore o a nessuno.

Associazione Riferimento_2

- Cardinalità (1,1) con Ordine, in quanto ogni ordine si riferisce ad una pianta
- Cardinalità (0,N) con Pianta, in quanto una pianta può essere oggetto di nessuno o più ordini

Associazione Sintomatologia

- Cardinalità (1,N) con Sintomo, in quanto ogni sintomo corrisponde ad una o più patologie.
- Cardinalità (1,N) con Patologia, in quanto ogni patologia ha uno o più sintomi.

Associazione Suscettibilità

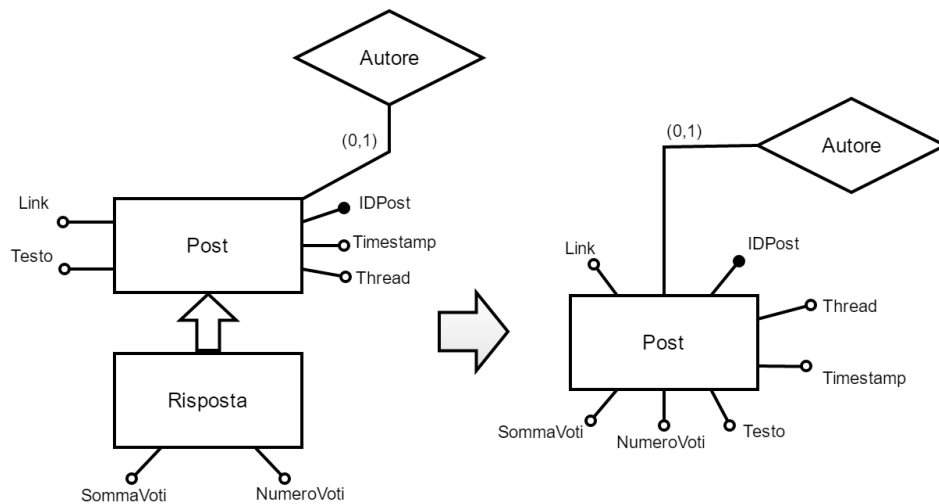
- Cardinalità (0,N) con Pianta, in quanto ogni pianta è suscettibile a zero o più patologie.
- Cardinalità (1,N) con Patologia, in quanto ogni patologia può colpire uno o più generi di piante.

Associazione Visualizzazione

- Cardinalità (0,N) con Sintomo, in quanto per ogni sintomo possono essere presenti zero o più immagini.
- Cardinalità (1,1) con Immagine_sintomo, in quanto ogni immagine raffigura uno e un solo sintomo.

Traduzione delle generalizzazioni

Nello schema E-R è presente una generalizzazione parziale con una sola entità figlia, ovvero un sottoinsieme. Nello specifico l'entità Risposta è figlia dell'entità Post. Abbiamo deciso di tradurre la generalizzazione accorpare l'entità figlia nel padre, aggiungendo, cioè, all'entità Post gli attributi SommaVoti e NumeroVoti, supponendo che per ogni post ci siano in media 15 risposte. Gli attributi SommaVoti e NumeroVoti, quindi, assumono valore NULL un numero di volte trascurabile.



Operazioni

Operazione 1

Descrizione: Calcolare, per una data sezione, il numero di piante che possono essere inserite prima di raggiungere la capienza massima.

Frequenza: 500 volte a settimana.

Operazione 2

Descrizione: Inserimento nel database di un contenitore in un data Sezione.

Frequenza: 12500 volte al giorno. (25 volte per ogni sezione dell'azienda)

Operazione 3

Descrizione: Elencare i costi dei giardini progettati da un dato utente.

Frequenza: 20 volte al giorno.

Operazione 4

Descrizione: Un utente modifica uno spazio per inserire una nuova pianta.

Frequenza: 20 volte al giorno.

Operazione 5

Descrizione: Elencare le rilevazioni di un dato contenitore.

Frequenza: 100 volte al giorno.

Operazione 6

Descrizione: Elencare gli utenti in ordine di credibilità con relative informazioni.

Frequenza: 1 volta al mese.

Operazione 7

Descrizione: Un post riceve una nuova valutazione.

Frequenza: 50 volte al giorno.

Operazione 8

Descrizione: Stilare un elenco degli ordini pendenti.

Frequenza: 1 volta al giorno.

Ridondanze

- **Ridondanza #1 - NumeroPiante**

L'attributo, appartenente all'entità "Sezione", costituisce una ridondanza in quanto mantiene il numero di piante presenti nella sezione. Questo valore si può calcolare anche contando le occorrenze dei contenitori presenti nei ripiani relativi a quella determinata sezione.

- **Ridondanza #2 - Preventivo**

L'attributo, appartenente all'entità "Spazio", consente di fornire ad un utente il costo indicativo di uno spazio da lui progettato. Questo valore può essere calcolato anche sommando i costi indicativi delle singole piante scelte dall'utente relative al medesimo spazio.

- **Ridondanza #3 - Credibilità**

L'attributo, appartenente all'entità "Utente", mantiene informazioni relative alla credibilità acquisita dall'utente. Questo valore può essere ottenuto calcolando la media delle valutazioni dei post scritti dallo stesso.

Tavola dei volumi

NOME	E/R	VOLUME	MOTIVAZIONE
Acquisto	R	6.000	Ogni ordine viene effettuato da un solo utente
Appartenenza_1	R	75.000	Ad ogni contenitore corrisponde un ripiano.
Appartenenza_2	R	1.500	Ad ogni ripiano corrisponde una sezione.
Appartenenza_3	R	500	Ad ogni sezione corrisponde una serra.
Appartenenza_4	R	100	Ad ogni serra corrisponde una sede.
Autore	R	360.000	Ogni post viene inviato da un solo utente.
Componente_giardino	E	144.000	Ogni settore ha in media 3 Componenti_giardino. $48.000 * 3 = 144.000$
Confine	R	192.000	Ogni vertice appartiene in media a 2 settori. $96.000 * 2 = 192.000$
Contenitore	E	75.000	Si suppone che per ogni ripiano ci siano in media 50 contenitori. $1.500 * 50 = 75.000$
Corrispondenza	R	360.000	Ogni intervento corrisponde a una sola tipologia di intervento
Cura	R	300	Ogni prodotto cura in media 3 patologie. $100 * 3 = 300$
Diagnosi	R	1.875	Supponiamo che solo una rilevazione su 200 presenti una patologia. $375000 : 200 = 1875$
Divisione	R	24.000	Ogni settore corrisponde ad un solo spazio.
Elemento_nutrizionale	E	16	Gli elementi essenziali per tutte le specie vegetali sono 16.
Esigenza_ambientale	E	1.000	Ipotesi.
Esigenza_intervento	E	500	Ipotesi.
Esigenza_nutrizionale	R	40.000	Ogni pianta ha bisogno in media di 4 elementi nutritivi.
Esigenza_terreno	R	10.000	Ogni specie di pianta ha bisogno di un tipo di terreno
Farmacologia	R	375	Ogni trattamento viene applicato con un solo farmaco.
Genere	R	75.000	Ogni pianta presente in azienda ha un solo genere di appartenenza
Immagine_sintomo	E	600	Per ogni sintomo ci sono in media 2 immagini. $300 * 2 = 600$
Inserimento_1	R	72.000	Ogni componente corrisponde ad un solo settore
Inserimento_2	R	24.000	Ogni vaso appartiene ad un solo settore.
Intervento	E	360.000	Per ogni scheda sono stati effettuati in media 5 interventi. $72.000 * 5 = 360.000$

NOME	E/R	VOLUME	MOTIVAZIONE
Istanza	R	72.000	Ogni componente corrisponde ad un'istanza di Pianta.
Manutenzione	R	360.000	Ogni intervento riguarda una sola scheda.
Monitoraggio	R	750.000	Ogni rilevazione corrisponde ad un contenitore presente in azienda
Necessità_1	R	17.500	Supponiamo che 3/4 delle piante abbiano due tipi di esigenze diverse in base al periodo e il restante un solo tipo.
Necessità_2	R	30.000	Ogni tipo di pianta ha bisogno in media di 3 tipologie di intervento. $10.000 * 3 = 30.000$
Ordine	E	6.000	Supponiamo che ci sia una media di un ordine ogni 2 utenti. $12.000 : 2 = 6.000$
Patologia	E	750	Ipotesi.
Pianta	E	10.000	Ipotesi sui tipi di piante salvate nel database dell'azienda.
Post	E	360.000	Supponiamo che ogni utente scriva in media 30 post. $12.000 * 30 = 360.000$
Preferenza	R	18.000	Un utente ha in media 1,5 piante preferite. $12.000 * 1,5 = 18.000$
Prodotto	E	100	Ipotesi
Progettazione	R	4.000	Ogni spazio viene progettato da un solo utente.
Proprietà	R	72.000	Ogni scheda corrisponde ad un solo utente.
Quarantena	R	1.750	Ogni trattamento viene applicato in media a 5 piante. $350 * 5 = 1.750$
Rappresentazione	R	72.000	Ogni scheda si riferisce ad un tipo di pianta.
Riferimento_1	R	4.500	3/4 degli ordini si riferiscono a piante presenti in azienda. $6.000 * 3/4 = 4.500$
Riferimento_2	R	6.000	Ogni ordine si riferisce ad una pianta.
Rilevazione	E	750.000	Si suppone che siano salvate in media 10 rilevazioni per ogni contenitore. $75.000 * 10 = 750.000$
Ripiano	E	1.500	Si suppone che per ogni sezione ci siano in media 3 ripiani. $500 * 3 = 1.500$
Scheda	E	72.000	In media ogni utente ha acquistato dall'azienda 6 piante. $12.000 * 6 = 72.000$
Sede	E	10	Ipotesi iniziale
Serra	E	100	Si suppone che in ogni sede abbia in media 10 serre. $10 * 10 = 100$

NOME	E/R	VOLUME	MOTIVAZIONE
Settore	E	48.000	In media ci sono 4 settori per ogni spazio. $12.000 * 4 = 48.000$
Sezione	E	500	Si suppone che per ogni serra ci siano in media 5 sezioni. $100 * 5 = 500$
Sintomatologia	R	2.250	Ogni patologia presenta in media 4 sintomi. $750 * 3 = 2250$
Sintomo	E	300	Ipotesi
Spazio	E	12.000	Ad ogni utente corrisponde in media 1 spazio
Suscettibilità	R	20.000	Supponiamo che ogni pianta sia suscettibile in media a 3 malattie. $10.000 * 2 = 20.000$
Terreno	E	12	Principali tipi di terreno adatti alla coltivazione.
Trattamento	E	375	Supponiamo che ci sia una media di un trattamento ogni 200 piante presenti nell'azienda. $75.000 : 200 = 375$
Utente	E	12.000	Ipotesi iniziale
Vaso	E	48.000	Ogni settore ha in media 1 vaso.
Vertice	E	192.000	Ogni settore ha in media 4 vertici. $48.000 * 4 = 192.000$
Visualizzazione	R	600	Ogni immagine corrisponde a un solo sintomo.

Tavole degli accessi

Operazione 1: Calcolare, per una data sezione, il numero di piante che possono essere inserite prima di raggiungere la capienza massima.

La tavola degli accessi di questa operazione cambia a seconda che sia presente o meno l'attributo NumeroPiante dell'entità Sezione (Ridondanza 1).

Senza Ridondanza 1		
Num. Operazioni elementari	Tipo Operazioni	Motivazione
1	L	Compio 1 operazione di lettura su Sezione per trovare la capienza massima della sezione
3	L	Compio 3 operazioni di lettura su Appartenenza_2 per trovare i ripiani che appartengono alla sezione
150	L	Compio 150 operazioni di lettura su Appartenenza_1 per contare i contenitori che appartengono alla sezione (Ogni ripiano contiene in media 50 contenitori)
154	Totale operazioni elementari per singola operazione	
154 x 500 = 77000	Totale operazioni elementari alla settimana	

Con Ridondanza 1		
Num. Operazioni elementari	Tipo Operazioni	Motivazione
1	L	Compio 1 operazione di lettura su Sezione per trovare la capienza massima della sezione e il numero di piante attualmente ospitate
1	Totale operazioni elementari per singola operazione	
1 x 500	Totale operazioni elementari alla settimana	

Operazione 2: Inserimento nel database di un contenitore in un data sezione.

La tavola degli accessi di questa operazione cambia a seconda che sia presente o meno l'attributo NumeroPiante dell'entità Sezione (Ridondanza 1).

Senza Ridondanza 1		
Num. Operazioni elementari	Tipo Operazioni	Motivazione
3	L	Compio 3 operazioni di lettura su Appartenenza_2 per recuperare il codice dei ripiani che appartengono alla sezione
1 x 2 = 2	S	Compio 1 operazione di scrittura su Contenitore per inserire le informazioni relative al contenitore
5	Totale operazioni elementari per singola operazione	
5 x 12500 = 62500	Totale operazioni elementari al giorno	

Con Ridondanza 1		
Num. Operazioni elementari	Tipo Operazioni	Motivazione
3	L	Compio 3 operazioni di lettura su Appartenenza_2 per recuperare il codice dei ripiani che appartengono alla sezione
1 x 2 = 2	S	Compio 1 operazione di scrittura su Contenitore per inserire le informazioni relative al contenitore
1 x 2 = 2	S	Compio 1 operazione di scrittura su Sezione per aggiornare l'attributo NumeroPiante
7	Totale operazioni elementari per singola operazione	
7 x 12500 = 87500	Totale operazioni elementari al giorno	

Operazione 3: Elencare i costi dei giardini progettati da un dato utente.

La tavola degli accessi di questa operazione cambia a seconda che sia presente o meno l'attributo Preventivo dell'entità Spazio (Ridondanza 2).

Senza Ridondanza 2		
Num. Operazioni elementari	Tipo Operazioni	Motivazione
1	L	Compio 1 operazione di lettura su Progettazione per recuperare il codice dello spazio progettato dall'utente
4	L	Compio 4 operazioni di lettura su Divisione per recuperare il codice dei settori relativi allo spazio
12	L	Compio 12 operazioni di lettura su Inserimento_1 per recuperare il codice dei componenti dello spazio
12	L	Compio 12 operazioni di lettura su Pianta per recuperare le informazioni sui costi delle piante che compongono lo spazio
29	Totale operazioni elementari per singola operazione	
29 x 20 = 580	Totale operazioni elementari al giorno	

Con Ridondanza 2		
Num. Operazioni elementari	Tipo Operazioni	Motivazione
1	L	Compio 1 operazione di lettura su Progettazione per recuperare il codice dello spazio progettato dall'utente
1	L	Compio 1 operazione di lettura su Spazio per recuperare il costo relativo allo Spazio
2	Totale operazioni elementari per singola operazione	
2 x 20 = 40	Totale operazioni elementari al giorno	

Operazione 4: Un utente modifica uno spazio per inserire una nuova pianta.

La tavola degli accessi di questa operazione cambia a seconda che sia presente o meno l'attributo Preventivo dell'entità Spazio (Ridondanza 2).

Senza Ridondanza 2		
Num. Operazioni elementari	Tipo Operazioni	Motivazione
1 x 2	S	Compio 1 operazione di Scrittura su Componente_giardino per inserire le informazioni riguardanti la nuova pianta
2	Totale operazioni elementari per singola operazione	
2 x 20 = 40	Totale operazioni elementari al giorno	

Con Ridondanza 2		
Num. Operazioni elementari	Tipo Operazioni	Motivazione
$1 \times 2 = 2$	S	Compio 1 operazione di Scrittura su Componente_giardino per inserire le informazioni riguardanti la nuova pianta
1	L	Compio 1 operazione di lettura su Pianta per ottenere il Costo della nuova pianta
1	L	Compio 1 operazione di lettura su Inserimento_1 per recuperare il codice del settore in cui è inserita la pianta
1	L	Compio 1 operazione di lettura su Divisione per Recuperare il codice dello Spazio a cui appartiene il Settore
$1 \times 2 = 2$	S	Compio 1 operazione di scrittura su Spazio per aggiornare l'attributo Preventivo
7	Totale operazioni elementari per singola operazione	
$7 \times 20 = 140$	Totale operazioni elementari al giorno	

Operazione 5: Elencare le rilevazioni di un dato contenitore.

L'operazione non coinvolge ridondanze.

Num. Operazioni elementari	Tipo Operazioni	Motivazione
10	L	Compio 10 operazioni di lettura su Rilevazione per trovare le informazioni sulle rilevazioni associate al contenitore
10	Totale operazioni elementari per singola operazione	
$10 \times 100 = 1000$	Totale operazioni elementari al giorno	

Operazione 6: Elencare gli utenti in ordine di credibilità con relative informazioni.

La tavola degli accessi di questa operazione cambia a seconda che sia presente o meno l'attributo Credibilità dell'entità Utente (Ridondanza 3).

Senza Ridondanza 3		
Num. Operazioni elementari	Tipo Operazioni	Motivazione
360000	L	Leggo tutte le istanze di Post per recuperare le informazioni riguardanti i voti di ogni post scritti da ogni utente
12000	L	Leggo tutte le istanze di Utente per recuperare le informazioni riguardanti gli utenti del forum
372000	Totale operazioni elementari per singola operazione	
$372000 \times 1 = 372000$	Totale operazioni elementari al mese	

Con Ridondanza 3		
Num. Operazioni elementari	Tipo Operazioni	Motivazione
12000	L	Leggo tutte le istanze di Utente per recuperare le informazioni riguardanti gli utenti del forum
12000	Totale operazioni elementari per singola operazione	
$12000 \times 1 = 12000$	Totale operazioni elementari al mese	

Operazione 7: Un post riceve una nuova valutazione.

La tavola degli accessi di questa operazione cambia a seconda che sia presente o meno l'attributo Credibilità dell'entità Utente (Ridondanza 3).

Senza Ridondanza 3		
Num. Operazioni elementari	Tipo Operazioni	Motivazione
$1 \times 2 = 2$	S	Compio 1 operazione di scrittura su Post per aggiornare le informazioni riguardanti il voto
2	Totale operazioni elementari per singola operazione	
$2 \times 50 = 100$	Totale operazioni elementari al giorno	

Con Ridondanza 3		
Num. Operazioni elementari	Tipo Operazioni	Motivazione
$1 \times 2 = 2$	S	Compio 1 operazione di scrittura su Post per aggiornare le informazioni riguardanti il voto
1	L	Compio 1 operazione di lettura su Autore per ottenere il Nickname dell'autore del post
30	L	Compio 30 operazioni di lettura su Autore Post per ottenere le informazioni sui voti dei post scritti dall'utente del post
$1 \times 2 = 2$	S	Compio 1 operazione di scrittura su Utente per aggiornare l'attributo Credibilità
35	Totale operazioni elementari per singola operazione	
$35 \times 50 = 150$	Totale operazioni elementari al giorno	

Operazione 8: Stilare un elenco degli ordini in stato pendente e le informazioni sulle piante richieste.

L'operazione non coinvolge ridondanze.

Num. Operazioni elementari	Tipo Operazioni	Motivazione
1200*	L	Compio 1200 operazioni di lettura su Ordine per ottenere il codice degli ordini in stato pendente
1200	L	Compio 1200 operazioni di lettura su Pianta per ottenere le informazioni riguardanti le piante richieste
2400	Totale operazioni elementari per singola operazione	
$2400 \times 1 = 2400$	Totale operazioni elementari al giorno	

*Si è stimato che il 20% degli ordini totali sia in stato pendente.

Analisi costo ridondanze

Ridondanza 1:

Operazione	Numero operazioni elementari al giorno	
	Senza ridondanza	Con ridondanza
Operazione 1	77000	1
Operazione 2	62500	87500
Totale	139500	0

Quindi Ridondanza 1 riduce, in media, i tempi di esecuzione delle operazioni.

Ridondanza 2:

Operazione	Numero operazioni elementari al giorno	
	Senza ridondanza	Con ridondanza
Operazione 3	580	40
Operazione 4	40	140
Totale	620	180

Quindi Ridondanza 2 riduce, in media, i tempi di esecuzione delle operazioni.

Ridondanza 3:

Operazione	Numero operazioni elementari al giorno	
	Senza ridondanza	Con ridondanza
Operazione 6	372000	12000
Operazione 7	100	150
Totale	372100	12150

Quindi Ridondanza 3 riduce, in media, i tempi di esecuzione delle operazioni.

Progettazione logica

Tabelle

Entità

- **Componente_giardino** (IDComponente, CoordinataX, CoordinataY, Raggio, IDPianta, IDSettore)
- **Contenitore** (IDContenitore, DimensioneContenitore, AltezzaPianta, Prezzo, IDPianta, IDRipiano)
- **Elemento_nutrizionale** (Simbolo, NomeElemento, Tipologia, Descrizione)
- **Esigenza_ambientale** (IDEsigenzaAmbientale, TipoEsposizione, OreDiLuce, PeriodicitàIrrigazione, Fabbisognoldrico, TemperaturaMassima, TemperaturaMinima)
- **Esigenza_intervento** (IDEsigenzaIntervento, NomeIntervento, Descrizione)
- **Immagine_sintomo** (IDImmagine, Link, IDSintomo)
- **Intervento** (IDIntervento, TipoIntervento, Indirizzo, Costo, Data, Ricorrenza, IDEsigenzaIntervento, IDScheda)
- **Ordine** (IDOrdine, TimeStamp, Stato, Utente, Indirizzo, Destinatario, Pianta, IDContenitore)
- **Patologia** (IDPatologia, NomePatologia, AgentePatogeno)
- **Pianta** (IDPianta, Nome, Genere, Cultivar, AltezzaMassima, LarghezzaMassima, IndiceAccrescimento, PeriodoFioritura, PeriodoFruttificazione, PeriodoVegetativo, Sempreverde, Dioica, IndiceManutenzione, PrezzoIndicativo, IDTerreno)
- **Post** (IDPost, Autore, Timestamp, Thread, Testo, SommaVoti, NumeroVoti, Link)
- **Prodotto** (IDProdotto, NomeCommerciale, PrincipiAttivi, Formulazione, ModalitàSomministrazione)
- **Rilevazione** (IDContenitore, Data, Umidità, Esalazione, ConcentrazioneElementi, IDPatologia)
- **Ripiano** (IDRipiano, Irrigazione, IDSezione)
- **Scheda** (IDScheda, Dimensione, AttualeCollocazione, DimensioneVaso, IDPianta, Utente)
- **Sede** (IDSede, NomeSede, Indirizzo, Città, NumeroDipendenti)
- **Serra** (IDSerra, NomeSerra, Indirizzo, Lunghezza, Altezza, Larghezza, IDSede)
- **Settore** (IDSettore, Coltivabile, TipoLuce, OreLuceGiornaliere, EsposizioneCardinale, IDSpazio)
- **Sezione** (IDSezione, NomeSezione, CapienzaMassima, NumeroPiante, Illuminazione, Umidità, Temperatura, IDSerra)
- **Sintomo** (IDSintomo, NomeSintomo, Descrizione)

- **Spazio** (IDSpazio, Preventivo, Utente)
- **Terreno** (IDTerreno, pH, Consistenza, Permeabilità)
- **Trattamento** (IDTrattamento, IDProdotto, Dosaggio, DataInizioTrattamento, DataFineTrattamento)
- **Utente** (Nickname, Nome, Cognome, Password, Email, CittàDiResidenza, DomandaSegreta, RispostaSegreta, Credibilità)
- **Vaso** (IDVaso, Materiale, CoordinataX, CoordinataY, Raggio, IDSettore)
- **Vertice** (IDVertice, CoordinataX, CoordinataY)

Relazioni

- **Confine** (IDVertice, IDSettore)
- **Cura** (IDPatologia, IDProdotto, Dosaggio)
- **Esigenza_nutrizionale** (IDPianta, Elemento, Concentrazione)
- **Necessità_1** (IDEsigenzaAmbientale, IDPianta, Periodo)
- **Necessità_2** (IDEsigenzaIntervento, IDPianta, Periodo, Frequenza)
- **Preferenza** (Utente, IDPianta)
- **Quarantena** (IDContenitore, IDTrattamento)
- **Sintomatologia** (IDPatologia, IDSintomo)
- **Suscettibilità** (IDPianta, IDPatologia, Periodo, Probabilità, Entità)

Vincoli di integrità referenziale

- Esiste un vincolo di integrità referenziale tra l'attributo IDPianta della tabella Componente_giardino e l'attributo IDPianta della tabella Pianta.
- Esiste un vincolo di integrità referenziale tra l'attributo IDSettore della tabella Componente_giardino e l'attributo IDSettore della tabella Settore.
- Esiste un vincolo di integrità referenziale tra l'attributo IDPianta della tabella Contenitore e l'attributo IDPianta della tabella Pianta.
- Esiste un vincolo di integrità referenziale tra l'attributo IDRipiano della tabella Contenitore e l'attributo IDRipiano della tabella Ripiano.
- Esiste un vincolo di integrità referenziale tra l'attributo IDSintomo della tabella Immagine_sintomo e l'attributo IDSintomo della tabella Sintomo.

- Esiste un vincolo di integrità referenziale tra l'attributo IDEsigenzaIntervento della tabella Intervento e l'attributo IDEsigenzaIntervento della tabella Esigenza_intervento.
- Esiste un vincolo di integrità referenziale tra l'attributo IDScheda della tabella Intervento e l'attributo IDScheda della tabella Scheda.
- Esiste un vincolo di integrità referenziale tra l'attributo Utente della tabella Ordine e l'attributo Nickname della tabella Utente.
- Esiste un vincolo di integrità referenziale tra l'attributo IDContenitore della tabella Ordine e l'attributo IDContenitore della tabella Contenitore.
- Esiste un vincolo di integrità referenziale tra l'attributo IDTerreno della tabella Pianta e l'attributo IDTerreno della tabella Terreno.
- Esiste un vincolo di integrità referenziale tra l'attributo Autore della tabella Post e l'attributo Nickname della tabella Utente.
- Esiste un vincolo di integrità referenziale tra l'attributo IDContenitore della tabella Rilevazione e l'attributo IDContenitore della tabella Contenitore.
- Esiste un vincolo di integrità referenziale tra l'attributo IDPatologia della tabella Rilevazione e l'attributo IDPatologia della tabella Patologia.
- Esiste un vincolo di integrità referenziale tra l'attributo IDSezione della tabella Ripiano e l'attributo IDSezione della tabella Sezione.
- Esiste un vincolo di integrità referenziale tra l'attributo IDPianta della tabella Scheda e l'attributo IDPianta della tabella Pianta.
- Esiste un vincolo di integrità referenziale tra l'attributo Utente della tabella Scheda e l'attributo Nickname della tabella Utente.
- Esiste un vincolo di integrità referenziale tra l'attributo IDSede della tabella Serra e l'attributo IDSede della tabella Sede.
- Esiste un vincolo di integrità referenziale tra l'attributo IDSpazio della tabella Settore e l'attributo IDSpazio della tabella Spazio.
- Esiste un vincolo di integrità referenziale tra l'attributo IDSerra della tabella Sezione e l'attributo IDSerra della tabella Serra.
- Esiste un vincolo di integrità referenziale tra l'attributo Utente della tabella Spazio e l'attributo Nickname della tabella Utente.
- Esiste un vincolo di integrità referenziale tra l'attributo IDProdotto della tabella Trattamento e l'attributo IDProdotto della tabella Prodotto.
- Esiste un vincolo di integrità referenziale tra l'attributo IDProdotto della tabella Trattamento e l'attributo IDProdotto della tabella Prodotto.
- Esiste un vincolo di integrità referenziale tra l'attributo IDSettore della tabella Vaso e l'attributo IDSettore della tabella Settore.
- Esiste un vincolo di integrità referenziale tra l'attributo IDVertice della tabella Confine e l'attributo IDVertice della tabella Vertice.
- Esiste un vincolo di integrità referenziale tra l'attributo IDSettore della tabella Confine e l'attributo

IDSettore della tabella Settore.

- Esiste un vincolo di integrità referenziale tra l'attributo IDPatologia della tabella Cura e l'attributo IDPatologia della tabella Patologia.
- Esiste un vincolo di integrità referenziale tra l'attributo IDProdotto della tabella Cura e l'attributo IDProdotto della tabella Prodotto.
- Esiste un vincolo di integrità referenziale tra l'attributo IDPianta della tabella Esigenza_nutrizionale e l'attributo IDPianta della tabella Pianta.
- Esiste un vincolo di integrità referenziale tra l'attributo Elemento della tabella Esigenza_nutrizionale e l'attributo Simbolo della tabella Elemento_nutrizionale.
- Esiste un vincolo di integrità referenziale tra l'attributo IDEsigenzaAmbientale della tabella Necessità_1 e l'attributo IDEsigenzaAmbientale della tabella Esigenza_ambientale.
- Esiste un vincolo di integrità referenziale tra l'attributo IDPianta della tabella Necessità_1 e l'attributo IDPianta della tabella Pianta.
- Esiste un vincolo di integrità referenziale tra l'attributo IDEsigenzaIntervento della tabella Necessità_2 e l'attributo IDEsigenzaIntervento della tabella Esigenza_intervento.
- Esiste un vincolo di integrità referenziale tra l'attributo IDPianta della tabella Necessità_2 e l'attributo IDPianta della tabella Pianta.
- Esiste un vincolo di integrità referenziale tra l'attributo Utente della tabella Preferenza e l'attributo Nickname della tabella Utente.
- Esiste un vincolo di integrità referenziale tra l'attributo IDPianta della tabella Preferenza e l'attributo IDPianta della tabella Pianta.
- Esiste un vincolo di integrità referenziale tra l'attributo IDContenitore della tabella Quarantena e l'attributo IDContenitore della tabella Contenitore.
- Esiste un vincolo di integrità referenziale tra l'attributo IDTrattamento della tabella Quarantena e l'attributo IDTrattamento della tabella Trattamento.
- Esiste un vincolo di integrità referenziale tra l'attributo IDPatologia della tabella Sintomatologia e l'attributo IDPatologia della tabella Patologia.
- Esiste un vincolo di integrità referenziale tra l'attributo IDSintomo della tabella Sintomatologia e l'attributo IDSintomo della tabella Sintomo.
- Esiste un vincolo di integrità referenziale tra l'attributo IDPianta della tabella Suscettibilità e l'attributo IDPianta della tabella Pianta.
- Esiste un vincolo di integrità referenziale tra l'attributo IDPatologia della tabella Suscettibilità e l'attributo IDPatologia della tabella Patologia.

Vincoli di integrità generici

- Non può essere inserita una pianta in una sezione se quest'ultima ha raggiunto la capienza massima.
- Una pianta non può essere sottoposta contemporaneamente a più trattamenti.

Normalizzazione

Tabella Componente_giardino:

Componente_giardino (IDComponente, CoordinataX, CoordinataY, Raggio, IDPianta, IDSettore)

- IDComponente → CoordinataX, CoordinataY, Raggio, IDPianta, IDSettore

Poiché la parte sinistra è superchiave, Componente_giardino è in BCNF.

Tabella Confine:

Confine (IDVertice, IDSettore)

Poiché non esistono dipendenze funzionali, Confine è in BCNF.

Tabella Contenitore:

Contenitore (IDContenitore, DimensioneContenitore, AltezzaPianta, IDPianta, IDRipiano)

- IDContenitore → DimensioneContenitore, AltezzaPianta, IDPianta, IDRipiano

Poiché la parte sinistra è superchiave, Contenitore è in BCNF.

Tabella Cura:

Cura (IDPatologia, IDProdotto, Dosaggio)

- IDPatologia, IDProdotto → Dosaggio

Poiché la parte sinistra è superchiave, Cura è in BCNF.

Tabella Elemento_nutrizionale:

Elemento_nutrizionale (Simbolo, NomeElemento, Tipologia, Descrizione)

- Simbolo → NomeElemento, Tipologia, Descrizione

Poiché la parte sinistra è superchiave, Elemento_nutrizionale è in BCNF.

Tabella Esigenza_ambientale:

Esigenza_ambientale (IDEsigenzaAmbientale, TipoEsposizione, OreDiLuce, PeriodicitàIrrigazione, Fabbisognoldrico, TemperaturaMassima, TemperaturaMinima)

- IDEsigenzaAmbientale → TipoEsposizione, OreDiLuce, PeriodicitàIrrigazione, Fabbisognoldrico, TemperaturaMassima, TemperaturaMinima

Poiché la parte sinistra è superchiave, Esigenza_ambientale è in BCNF.

Tabella Esigenza_intervento:

Esigenza_intervento (IDEsigenzaIntervento, NomeIntervento, Descrizione)

- IDEsigenzaIntervento → NomeIntervento, Descrizione

Poiché la parte sinistra è superchiave, Esigenza_intervento è in BCNF.

Tabella Esigenza_nutrizionale:

Esigenza_nutrizionale (IDPianta, Elemento, Concentrazione)

- IDPianta, Elemento → Concentrazione

Poiché la parte sinistra è superchiave, Esigenza_nutrizionale è in BCNF.

Tabella Immagine_sintomo:

Immagine_sintomo (IDImmagine, Link, IDSintomo)

- IDImmagine → Link, IDSintomo

Poiché la parte sinistra è superchiave, Immagine_sintomo è in BCNF.

Tabella Intervento:

Intervento (IDIntervento, TipoIntervento, Indirizzo, Costo, Data, Ricorrenza, IDEsigenzaIntervento, IDScheda)

- IDIntervento → Indirizzo, TipoIntervento, Costo, Data, Ricorrenza, IDEsigenzaIntervento, IDScheda

Poiché la parte sinistra è superchiave, Intervento è in BCNF.

Tabella Necessità_1:

Necessità_1 (IDESigenzaAmbientale, IDPianta, Periodo)

- IDEsigenzaAmbientale, IDPianta → Periodo

Poiché la parte sinistra è superchiave, Necessità_1 è in BCNF.

Tabella Necessità_2:

Necessità_2 (IDESigenzaIntervento, IDPianta, Periodo, Frequenza)

- IDEsigenzaIntervento, IDPianta → Periodo, Frequenza

Poiché la parte sinistra è superchiave, Necessità_2 è in BCNF.

Tabella Ordine:

Ordine (IDOrdine, TimeStamp, Stato, Utente, Indirizzo, Destinatario, Pianta, IDContenitore)

- IDOrdine → TimeStamp, Stato, Utente, Indirizzo, Destinatario, Pianta, IDContenitore

Poiché la parte sinistra è superchiave, Ordine è in BCNF.

Tabella Patologia:

Patologia (IDPatologia, NomePatologia, AgentePatogeno)

- IDPatologia → NomePatologia, AgentePatogeno

Poiché la parte sinistra è superchiave, Patologia è in BCNF.

Tabella Pianta:

Pianta (IDPianta, Nome, Genere, Cultivar, AltezzaMassima, LarghezzaMassima, IndiceAccrescimento, PeriodoFioritura, PeriodoFruttificazione, PeriodoVegetativo, Sempreverde, Dioica, IndiceManutenzione, PrezzoIndicativo, IDTerreno)

- IDPianta → Nome, Genere, Cultivar, AltezzaMassima, LarghezzaMassima, IndiceAccrescimento, PeriodoFioritura, PeriodoFruttificazione, PeriodoVegetativo, Sempreverde, Dioica, IndiceManutenzione, PrezzoIndicativo, IDTerreno

Poiché la parte sinistra è superchiave, Generalità_pianta è in BCNF.

Tabella Post:

Post (IDPost, Autore, Timestamp, Thread, Testo, SommaVoti, NumeroVoti, Link)

- IDPost → Autore, Timestamp, Thread, Testo, SommaVoti, NumeroVoti, Link

Poiché la parte sinistra è superchiave, Post è in BCNF.

Tabella Preferenza:

Preferenza (Utente, IDPianta)

Poiché non esistono dipendenze funzionali, Preferenza è in BCNF.

Tabella Prodotto:

Prodotto (IDProdotto, NomeCommerciale, PrincipiAttivi, Formuazione, ModalitàSomministrazione)

- IDProdotto → NomeCommerciale, PrincipiAttivi, Formulazione, ModalitàSomministrazione

Poiché la parte sinistra è superchiave, Prodotto è in BCNF.

Tabella Quarantena:

Quarantena (IDContenitore, IDTrattamento)

Poiché non esistono dipendenze funzionali, Quarantena è in BCNF.

Tabella Rilevazione:

Rilevazione (IDContenitore, Data, Umidità, Esalazione, ConcentrazioneElementi, IDPatologia)

- IDContenitore, Data → Umidità, Esalazione, ConcentrazioneElementi, IDPatologia

Poiché la parte sinistra è superchiave, Rilevazione è in BCNF.

Tabella Ripiano:

Ripiano (IDRipiano, Irrigazione, IDSezione)

- IDRipiano → Irrigazione, IDSezione

Poiché la parte sinistra è superchiave, Ripiano è in BCNF.

Tabella Scheda:

Scheda (IDScheda, Dimensione, AttualeCollocazione, DimensioneVaso, IDPianta, Utente)

- IDScheda → Dimensione, AttualeCollocazione, DimensioneVaso, IDPianta, Utente

Poiché la parte sinistra è superchiave, Scheda è in BCNF.

Tabella Sede:

Sede (IDSede, NomeSede, Indirizzo, Città, NumeroDipendenti)

- IDSede → NomeSede, Indirizzo, Città, NumeroDipendenti

Poiché la parte sinistra è superchiave, Sede è in BCNF.

Tabella Serra:

Serra (IDSerra, NomeSerra, Indirizzo, Lunghezza, Altezza, Larghezza, IDSede)

- IDSerra → NomeSerra, Indirizzo, Lunghezza, Altezza, Larghezza, IDSede

Poiché la parte sinistra è superchiave, Serra è in BCNF.

Tabella Settore:

Settore (IDSettore, Coltivabile, TipoLuce, OreLuceGiornaliere, EsposizioneCardinale, IDSpazio)

- IDSettore → Coltivabile, TipoLuce, OreLuceGiornaliere, EsposizioneCardinale, IDSpazio

Poiché la parte sinistra è superchiave, Settore è in BCNF.

Tabella Sezione:

Sezione (IDSezione, NomeSezione, CapienzaMassima, NumeroPiante, Illuminazione, Umidità, Temperatura, IDSerra)

- IDSezione → NomeSezione, CapienzaMassima, , NumeroPiante, Illuminazione, Umidità, Temperatura, IDSerra

Poiché la parte sinistra è superchiave, Sezione è in BCNF.

Tabella Sintomatologia:

Sintomatologia (IDPatologia, IDSintomo)

Poiché non esistono dipendenze funzionali, Sintomatologia è in BCNF.

Tabella Sintomo:

Sintomo (IDSintomo, NomeSintomo, Descrizione)

- IDSintomo → NomeSintomo, Descrizione

Poiché la parte sinistra è superchiave, Sintomo è in BCNF.

Tabella Spazio:

Spazio (IDSpazio, Preventivo, Utente)

- IDSpazio → Preventivo, Utente

Poiché la parte sinistra è superchiave, Spazio è in BCNF.

Tabella Suscettibilità:

Suscettibilità (IDPianta, IDPatologia, Periodo, Probabilità, Entità)

- IDPianta, IDPatologia → Periodo, Probabilità, Entità

Poiché la parte sinistra è superchiave, Suscettibilità è in BCNF.

Tabella Terreno:

Terreno (IDTerreno, pH, Consistenza, Permeabilità)

- IDTerreno → pH, Consistenza, Permeabilità

Poiché la parte sinistra è superchiave, Terreno è in BCNF.

Tabella Trattamento:

Trattamento (IDTrattamento, IDProdotto, Dosaggio, DataInizioTrattamento, DataFineTrattamento)

- IDTrattamento → IDProdotto, Dosaggio, DataInizioTrattamento, DataFineTrattamento

Poiché la parte sinistra è superchiave, Trattamento è in BCNF.

Tabella Utente:

Utente (Nickname, Nome, Cognome, Password, Email, CittàDiResidenza, DomandaSegreta, RispostaSegreta, Credibilità)

- Nickname → Nome, Cognome, Password, Email, CittàDiResidenza, DomandaSegreta, RispostaSegreta, Credibilità

Poiché la parte sinistra è superchiave, Utente è in BCNF.

Tabella Vaso:

Vaso (IDVaso, Materiale, CoordinataX, CoordinataY, Raggio, IDSettore)

- IDVaso → Materiale, CoordinataX, CoordinataY, Raggio, IDSettore

Poiché la parte sinistra è superchiave, Vaso è in BCNF.

Tabella Vertice:

Vertice (IDVertice, CoordinataX, CoordinataY)

- IDVertice → CoordinataX, CoordinataY

Poiché la parte sinistra è superchiave, Vertice è in BCNF.

Script SQL

Creazione Database

```
DROP DATABASE Progetto;
```

```
CREATE DATABASE Progetto;
```

```
USE Progetto;
```

```
-- ENTITA'
```

```
-- Tabella Terreno
```

```
DROP TABLE IF EXISTS Terreno;
```

```
CREATE TABLE Terreno(  
    IDTerreno INT UNSIGNED NOT NULL AUTO_INCREMENT,  
    pH FLOAT NOT NULL,  
    Consistenza VARCHAR(50) NOT NULL,  
    Permeabilita VARCHAR(50) NOT NULL,  
    PRIMARY KEY (IDTerreno)  
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;
```

```
-- Tabella Pianta
```

```
DROP TABLE IF EXISTS Pianta;
```

```
CREATE TABLE Pianta(  
    IDPianta INT UNSIGNED NOT NULL AUTO_INCREMENT,  
    Nome VARCHAR(50) NOT NULL,  
    Genere VARCHAR(50) NOT NULL,  
    Cultivar VARCHAR(50),  
    AltezzaMassima FLOAT NOT NULL,  
    LarghezzaMassima FLOAT NOT NULL,  
    IndiceAccrescimento VARCHAR(50) NOT NULL,
```

```

PeriodoFioritura VARCHAR(50),
PeriodoFruttificazione VARCHAR(50),
PeriodoVegetativo VARCHAR(50) NOT NULL,
SempreVerde CHAR(2) NOT NULL,
Dioica CHAR(2) NOT NULL,
IndiceManutenzione VARCHAR(50) NOT NULL,
PrezzoIndicativo SMALLINT UNSIGNED NOT NULL,
IDTerreno INT UNSIGNED NOT NULL,
UNIQUE (Nome, Genere, Cultivar),
PRIMARY KEY (IDPianta),
FOREIGN KEY (IDTerreno) REFERENCES Terreno(IDTerreno) ON DELETE NO ACTION ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-- -----
-- Tabella Elemento_nutrizionale
-- -----

```

```

DROP TABLE IF EXISTS Elemento_nutrizionale;
CREATE TABLE Elemento_nutrizionale (
    Simbolo CHAR(2) NOT NULL,
    NomeElemento VARCHAR(25) NOT NULL,
    Tipologia VARCHAR(25) NOT NULL,
    Descrizione VARCHAR(200),
    PRIMARY KEY (Simbolo)
)ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

-- -----
-- Tabella Esigenza_nutrizionale
-- -----

```

```

DROP TABLE IF EXISTS Esigenza_nutrizionale;
CREATE TABLE Esigenza_nutrizionale(
    IDPianta INT UNSIGNED NOT NULL,
    Elemento VARCHAR(50) NOT NULL,
    Concentrazione FLOAT NOT NULL,
    PRIMARY KEY (IDPianta, Elemento),
    FOREIGN KEY (IDPianta) REFERENCES Pianta(IDPianta) ON DELETE CASCADE ON UPDATE NO ACTION,
    FOREIGN KEY (Elemento) REFERENCES Elemento_nutrizionale(Simbolo) ON DELETE CASCADE ON UPDATE
    NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-- -----
-- Tabella Esigenza_ambientale
-- -----

```

```

DROP TABLE IF EXISTS Esigenza_ambientale;
CREATE TABLE Esigenza_ambientale (
    IDEsigenzaAmbientale INT UNSIGNED NOT NULL AUTO_INCREMENT,
    TipoEsposizione VARCHAR(50) NOT NULL,
    OreDiLuce TINYINT NOT NULL,

```

```

Periodicit Irrigazione VARCHAR(50) NULL,
Fabbisognoldrico CHAR(5) NULL,
TemperaturaMassima TINYINT NOT NULL,
TemperaturaMinima TINYINT NOT NULL,
PRIMARY KEY (IDEsigenzaAmbientale)
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

-----
-- Tabella Necessita_1
-----

```

```

DROP TABLE IF EXISTS Necessita_1;
CREATE TABLE Necessita_1(
    IDEsigenzaAmbientale INT UNSIGNED NOT NULL,
    IDPianta INT UNSIGNED NOT NULL,
    Periodo VARCHAR(100),
    PRIMARY KEY (IDEsigenzaAmbientale, IDPianta),
    FOREIGN KEY (IDEsigenzaAmbientale) REFERENCES Esigenza_ambientale(IDEsigenzaAmbientale) ON
    DELETE CASCADE ON UPDATE NO ACTION,
    FOREIGN KEY (IDPianta) REFERENCES Pianta(IDPianta) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Esigenza_intervento
-----

```

```

DROP TABLE IF EXISTS Esigenza_intervento;
CREATE TABLE Esigenza_intervento(
    IDEsigenzaIntervento INT UNSIGNED NOT NULL AUTO_INCREMENT,
    NomeIntervento VARCHAR(50) NOT NULL,
    Descrizione VARCHAR(200) NOT NULL,
    PRIMARY KEY (IDEsigenzaIntervento)
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Necessita_2
-----

```

```

DROP TABLE IF EXISTS Necessita_2;
CREATE TABLE Necessita_2(
    IDEsigenzaIntervento INT UNSIGNED NOT NULL,
    IDPianta INT UNSIGNED NOT NULL,
    Periodo VARCHAR(100) NOT NULL,
    Frequenza VARCHAR(100),
    PRIMARY KEY (IDEsigenzaIntervento, IDPianta),
    FOREIGN KEY (IDEsigenzaIntervento) REFERENCES Esigenza_intervento(IDEsigenzaIntervento) ON DELETE
    CASCADE ON UPDATE NO ACTION,
    FOREIGN KEY (IDPianta) REFERENCES Pianta(IDPianta) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

-- Tabella Patologia

```
DROP TABLE IF EXISTS Patologia;
CREATE TABLE Patologia(
    IDPatologia INT UNSIGNED NOT NULL AUTO_INCREMENT,
    NomePatologia VARCHAR(50) NOT NULL,
    AgentePatogeno VARCHAR(50) NOT NULL,
    PRIMARY KEY (IDPatologia)
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;
```

-- Tabella Suscettibilita

```
DROP TABLE IF EXISTS Suscettibilita;
CREATE TABLE Suscettibilita(
    IDPianta INT UNSIGNED NOT NULL,
    IDPatologia INT UNSIGNED NOT NULL,
    Periodo VARCHAR(50) NOT NULL,
    Probabilita VARCHAR(50) NOT NULL,
    Entita VARCHAR(50) NOT NULL,
    PRIMARY KEY (IDPianta, IDPatologia),
    FOREIGN KEY (IDPianta) REFERENCES Pianta(IDPianta) ON DELETE CASCADE ON UPDATE NO ACTION,
    FOREIGN KEY (IDPatologia) REFERENCES Patologia(IDPatologia) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;
```

-- Tabella Sintomo

```
DROP TABLE IF EXISTS Sintomo;
CREATE TABLE Sintomo(
    IDSintomo INT UNSIGNED NOT NULL AUTO_INCREMENT,
    NomeSintomo CHAR(50) NOT NULL,
    Descrizione VARCHAR(200) NOT NULL,
    PRIMARY KEY (IDSintomo)
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;
```

-- Tabella Sintomatologia

```
DROP TABLE IF EXISTS Sintomatologia;
CREATE TABLE Sintomatologia(
    IDPatologia INT UNSIGNED NOT NULL,
    IDSintomo INT UNSIGNED NOT NULL,
    PRIMARY KEY (IDPatologia, IDSintomo),
    FOREIGN KEY (IDPatologia) REFERENCES Patologia(IDPatologia) ON DELETE CASCADE ON UPDATE NO
```

```

        ACTION,
        FOREIGN KEY (IDSintomo) REFERENCES Sintomo(IDSintomo) ON DELETE CASCADE ON UPDATE NO ACTION
    )ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Immagine_sintomo
-----

```

```

DROP TABLE IF EXISTS Immagine_sintomo;
CREATE TABLE Immagine_sintomo(
    IDImmagine INT UNSIGNED NOT NULL AUTO_INCREMENT,
    Link VARCHAR(250) NOT NULL,
    IDSintomo INT UNSIGNED NOT NULL,
    PRIMARY KEY (IDImmagine),
    FOREIGN KEY (IDSintomo) REFERENCES Sintomo(IDSintomo) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Prodotto
-----

```

```

DROP TABLE IF EXISTS Prodotto;
CREATE TABLE Prodotto(
    IDProdotto INT UNSIGNED NOT NULL AUTO_INCREMENT,
    NomeCommerciale VARCHAR(50) NOT NULL,
    PrincipiAttivi VARCHAR(200) NOT NULL,
    Formulazione CHAR(2) NOT NULL,
    ModalitaSomministrazione VARCHAR(200) NOT NULL,
    PRIMARY KEY (IDProdotto)
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Cura
-----

```

```

DROP TABLE IF EXISTS Cura;
CREATE TABLE Cura(
    IDPatologia INT UNSIGNED NOT NULL,
    IDProdotto INT UNSIGNED NOT NULL,
    Dosaggio VARCHAR(100) NOT NULL,
    PRIMARY KEY (IDPatologia, IDProdotto),
    FOREIGN KEY (IDPatologia) REFERENCES Patologia(IDPatologia) ON DELETE CASCADE ON UPDATE NO ACTION,
    FOREIGN KEY (IDProdotto) REFERENCES Prodotto(IDProdotto) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Trattamento
-----

```

```

DROP TABLE IF EXISTS Trattamento;
CREATE TABLE Trattamento(
    IDTrattamento INT UNSIGNED NOT NULL AUTO_INCREMENT,
    IDProdotto INT UNSIGNED NOT NULL,
    Dosaggio VARCHAR(50) NOT NULL,
    DataInizioTrattamento DATE NOT NULL,
    DataFineTrattamento DATE,
    PRIMARY KEY (IDTrattamento),
    FOREIGN KEY (IDProdotto) REFERENCES Prodotto(IDProdotto) ON DELETE CASCADE ON UPDATE NO
    ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-- -----
-- Tabella Sede
-- -----

```

```

DROP TABLE IF EXISTS Sede;
CREATE TABLE Sede(
    IDSede INT UNSIGNED NOT NULL AUTO_INCREMENT,
    NomeSede VARCHAR(50) NOT NULL,
    Indirizzo VARCHAR(50) NOT NULL,
    Citta VARCHAR(50) NOT NULL,
    NumeroDipendenti SMALLINT UNSIGNED NOT NULL,
    PRIMARY KEY (IDSede)
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-- -----
-- Tabella Serra
-- -----

```

```

DROP TABLE IF EXISTS Serra;
CREATE TABLE Serra(
    IDSerra INT UNSIGNED NOT NULL AUTO_INCREMENT,
    NomeSerra VARCHAR(50) NOT NULL,
    Indirizzo VARCHAR(50) NOT NULL,
    Lunghezza SMALLINT NOT NULL,
    Altezza SMALLINT NOT NULL,
    Larghezza SMALLINT NOT NULL,
    IDSede INT UNSIGNED NOT NULL,
    PRIMARY KEY (IDSerra),
    FOREIGN KEY (IDSede) REFERENCES Sede(IDSede) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-- -----
-- Tabella Sezione
-- -----

```

```

DROP TABLE IF EXISTS Sezione;
CREATE TABLE Sezione(
    IDSezione INT UNSIGNED NOT NULL AUTO_INCREMENT,

```



```

NomeSezione VARCHAR(50) NOT NULL,
CapienzaMassima INT NOT NULL,
NumeroPiante INT NOT NULL,
Illuminazione VARCHAR(50) NOT NULL,
Umidita INT NOT NULL,
Temperatura INT NOT NULL,
IDSerra INT UNSIGNED NOT NULL,
PRIMARY KEY (IDSezione),
FOREIGN KEY (IDSerra) REFERENCES Serra(IDSerra) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Ripiano
-----

```

```

DROP TABLE IF EXISTS Ripiano;
CREATE TABLE Ripiano(
    IDRipiano INT UNSIGNED NOT NULL AUTO_INCREMENT,
    Irrigazione VARCHAR(50) NOT NULL,
    IDSezione INT UNSIGNED NOT NULL,
    PRIMARY KEY (IDRipiano),
    FOREIGN KEY (IDSezione) REFERENCES Sezione(IDSezione) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Contenitore
-----

```

```

DROP TABLE IF EXISTS Contenitore;
CREATE TABLE Contenitore (
    IDContenitore INT UNSIGNED AUTO_INCREMENT NOT NULL,
    DimensioneContenitore VARCHAR(25) NOT NULL,
    AltezzaPianta VARCHAR(25) NOT NULL,
    Prezzo FLOAT NOT NULL,
    IDPianta INT UNSIGNED NOT NULL,
    IDRipiano INT UNSIGNED NOT NULL,
    PRIMARY KEY (IDContenitore),
    FOREIGN KEY (IDPianta) REFERENCES Pianta(IDPianta) ON DELETE NO ACTION ON UPDATE NO ACTION,
    FOREIGN KEY (IDRipiano) REFERENCES Ripiano(IDRipiano) ON DELETE CASCADE ON UPDATE NO ACTION
) ENGINE=InnoDB DEFAULT CHARSET=latin1;

```

```

-----
-- Tabella Quarantena
-----

```

```

DROP TABLE IF EXISTS Quarantena;
CREATE TABLE Quarantena(
    IDContenitore INT UNSIGNED NOT NULL,
    IDTrattamento INT UNSIGNED NOT NULL,
    PRIMARY KEY (IDContenitore, IDTrattamento),

```

```

        FOREIGN KEY (IDContenitore) REFERENCES Contenitore(IDContenitore) ON DELETE CASCADE ON UPDATE
        NO ACTION,
        FOREIGN KEY (IDTrattamento) REFERENCES Trattamento(IDTrattamento) ON DELETE CASCADE ON UPDATE
        NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-- -----
-- Tabella Rilevazione
-- -----

```

```

DROP TABLE IF EXISTS Rilevazione;
CREATE TABLE Rilevazione(
    IDContenitore INT UNSIGNED NOT NULL,
    Data DATE NOT NULL,
    Umidita VARCHAR(50) NOT NULL,
    Esalazione VARCHAR(50),
    ConcentrazioneElementi VARCHAR(200) NOT NULL,
    IDPatologia INT UNSIGNED,
    PRIMARY KEY (IDContenitore, Data),
    FOREIGN KEY (IDContenitore) REFERENCES Contenitore(IDContenitore) ON DELETE CASCADE ON UPDATE
    NO ACTION,
    FOREIGN KEY (IDPatologia) REFERENCES Patologia(IDPatologia) ON DELETE CASCADE ON UPDATE NO
    ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-- -----
-- Tabella Utente
-- -----

```

```

DROP TABLE IF EXISTS Utente;
CREATE TABLE Utente(
    Nickname VARCHAR(50) NOT NULL,
    Nome VARCHAR(50) NOT NULL,
    Cognome VARCHAR(50) NOT NULL,
    Password VARCHAR(100) NOT NULL,
    Email VARCHAR(100) NOT NULL,
    CittaDiResidenza VARCHAR(50),
    DomandaSegreta VARCHAR(100) NOT NULL,
    RispostaSegreta VARCHAR(100) NOT NULL,
    Credibilita FLOAT,
    PRIMARY KEY (Nickname)
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-- -----
-- Tabella Preferenza
-- -----

```

```

DROP TABLE IF EXISTS Preferenza;
CREATE TABLE Preferenza(
    Utente VARCHAR(50),

```

```

        IDPianta INT UNSIGNED NOT NULL,
        PRIMARY KEY (Utente, IDPianta),
        FOREIGN KEY (Utente) REFERENCES Utente(Nickname) ON DELETE CASCADE ON UPDATE NO ACTION,
        FOREIGN KEY (IDPianta) REFERENCES Pianta(IDPianta) ON DELETE CASCADE ON UPDATE NO ACTION
    )ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Post
-----

```

```

DROP TABLE IF EXISTS Post;
CREATE TABLE Post(
    IDPost INT UNSIGNED NOT NULL AUTO_INCREMENT,
    Autore VARCHAR(50),
    Timestamp TIMESTAMP NOT NULL,
    Thread VARCHAR(50),
    Testo TEXT NOT NULL,
    SommaVoti INT UNSIGNED,
    NumeroVoti INT UNSIGNED,
    link VARCHAR(200),
    PRIMARY KEY (IDPost),
    FOREIGN KEY (Autore) REFERENCES Utente(Nickname) ON DELETE SET NULL ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Ordine
-----

```

```

DROP TABLE IF EXISTS Ordine;
CREATE TABLE Ordine(
    IDOrdine INT UNSIGNED NOT NULL AUTO_INCREMENT,
    Timestamp TIMESTAMP NOT NULL,
    Stato VARCHAR(50) NOT NULL,
    Utente VARCHAR(50) NOT NULL,
    Indirizzo VARCHAR(100) NOT NULL,
    Destinatario VARCHAR(100) NOT NULL,
    IDPianta INT UNSIGNED NOT NULL,
    IDContenitore INT UNSIGNED,
    PRIMARY KEY (IDOrdine),
    FOREIGN KEY (Utente) REFERENCES Utente(Nickname) ON DELETE CASCADE ON UPDATE NO ACTION,
    FOREIGN KEY (IDContenitore) REFERENCES Contenitore(IDContenitore) ON DELETE SET NULL ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Scheda
-----

```

```

DROP TABLE IF EXISTS Scheda;
CREATE TABLE Scheda(

```

```

    IDScheda INT UNSIGNED NOT NULL AUTO_INCREMENT,
    Dimensione VARCHAR(100) NOT NULL,
    AttualeCollocazione VARCHAR(50) NOT NULL,
    DimensioneVaso VARCHAR(50),
    IDPianta INT UNSIGNED NOT NULL,
    Utente VARCHAR(50) NOT NULL,
    PRIMARY KEY (IDScheda),
    FOREIGN KEY (IDPianta) REFERENCES Pianta(IDPianta) ON DELETE NO ACTION ON UPDATE NO ACTION,
    FOREIGN KEY (Utente) REFERENCES Utente(Nickname) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Intervento
-----

```

```

DROP TABLE IF EXISTS Intervento;
CREATE TABLE Intervento(
    IDIntervento INT UNSIGNED NOT NULL AUTO_INCREMENT,
    TipoIntervento VARCHAR(100),
    Indirizzo VARCHAR(100),
    Costo SMALLINT UNSIGNED,
    Data DATE NOT NULL,
    Ricorrenza VARCHAR(100),
    IDEsigenzaIntervento INT UNSIGNED NOT NULL,
    IDScheda INT UNSIGNED NOT NULL,
    PRIMARY KEY (IDIntervento),
    FOREIGN KEY (IDEsigenzaIntervento) REFERENCES Esigenza_intervento(IDEsigenzaIntervento) ON DELETE
    CASCADE ON UPDATE NO ACTION,
    FOREIGN KEY (IDScheda) REFERENCES Scheda(IDScheda) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Spazio
-----

```

```

DROP TABLE IF EXISTS Spazio;
CREATE TABLE Spazio(
    IDSpazio INT UNSIGNED NOT NULL AUTO_INCREMENT,
    Preventivo INT NOT NULL,
    Utente CHAR(50) NOT NULL,
    PRIMARY KEY (IDSpazio),
    FOREIGN KEY (Utente) REFERENCES Utente(Nickname) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-----
-- Tabella Settore
-----

```

```

DROP TABLE IF EXISTS Settore;
CREATE TABLE Settore(

```

```

        IDSettore INT UNSIGNED NOT NULL AUTO_INCREMENT,
        Coltivabile CHAR(2) NOT NULL,
        TipoLuce VARCHAR(50) NOT NULL,
        OreLuceGiornaliere TINYINT NOT NULL,
        EsposizioneCardinale VARCHAR(50) NOT NULL,
        IDSpazio INT UNSIGNED NOT NULL,
        PRIMARY KEY (IDSettore),
        FOREIGN KEY (IDSpazio) REFERENCES Spazio(IDSpazio) ON DELETE CASCADE ON UPDATE NO ACTION
    )ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-- -----
-- Tabella Vertice
-- -----

```

```

DROP TABLE IF EXISTS Vertice;
CREATE TABLE Vertice(
    IDVertice INT UNSIGNED NOT NULL AUTO_INCREMENT,
    CoordinataX DOUBLE NOT NULL,
    CoordinataY DOUBLE NOT NULL,
    PRIMARY KEY (IDVertice)
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-- -----
-- Tabella Confine
-- -----

```

```

DROP TABLE IF EXISTS Confine;
CREATE TABLE Confine(
    IDVertice INT UNSIGNED NOT NULL,
    IDSettore INT UNSIGNED NOT NULL,
    PRIMARY KEY (IDVertice, IDSettore),
    FOREIGN KEY (IDVertice) REFERENCES Vertice(IDVertice) ON DELETE CASCADE ON UPDATE NO ACTION,
    FOREIGN KEY (IDSettore) REFERENCES Settore(IDSettore) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```

-- -----
-- Tabella Componente_giardino
-- -----

```

```

DROP TABLE IF EXISTS Componente_giardino;
CREATE TABLE Componente_giardino (
    IDComponente INT UNSIGNED NOT NULL AUTO_INCREMENT,
    CoordinataX DOUBLE NOT NULL,
    CoordinataY DOUBLE NOT NULL,
    Raggio DOUBLE NOT NULL,
    IDPianta INT UNSIGNED NOT NULL,
    IDSettore INT UNSIGNED NOT NULL,
    PRIMARY KEY (IDComponente),
    FOREIGN KEY (IDPianta) REFERENCES Pianta(IDPianta) ON DELETE CASCADE ON UPDATE NO ACTION,
    FOREIGN KEY (IDSettore) REFERENCES Settore(IDSettore) ON DELETE CASCADE ON UPDATE NO ACTION
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

```

```
) ENGINE=InnoDB DEFAULT CHARSET=latin1;
```

```
-- Tabella Vaso
```

```
DROP TABLE IF EXISTS Vaso;
```

```
CREATE TABLE Vaso(
```

```
    IDVaso INT UNSIGNED NOT NULL AUTO_INCREMENT,
```

```
    Materiale VARCHAR(50) NOT NULL,
```

```
    CoordinataX DOUBLE NOT NULL,
```

```
    CoordinataY DOUBLE NOT NULL
```

```
    Raggio DOUBLE NOT NULL,
```

```
    IDSettore INT UNSIGNED NOT NULL,
```

```
    PRIMARY KEY (IDVaso),
```

```
    FOREIGN KEY (IDSettore) REFERENCES Settore(IDSettore) ON DELETE CASCADE ON UPDATE NO ACTION
```

```
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;
```

Popolamento

```
-- Inserimento in Terreno
```

```
INSERT INTO Terreno
```

```
VALUES(1, 8, 'Argilloso, calcareo', 'medio-alta'),
```

```
(2, 7, 'Sabbioso', 'medio-alta'),
```

```
(3, 6.5, 'Sabbioso', 'alta'),
```

```
(4, 5.5, 'Sabbioso, umifero', 'alta'),
```

```
(5, 7, 'Gessoso, argilloso', 'medio-bassa'),
```

```
(6, 7.5, 'Gessoso, argilloso', 'media'),
```

```
(7, 8.5, 'Argilloso', 'medio-bassa'),
```

```
(8, 7, 'Gessoso, sabbioso', 'Bassa'),
```

```
(9, 8, 'Gessoso', 'Alta'),
```

```
(10, 6, 'Argilloso', 'Bassa');
```

```
-- Inserimento in Pianta
```

```
INSERT INTO Pianta
```

```
VALUES(1, 'Olivo', 'Olea', 'Picholine', 15, 15, 'Molto lento', 'apr - giu', 'ott - dic', 'gen - dic', 'si', 'no', 'Medio', 15, 1),
```

```
(2, 'Oleandro', 'Nerium', null, 10, 8, 'Medio', 'mag - dic', 'set - ott', 'gen - dic', 'si', 'no', 'Semplice', 5, 1),
```

```
(3, 'Verbena Odorosa', 'Aloysia', null, 3, 1.5, 'Veloce', 'mag - giu', null, 'mar - ott', 'no', 'no', 'Semplice', 3, 2),
```

```
(4, 'Palmetto', 'Sabal', 'Bermudana', 25, 8, 'Lento', 'lug - ago', 'set - ott', 'gen - dic', 'si', 'no', 'Semplice', 30, 3),
```

```
(5, 'Ortensia', 'Hydrangea', 'Annabelle', 1.5, 1.5, 'Veloce', 'lug - set', null, 'mar - nov', 'no', 'no', 'Medio', 12, 4),
```

```
(6, 'Salice piangente', 'Salix', null, 25, 20, 'Lento', 'apr - mag', null, 'feb - ott', 'no', 'si', 'Medio', 35, 5),
```

```
(7, 'Glicine', 'Wisteria', 'Black dragon', 12, 10, 'Medio', 'apr - giu', 'set - ott', 'mar - nov', 'no', 'no', 'Medio', 10, 6),
```

```
(8, 'Rosa di Natale', 'Helleborus', 'Emma', 0.35, 0.5, 'Molto veloce', 'gen - feb', null, 'gen - dic', 'si', 'no', 'Semplice',
```

13, 7),
 (9, 'Alloro', 'Laurus', null, 20, 10, 'Molto Lento', 'apr - giu', 'ago - ott', 'gen - dic', 'si', 'si', 'Semplice', 7, 8),
 (10, 'Vite', 'Vitis', 'Sangiovese', 12, 8, 'Veloce', 'mag - giu', 'set - ott', 'mar - nov', 'no', 'no', 'Difficile', 2, 9),
 (11, 'Filadelfo', 'Philadelphus', 'Belle etoile', 5, 2, 'Medio', 'mag - giu', null, 'mar - dic', 'no', 'no', 'Semplice', 10, 5);

 -- Inserimento Elemento_nutrizionale

INSERT INTO Elemento_nutrizionale
 VALUES('N', 'Azoto', 'Macroelemento', 'È la sostanza che serve per l'accrescimento delle piante.'),
 ('P', 'Fosforo', 'Macroelemento', 'Agevola la fioritura e l'accrescimento e relativa maturazione dei frutti oltre che un miglior sviluppo dell'apparato radicale.'),
 ('K', 'Potassio', 'Macroelemento', 'Favorisce e migliora il sapore, il colore e la consistenza dei frutti; aiuta le piante a resistere al freddo, alla siccità e all'attacco dei parassiti.'),
 ('Ca', 'Calcio', 'Mesoelemento', 'Rinforza le piante.'),
 ('Mg', 'Magnesio', 'Mesoelemento', 'Insieme al ferro rappresenta un elemento essenziale della molecola della clorofilla, molto importante per la fotosintesi clorofilliana.'),
 ('S', 'Zolfo', 'Mesoelemento', 'È la sostanza che concorre al sapore di molti ortaggi e frutti.'),
 ('Cl', 'Cloro', 'Mesoelemento', 'Aiuta la crescita delle radici'),
 ('Fe', 'Ferro', 'Microelemento', 'Insieme al magnesio rappresenta un componente principale della molecola della clorofilla.'),
 ('Cu', 'Rame', 'Microelemento', 'Insieme al ferro partecipa alla produzione della clorofilla oltre che essere un costituente di molti enzimi necessari alla regolazione dei processi vitali della pianta.'),
 ('Zn', 'Zinco', 'Microelemento', 'Serve per la formazione e lo sviluppo dei semi.'),
 ('Mn', 'Manganese', 'Microelemento', 'Interviene nella sintesi della clorofilla.'),
 ('B', 'Boro', 'Microelemento', 'Partecipa, insieme ad altri elementi, alla formazione di fiori, frutti e radici.'),
 ('Mo', 'Molibdeno', 'Microelemento', 'È una sostanza che serve allo sviluppo e alla crescita delle piante.');

 -- Inserimento Esigenza_nutrizionale

INSERT INTO Esigenza_nutrizionale
 VALUES(1, 'N', 1.22),
 (1, 'Ca', 2.64),
 (1, 'Mg', 0.22),
 (2, 'P', 1.42),
 (2, 'K', 1.65),
 (3, 'K', 1.4),
 (3, 'Fe', 0.21),
 (5, 'Fe', 0.6),
 (5, 'N', 1.5),
 (7, 'K', 1.3),
 (7, 'P', 0.9),
 (8, 'K', 1.3),
 (9, 'N', 1.6),
 (9, 'Mn', 0.6),
 (9, 'Zn', 0.12);

-- Inserimento Esigenza_ambientale

```
INSERT INTO Esigenza_ambientale
VALUES(1, 'Mezzombra, pieno sole', 12, 'Una volta alla settimana', 'Basso', 40, -5),
(2, 'Mezzombra, pieno sole', 8, 'Mai', 'Nullo', 30, -5),
(3, 'Pieno sole', 10, 'Una volta ogni tre giorni', 'Basso', 40, 5),
(4, 'Pieno sole', 12, 'Una volta ogni due giorni', 'Medio', 40, 5),
(5, 'Mezzombra', 10, 'Ogni giorno', 'alto', 36, 0),
(6, 'Mezzombra, ombra', 8, 'Mai', 'Nullo', 20, -10),
(7, 'Pieno sole', 10, 'Ogni giorno', 'alto', 37, -10),
(8, 'Pieno sole', 12, 'Solo nei periodi di siccità prolungata', 'Basso', 37, -15),
(9, 'Ombra', 9, 'Una volta alla settimana', 'Basso', 30, -15),
(10, 'Mezzombra, pieno sole', 10, 'Solo nei periodi di siccità prolungata', 'Basso', 40, -10);
```

-- Inserimento Necessita_1

```
INSERT INTO Necessita_1
VALUES(1, 1, 'mar - ott'),
(2, 1, 'nov - feb'),
(1, 2, "tutto l'anno"),
(3, 3, 'mar - ott'),
(4, 4, 'mar - set'),
(5, 5, 'mar - nov'),
(6, 5, 'dic - feb'),
(7, 6, 'feb - ott'),
(8, 7, "tutto l'anno"),
(9, 8, "tutto l'anno"),
(10, 9, "tutto l'anno"),
(8, 10, "tutto l'anno");
```

-- Inserimento Esigenza_intervento

```
INSERT INTO Esigenza_intervento
VALUES(1, 'Concimazione', 'Concimare con elementi nutritivi mancanti nel terreno'),
(2, 'Potatura di ringiovanimento', 'Tagliare tutti i rami vecchi, secchi o improduttivi'),
(3, 'Spollonatura', 'Tagliare i polloni'),
(4, 'Diradamento', 'Tagliare i rami ritenuti in soprannumero'),
(5, 'Potatura di contenimento', 'Tagliare i rami per contenere le dimensioni della pianta'),
(6, 'Rinvaso', 'Spostare la pianta in un vaso delle dimensioni adeguate'),
(7, 'Spuntatura', 'Accorciare tutti i rami giovani e che hanno fiorito, lasciando per ognuno di essi tre o quattro gemme'),
(8, 'Potatura di riforma', "Praticata quando sorge l'esigenza di variare la forma di un albero"),
(9, 'Potatura di fruttificazione', "Praticata per favorire lo sviluppo dei rami che portano fiori e/o frutti per
```


migliorarne la qualità"),
(10, 'Cimatura', 'Eliminare i nodi terminali di un ramo');

-- Inserimento Necessita_2

```
INSERT INTO Necessita_2
VALUES(1, 1, 'Febbraio-marzo', 'Ogni 4 anni'),
(2, 1, 'Marzo-maggio', 'Ogni 4 anni'),
(3, 1, 'Marzo-maggio', 'Ogni 2 anni'),
(4, 2, 'Febbraio-marzo o settembre-ottobre', 'Ogni anno'),
(2, 2, "Durante tutto l'anno", 'Ogni anno'),
(5, 5, "Durante tutto l'anno", 'Ogni 3 anni'),
(1, 5, 'Gennaio-Febbraio', 'Ogni 3 anni'),
(7, 7, 'Gennaio-febbraio', 'Ogni anno'),
(1, 8, 'Marzo-novembre', 'Ogni anno'),
(8, 9, 'Gennaio-Febbraio', 'Ogni anno'),
(1, 10, 'Gennaio-Febbraio', 'Ogni anno');
```

-- Inserimento Patologia

```
INSERT INTO Patologia
VALUES(1, "Complesso del disseccamento rapido dell'olivo", 'Xylella fastidiosa'),
(2, "Rogna", 'Pseudomonas savastanoi'),
(3, 'Infestazione da punteruolo rosso', 'Rhynchophorus ferrugineus'),
(4, 'Infestazione da ragnetto rosso', 'Tetranychus urticae'),
(5, 'Infestazione da afidi', 'Aphidoidea'),
(6, 'Infestazione da cocciniglia scudettata', 'Coccidi'),
(7, 'Botrite', 'Botris cinerea'),
(8, 'Oidio', 'Erysiphaceae'),
(9, 'Infestazione da Metcalfa', 'Metcalfa Pruinosa'),
(10, 'Infestazione da Psilla', 'Trioza alacris');
```

-- Inserimento Suscettibilita

```
INSERT INTO Suscettibilita
VALUES(1, 1, "Tutto l'anno", 'Bassa', 'Grave'),
(1, 2, "Tutto l'anno", 'Alta', 'Lieve'),
(2, 2, "Tutto l'anno", 'Alta', 'Lieve'),
(4, 3, 'feb - set', 'Alta', 'Grave'),
(4, 5, 'feb -giu', 'Alta', 'Lieve'),
(5, 4, 'apr - ago', 'Media', 'Lieve'),
(7, 6, 'mar - set', 'Media', 'Lieve'),
(7, 8, 'giu - lug', 'Bassa', 'Media'),
(8, 5, 'feb -giu', 'Media', 'Lieve'),
```

```
(9, 10, 'mar - mag', 'Media', 'Media'),  
(10, 7, 'ago - set', 'Media', 'Grave');
```

```
-- Inserimento Sintomo
```

```
INSERT INTO Sintomo
```

```
VALUES(1, 'Bruscatura', 'Disseccamento delle foglie nella parte apicale e/o marginale'),  
(2, 'Escrescenze', 'Crescita di protuberanze tumorali sui rami e/o sulle foglie'),  
(3, 'Chioma ad ombrello aperto', 'Portamento anomalo della chioma delle piante'),  
(4, 'Perdita della chioma', 'La pianta perde la chioma e appare come "capitozzata"),  
(5, 'Ingiallimento', 'Le foglie prendono un aspetto giallo-rugginoso'),  
(6, 'Arrotolamento fogliare', 'Il lembo fogliare si ripiega su se stesso'),  
(7, 'Fumaggine', 'Feltro di colore nerastro sulle foglie'),  
(8, 'Deformazione fogliare', 'Le foglie appaiono deformate'),  
(9, 'Muffa grigia', 'Comparsa di macchie grigiastre sulle foglie, sui germogli o sui frutti in via di maturazione'),  
(10, 'Macchie bianche', 'Chiazze bianche sulla pagina superiore delle foglie, sui germogli e sui fiori');
```

```
-- Inserimento Sintomatologia
```

```
INSERT INTO Sintomatologia
```

```
VALUES(1, 1),  
(2, 2),  
(3, 3),  
(3, 4),  
(4, 5),  
(5, 6),  
(5, 7),  
(6, 7),  
(6, 8),  
(7, 9),  
(8, 10),  
(9, 7),  
(10, 5),  
(10, 6);
```

```
-- Inserimento Immagine_sintomo
```

```
INSERT INTO Immagine_sintomo
```

```
VALUES(1, 'http://2.bp.blogspot.com/-207ACJTAQe4/VOcgLVL3GtI/AAAAAAAAADg/c3ncrxoe9rQ/s1600/xyl1.jpeg',  
1),  
(2, 'http://agroambiente.info.arsia.toscana.it/docs/lab/rognOlivoRicciolini/images/untitled_009.jpg', 2),  
(3,  
'http://img05.comecoltivare.it/it/piante_coltivazione/oleandro/immagini/pseudomonas_syringae_savastanoi.jpg',  
2),  
(4, 'https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQNHHNIOgaZ6qkz3bZRqZ3n5dJENDhvnSePRgrb-
```

zNoP18T70px', 3),
 (5, 'http://www.qlnews.it/wp-content/uploads/2015/11/palma-punteruolo-rosso.jpg', 4),
 (6, 'http://www.idroponica.it/db_img/img/attacco-foglie-ragnetto-rosso_13B-4088.jpg', 5),
 (7, 'http://www.coltivarelorto.it/Images/2014/012DIC/09713-2012-12-07-FIG-003-FERRO.jpg', 5),
 (8, 'https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTeEy25YsVQRNTyPyqVgEnpDFUcy9XTZx0SBaJ64zyx-SVRFoZ_AGU-A2w', 6),
 (9, 'https://upload.wikimedia.org/wikipedia/commons/thumb/3/38/Scale_and_sooty_mold_on_a_Eucalyptus_tree.jpg/220px-Scale_and_sooty_mold_on_a_Eucalyptus_tree.jpg', 7),
 (10, 'http://www.auroras.eu/wp-content/uploads/botritegrappo-263x300.png', 9),
 (11, 'http://www.legeorgiche.it/wp-content/uploads/2014/09/botrite-muffa-grigia-come-eliminarla.jpg', 9);

 -- Inserimento Prodotto

INSERT INTO Prodotto

VALUES(1, 'Reldan 22', 'Clorpirifos-metile', 'EC', 'Nebulizzazione, irrigazione, applicazione localizzata'),
 (2, 'Kohinor Plus Palme Tris', 'Imidacloprid e Ciflutrin', 'EC', 'Nebulizzazione, applicazione localizzata'),
 (3, 'Borneo', 'Etoxazole', 'SC', 'Nebulizzazione'),
 (4, 'Confidor', 'Imidacloprid', 'SL', 'Nebulizzazione, irrigazione'),
 (5, 'Calypso Garden Protector', 'Thiacloprid', 'SE', 'Nebulizzazione, irrigazione'),
 (6, 'Scala', 'Pyrimethanil', 'SC', 'Nebulizzazione'),
 (7, 'Folicur giardino', 'Tebuconazolo', 'SE', 'Nebulizzazione'),
 (8, 'Emerald 40', 'Tetraconazolo', 'EW', 'Nebulizzazione'),
 (9, 'Neemazal T/S', 'Azadiractina', 'EC', 'Nebulizzazione'),
 (10, 'Vermitec', 'Abamectina', 'EC', 'Nebulizzazione');

 -- Inserimento Cura

INSERT INTO Cura

VALUES(3, 1, "2. ml/l d'acqua"),
 (3, 2, "5 ml/l d'acqua"),
 (4, 3, "0,35 ml/l d'acqua"),
 (5, 4, "0,5 ml/l d'acqua"),
 (6, 4, "0,75 ml/l d'acqua"),
 (5, 5, "15 ml/l d'acqua"),
 (6, 5, "20 ml/l d'acqua"),
 (9, 5, "20 ml/l d'acqua"),
 (7, 6, "2 ml/l d'acqua"),
 (8, 7, "3 ml/l d'acqua"),
 (8, 8, "0,7 ml/l d'acqua"),
 (10, 9, "3 ml/l d'acqua"),
 (10, 10, "0,6 ml/l d'acqua");

-- Inserimento Sede

INSERT INTO Sede

VALUES(1, 'Noncello', 'Via Cappuccini, 23', 'Pordenone', 12),
(2, 'Mazzini', 'Torino', 'Via Sandro Botticelli', 35),
(3, 'Fibonacci', 'Via Cisanello, 12', 'Pisa', 20),
(4, 'Galante', 'Contrada Galante, 5', 'Lecce', 15),
(5, 'Pinelli', 'Via Fatebenefratelli, 11', 'Milano', 43),
(6, 'Modugno', 'Via Francesco Crispi, 43', 'Bari', 38),
(7, 'De Meis', 'Via Luigi Pirandello, 100', 'Roma', 32),
(8, 'Impastato', 'Via Quintino Sella, 4', 'Palermo', 19),
(9, 'Nduja', 'Via Savocello, 10', 'Tiriolo', 24),
(10, 'Da Vinci', 'Corso Vittorio Emanuele II, 80', 'Napoli', 36);

-- Inserimento Serra

INSERT INTO Serra

VALUES(1, 'Blu', 'Via Cappuccini, 23', 12, 3, 7, 1),
(2, 'Verde', 'Via Cappuccini, 23', 18, 3, 7, 1),
(3, 'Alberi', 'Via Giaveno, 5', 20, 4, 10, 2),
(4, 'Fiori', 'Via Giaveno, 5', 15, 4, 15, 2),
(5, 'Serra_1', 'Viale dei Caduti', 30, 10, 20, 5),
(6, 'De Meis', 'Via Luigi Pirandello, 100', 20, 5, 10, 7),
(7, 'Serra A', 'Via Chieti', 11, 3, 5, 8),
(8, 'Serra B', 'via Marco Polo, 12', 10, 3, 10, 8),
(9, 'Serra C', 'via Amerigo Vespucci', 15, 5, 15, 8),
(10, 'Grandi Speranze', 'Via Domodossola, 8', 20, 5, 25, 10);

-- Inserimento Sezione

INSERT INTO Sezione

VALUES(1, 'SezBlu', 60, 20, '12 ore al giorno', 60, 25, 1),
(2, 'Verde est', 30, 30, '10 ore al giorno', 70, 30, 2),
(3, 'Verde ovest', 45, 13, '8 ore al giorno', 55, 30, 2),
(4, 'Frutta', 30, 20, 'luce solare', 70, 20, 3),
(5, 'Estivi', 60, 55, '10 ore al giorno', 45, 25, 4),
(6, 'Perenni', 60, 30, '10 ore al giorno', 45, 25, 4),
(7, 'Sez 1', 20, 5, '9 ore al giorno', 63, 30, 7),
(8, 'Sez 2', 30, 0, '10 ore al giorno', 52, 23, 7),
(9, 'Crescita rapida', 35, 35, 'luce solare', 45, 30, 9),
(10, 'Esotica', 50, 38, '12 ore al giorno', 76, 35, 10);

-- Inserimento Ripiano

INSERT INTO Ripiano

VALUES(1, 'alta', 1),
(2, 'media', 1),
(3, 'alta', 2),
(4, 'alta', 3),
(5, 'bassa', 5),
(6, 'media', 5),
(7, 'alta', 4),
(8, 'bassa', 7),
(9, 'media', 8),
(10, 'alta', 10);

-- Inserimento Contenitore

INSERT INTO Contenitore

VALUES(1, 'D:20cm A:25cm', '50 cm', 4.99, 2, 1),
(2, 'D:30cm A:35cm', '1.5 m', 12.99, 1, 2),
(3, 'D:20cm A:20cm', '1 m', 3.99, 3, 5),
(4, 'D:20cm A:25cm', '75 cm', 3.99, 10, 4),
(5, 'D:15cm A:15cm', '25 cm', 10.59, 8, 3),
(6, 'D:15cm A:15cm', '35 cm', 12.99, 8, 3),
(7, 'D:25cm A:25cm', '1 m', 5.99, 9, 7),
(8, 'D:30cm A:40cm', '2 m', 28.99, 4, 8),
(9, 'D:25cm A:25cm', '1 m', 15.99, 6, 9),
(10, 'D:20cm A:15cm', '45 cm', 9.99, 5, 10);

-- Inserimento Trattamento

INSERT INTO Trattamento

VALUES(1, 3, "0,35 ml/l d'acqua", '2014-06-13', '2014-06-19'),
(2, 4, "0,5 ml/l d'acqua", '2016-04-01', '2016-04-27'),
(3, 4, "0,6 ml/l d'acqua", '2017-02-15', null),
(4, 5, "20 ml/l d'acqua", '2016-09-21', '2016-10-30'),
(5, 2, "5 ml/l d'acqua", '2015-02-02', '2015-03-02'),
(6, 2, "5 ml/l d'acqua", '2015-02-08', '2015-02-15'),
(7, 7, "3 ml/l d'acqua", '2017-01-31', null),
(8, 7, "3,5 ml/l d'acqua", '2016-04-20', '2016-05-27'),
(9, 7, "3,5 ml/l d'acqua", '2016-04-30', '2016-05-17'),
(10, 9, "3 ml/l d'acqua", '2017-02-10', null);

-- Inserimento Quarantena

INSERT INTO Quarantena

VALUES(1, 1),
(3, 2),
(4, 2),
(3, 3),
(4, 3),
(5, 2),
(6, 2),
(7, 9),
(8, 9),
(9, 9),
(10, 5);

-- Inserimento Rilevazione

INSERT INTO Rilevazione

VALUES(1, '2017-02-09', '50%', null, 'N:0.8g, Cu:0.4g, K:1.2g', null),
(1, '2017-02-10', '50%', null, 'N:0.82g, Cu:0.4g, K:1.18g', null),
(2, '2017-02-09', '63%', 'Metano', 'Ca:0.8g, Mn:0.3g', null),
(3, '2017-02-10', '58%', null, 'K:0.91g, Fe:0.12g', 2),
(4, '2017-02-09', '62%', null, 'P:0.78g, Fe:0.29g K:0.23g', null),
(5, '2017-02-08', '74%', 'Ammoniac', 'N:0.52g, K:0.67g, Fe:0.12g', 7),
(7, '2017-02-10', '62%', null, 'Ca:0.46g, Fe:0.2g, K:1.12g', 5),
(8, '2017-02-9', '43%', 'Metano', 'Fe:0.2g, N:0.43g', 3),
(8, '2017-02-10', '42%', 'Metano', 'Fe:0.3g, N:0.48g K:0.3g', 3),
(10, '2017-02-09', '69%', null, 'Zn:0.3g, P:1.02g', 6);

-- Inserimento Utente

INSERT INTO Utente

VALUES('Billy82', 'Roberto', 'Pasolini', 'abc123d', 'r.pasolini@gmail.com', 'Milano', 'Il nome del tuo primo animale domestico?', 'Gorzorg', 2.5),
('Fragolina1', 'Giovanna', 'Angiolini', 'IpsoDicto', 'giovannanna.a@outlook.com', 'Bassano Del Grappa', 'nome della tua zia preferita?', 'Mariolina', 3.4),
('NikoBellic', 'Domenico', 'Dellisanti', 'mich3l4', 'ilmastino12@yahoo.com', 'Squinzano', 'Il compleanno di tua moglie?', 'non lo so', 0),
('Giups', 'Giuseppe', 'Pompilio', 'ForzaAtalanta', 'giu.seppe1a@gmail.com', 'Volterra', 'il nome del gatto?', 'Attila', 4.5),
('IlSignorMarioRossi', 'Mario', 'Rossi', 'HoComprato2Mele', 'mario.rossi@example.it', 'Milano', 'Il tuo piatto preferito?', 'Cozze gratinate', 5),
('Nicole23', 'Nicole', 'Landi', 'a1b2c3scaccoalre', 'nic.min123@hotmail.it', 'Rovigo', 'Il Cognome di tua madre?', 'Wilson', 1.4),

```

('AjejeBrazorf', 'Aldo', 'Baglio', '4ld0', 'pdor.kmer@outlook.com', null, 'Il nome del tuo cane?', 'Ringhio', 4.3),
('IlMoralizzatore', 'Ludovica', 'Cerini', 'Schell3e4t0', 'cerini.ludovica@gmail.it', 'Bologna', 'Il giorno del tuo
matrimonio?', '25/08/2013', 4.3),
('Royksopp', 'Alberto', 'Bianchi', '123asd987', 'berto.white@yahoo.com', null, 'Il nome di tua nonna?', 'Leonarda',
3),
('Mammolo83', 'Guido', 'Calandrin', 'O5tr3gA', 'guido.cala83@hotmail.it', 'Venezia', 'Il nome del pappagallo?',
'Flint', 4.2);

```

```

-----
-- Inserimento Preferenza
-----

```

```

INSERT INTO Preferenza

```

```

VALUES('Billy82', 2),
('Billy82', 8),
('Fragolina1', 4),
('Giups', 5),
('Giups', 7),
('Giups', 6),
('Royksopp', 10),
('Mammolo83', 2),
('Mammolo83', 3),
('Nicole23', 1);

```

```

-----
-- Inserimento Scheda
-----

```

```

INSERT INTO Scheda

```

```

VALUES(1, 'Altezza: 10m, diametro: 3m', 'Piena terra', null, 4, 'Billy82'),
(2, 'Altezza: 10m, diametro: 3m', 'Piena terra', null, 4, 'Billy82'),
(3, 'Altezza: 1m, diametro: 1m', 'Vaso', 'Altezza: 1m, Diametro: 1m', 3, 'Fragolina1'),
(4, 'Altezza: 2m, diametro: 1.5m', 'Piena terra', null, 9, 'Giups'),
(5, 'Altezza: 12m, diametro: 4m', 'Piena terra', null, 6, 'IlSignorMarioRossi'),
(6, 'Altezza: 0.35m, diametro: 0.5m', 'Vaso', 'Altezza: 0.25m, diametro: 0.25m', 8, 'NikoBellic'),
(7, 'Altezza: 1.5m, diametro: 1m', 'Piena terra', null, 10, 'Mammolo83'),
(8, 'Altezza: 5m, diametro: 5m', 'Piena terra', null, 1, 'Royksopp'),
(9, 'Altezza: 1m, diametro: 0.5m', 'Vaso', 'Altezza: 0.35m, diametro: 0.35m', 7, 'AjejeBrazorf'),
(10, 'Altezza: 2m, diametro: 3m', 'Vaso', 'Altezza: 0.35m, diametro: 0.35m', 2, 'Nicole23');

```

```

-----
-- Inserimento Intervento
-----

```

```

INSERT INTO Intervento

```

```

VALUES(1, 'Programmata', 'Via Pico Della Mirandola, 12', 25, '2017-03-21', 'Ogni anno', 2, 4),
(2, 'Richiesta', 'Via Giuseppe Giusti, 82', 35, '2016-08-10', null, 6, 3),
(3, null, null, null, '2015-04-12', null, 5, 1),
(4, 'Richiesta', 'Via Roma, 5', 30, '2017-03-08', null, 8, 5),
(5, 'Programmata', 'Via Botticelli, 102', 40, '2017-05-10', 'Ogni 4 anni', 9, 8),

```

(6, 'Richiesta', 'Via Galileo Galilei, 1', null, '2017-07-16', null, 6, 9),
(7, 'Automatica', 'Via Cosenza, 52', null, '2017-03-01', 'Ogni 2 anni', 3, 4),
(8, null, null, null, '2015-07-27', null, 4, 10),
(9, 'Programmata', 'Via Crispi, 8', 25, '2017-05-15', 'Ogni anno', 2, 7),
(10, null, 'Via Antonio Gramsci', null, '2016-08-08', null, 6, 6);

-- Inserimento Post

INSERT INTO Post

VALUES(1, 'Fragolina1', '2016-05-12 15:24:13', 'Aiuto afidi!', 'Stamattina ho notato che le mie rose erano piene di afidi...come faccio?', null, null, null),
(2, 'Giups', '2016-05-12 18:08:43', 'Aiuto afidi!', 'Prova con Confidor', 26, 6, 'http://www.bayergarden.it/Prodotti/c/Confidor-200-SL'),
(3, 'Nicole23', '2016-05-13 09:32:14', 'Aiuto afidi!', 'Se non vuoi usare prodotti chimici prova a spurzzarci sopra un decotto all'aglio.', 56, 12, null),
(4, 'Royksopp', '2016-08-21 16:54:51', 'Raccolta noci', 'Qualcuno sa quando è il periodo migliore per raccogliere le noci?', null, null, null),
(5, 'AjejeBrazorf', '2017-01-23 18:21:39', 'Consigli Dionea', 'Sono alla prima esperienza, ho comprato una dionea ma in due settimane è cresciuto un fusto centrale stretto e lungo di almeno 10 cm! è normale?', null, null, null),
(6, 'Mammolo83', '2017-01-23 18:54:12', 'Consigli Dionea', 'Ciao. Tranquillo è solo uno stelo florale. ti lascio un link con altre informazioni utili.', 52, 11, 'https://www.giardinaggio.it/piante-grasse/piante-carnivore/pianta-carnivora-dionea.asp'),
(7, 'NikoBellic', '2016-12-03 15:32:16', 'Un po' di fiori', 'Visto che ora la natura è un tripudio di fiori, ne posto alcuni dei miei bonsai', null, null, 'https://giardinaggio.it/attachments/a-004-jpg.145020/'),
(8, 'IlMoralizzatore', '2016-12-03 18:02:23', 'Un po' di fiori', 'Sono orribili!', 5, 9, null),
(9, 'IlSignorMarioRossi', '2017-02-03 11:40:12', 'Lavanda più profumata', 'Buongiorno! Questa primavera vorrei piantare della lavanda in giardino. Però, come da titolo, vorrei sapere quale sia la varietà più profumata. Grazie :)', null, null, null),
(10, 'Giups', '2017-02-04 15:09:25', 'Lavanda più profumata', 'Dipende dagli spazi che hai...io conosco l'angustifolia (che rimane più piccola)', 21, 6, 'https://www.giardinaggio.it/giardino/piante/lavanda_NG3.jpg');

-- Inserimento Ordine

INSERT INTO Ordine

VALUES(1, '2017-02-5 13:03:14', 'Spedito', 'NikoBellic', 'Via Luigi Masi, 4, Squinzano (LE)', 'Domenico Dellisanti', 3, 3),
(2, '2017-02-10 08:21:16', 'In processazione', 'Nicole23', 'Via Giovanni Pascoli, 3, Rovigo (RO)', 'Nicole Landi', 10, 4),
(3, '2016-12-14 12:05:23', 'Evaso', 'Royksopp', 'Corso Pietro Vannucci, 20, Perugia (PG)', 'Goffredo Bianchi', 2, null),
(4, '2016-11-28 17:25:54', 'Evaso', 'Fragolina1', 'Viale Venezia, 82, Bassano del Grappa (VI)', 'Giovanna Angiolini', 5, null),
(5, '2017-01-20 09:39:42', 'In preparazione', 'Fragolina1', 'Viale Venezia, 82, Bassano del Grappa (VI)', 'Giovanna Angiolini', 1, 2),
(6, '2017-02-04 11:12:13', 'Pendente', 'AjejeBrazorf', 'Via Napoleone Colajanni, 42, Bari (BA)', 'Aldo Baglio', 1, null),
(7, '2016-12-23 21:54:02', 'Evaso', 'IlSignorMarioRossi', 'Via Antonio Rosmini, 5, Milano (MI)', 'Mario Rossi', 8, null),
(8, '2016-10-10 16:42:38', 'Pendente', 'Mammolo83', 'Via Rizzetti, 39, Castelfranco Veneto (TV)', 'Guido Calandrin',


```
1, null),
(9, '2017-01-03 20:32:00', 'Spedito', 'Billy82', 'Via Dalmazia, 17, Lanciano (CH)', 'Lucrezia Bellanova', 6, 9),
(10, '2017-02-01 21:03:43', 'In Processazione', 'Royksopp', 'Corso Pietro Vannucci, 20, Perugia (PG)', 'Alberto Bianchi', 9, 7);
```

```
-- Inserimento Spazio
```

```
INSERT INTO Spazio
VALUES(1, 110, 'NikoBellic'),
(2, 95, 'NikoBellic'),
(3, 84, 'Nicole23'),
(4, 62, 'Giups'),
(5, 76, 'Giups'),
(6, 80, 'Giups'),
(7, 138, 'IlMoralizzatore'),
(8, 50, 'Fragolina1'),
(9, 52, 'Fragolina1'),
(10, 86, 'Mammolo83');
```

```
-- Inserimento Settore
```

```
INSERT INTO Settore
VALUES(1, 'no', 'Diretta', 12, 'sud', 1),
(2, 'si', 'Diretta', 12, 'sud', 1),
(3, 'si', "Mezz'ombra", 10, 'est', 1),
(4, 'si', 'Diretta', 8, 'nord', 3),
(5, 'no', 'Ombra', 5, 'nord', 3),
(6, 'si', 'Diretta', 12, 'sud', 4),
(7, 'no', "Mezz'ombra", 8, 'ovest', 5),
(8, 'si', 'Diretta', 11, 'sud', 5),
(9, 'si', "Mezz'ombra", 9, 'sud', 6),
(10, 'no', 'Diretta', 10, 'est', 6);
```

```
-- Inserimento Vertice
```

```
INSERT INTO Vertice
VALUES(1, 0, 0),
(2, 0, 4.5),
(3, 4.5, 4.5),
(4, 0, 6),
(5, 5.8, 12),
(6, 5.8, 0),
(7, 0, 0),
(8, 0, 12),
```

```
(9, 0.5, 0),  
(10, 0.5, 4),  
(11, 5.5, 0),  
(12, 5.5, 4);
```

```
--  
-----  
-- Inserimento Confine  
-----
```

```
INSERT INTO Confine  
VALUES(1, 1),  
(2, 1),  
(3, 1),  
(4, 4),  
(5, 4),  
(6, 4),  
(7, 4),  
(4, 5),  
(8, 5),  
(5, 5),  
(9, 6),  
(10, 6),  
(11, 6),  
(12, 6);
```

```
--  
-----  
-- Inserimento Vaso  
-----
```

```
INSERT INTO Vaso  
VALUES(1, 'Plastica', 1, 1, 0.2, 1),  
(2, 'Terracotta', 4, 4, 0.3, 1),  
(3, 'Plastica', 1, 3.5, 0.2, 1),  
(4, 'Plastica', 1.4, 2.3, 0.25, 3),  
(5, 'Cemento', 2, 2.5, 0.30, 7),  
(6, 'Terracotta', 2, 1, 0.15, 6),  
(7, 'Terracotta', 2, 5, 0.15, 6),  
(8, 'Plastica', 8.3, 2.1, 0.20, 2),  
(9, 'Cemento', 5, 5, 0.22, 7),  
(10, 'Plastica', 10.5, 5, 0.22, 7);
```

```
--  
-----  
-- Inserimento Componente_giardino  
-----
```

```
INSERT INTO Componente_giardino  
VALUES(1, 1, 1, 0.15, 8, 1),  
(2, 1, 3.5, 0.15, 8, 1),  
(3, 2, 2.5, 2.5, 1, 3),  
(4, 2.5, 6.3, 0.5, 7, 4),
```

```
(5, 1.3, 1, 0.4, 10, 8),
(6, 4.3, 1, 0.4, 10, 8),
(7, 2.1, 2.2, 1.5, 6, 7),
(8, 10.5, 5, 0.20, 5, 7),
(9, 5.41, 3.75, 1, 9, 6),
(10, 8, 8, 2, 4, 3);
```

Operazioni

```
-----
-- Operazione 1
-----
```

```
DROP PROCEDURE IF EXISTS SpazioDisponibile;

DELIMITER $$
CREATE PROCEDURE SpazioDisponibile(IN _sez INT UNSIGNED)
BEGIN
    SELECT (CapienzaMassima - NumeroPiante) as Numero
    FROM Sezione
    WHERE IDSezione = _sez;
END $$

DELIMITER ;
```

```
-----
-- Operazione 2
-----
```

```
DROP PROCEDURE IF EXISTS InserisciContenitore;

DELIMITER $$
CREATE PROCEDURE InserisciContenitore(IN _dimensione_cont VARCHAR(100),
                                       IN _altezza_pianta VARCHAR(100),
                                       IN _prezzo FLOAT,
                                       IN _pianta INT UNSIGNED,
                                       IN _ripiano INT UNSIGNED)
BEGIN
    INSERT INTO Contenitore(DimensioneContenitore, AltezzaPianta, Prezzo, IDPianta, IDRipiano)
    VALUES(_dimensione_cont, _altezza_pianta, _prezzo, _pianta, _ripiano);
END $$

DELIMITER ;
```

```
-----  
-- Operazione 3  
-----
```

```
DROP PROCEDURE IF EXISTS ElencoCosti;
```

```
DELIMITER $$  
CREATE PROCEDURE ElencoCosti(IN _utente VARCHAR(100))  
BEGIN  
    SELECT IDSpazio, Preventivo  
    FROM Spazio  
    WHERE Utente = _utente;  
END $$
```

```
DELIMITER ;
```

```
-----  
-- Operazione 4  
-----
```

```
DROP PROCEDURE IF EXISTS InserisciComponente;  
DELIMITER $$
```

```
CREATE PROCEDURE InserisciComponente(IN _x DOUBLE,  
                                      IN _y DOUBLE,  
                                      IN _raggio DOUBLE,  
                                      IN _pianta INT UNSIGNED,  
                                      IN _settore INT UNSIGNED)  
BEGIN  
    INSERT INTO Componente_giardino(CoordinataX, CoordinataY, Raggio, IDPianta, IDSettore)  
    VALUES(_x, _y, _raggio, _pianta, _settore);  
END $$  
  
DELIMITER ;
```

```
-----  
-- Operazione 5  
-----
```

```
DROP PROCEDURE IF EXISTS ElencoRilevazioni;  
DELIMITER $$
```

```
CREATE PROCEDURE ElencoRilevazioni(IN _cont INT UNSIGNED)  
BEGIN  
    SELECT *  
    FROM Rilevazione  
    WHERE IDContenitore = _cont;
```

```
END $$
```

```
DELIMITER ;
```

```
-----  
-- Operazione 6  
-----
```

```
DROP PROCEDURE IF EXISTS ElencoUtenti;  
DELIMITER $$
```

```
CREATE PROCEDURE ElencoUtenti()  
BEGIN  
    SELECT *  
    FROM Utente  
    ORDER BY Credibilita DESC;  
END $$
```

```
DELIMITER ;
```

```
-----  
-- Operazione 7  
-----
```

```
DROP PROCEDURE IF EXISTS AggiornaVoto;  
DELIMITER $$
```

```
CREATE PROCEDURE AggiornaVoto(IN _voto INT,  
                                IN _post INT UNSIGNED)  
BEGIN  
    UPDATE Post  
    SET  
        SommaVoti = SommaVoti + _voto,  
        NumeroVoti = NumeroVoti + 1  
    WHERE IDPost = _post;  
END $$
```

```
DELIMITER ;
```

```
-----  
-- Operazione 8  
-----
```

```
DROP PROCEDURE IF EXISTS ElencoPendenti;  
DELIMITER $$
```

```
CREATE PROCEDURE ElencoPendenti()
```

```

BEGIN
    SELECT o.Utente, o.Timestamp, P.Nome, P.Genere, P.Cultivar
    FROM
        Ordine o
        NATURAL JOIN
        Pianta P
    WHERE O.Stato = 'Pendente';
END $$

DELIMITER ;

```

Ridondanze e vincoli di integrità generici

/*il trigger incrementa l'attributo NumeroPiante ogni volta che viene inserita un nuovo contenitore in una sezione e impedisce l'inserimento nel caso in cui la sezione sia piena*/

```

DROP TRIGGER IF EXISTS Aggiorna_NumeroPiante_ins;

DELIMITER $$

CREATE TRIGGER Aggiorna_NumeroPiante_ins
BEFORE INSERT ON Contenitore
FOR EACH ROW
BEGIN
    DECLARE _sez INT UNSIGNED DEFAULT 0;
    DECLARE _ok INT DEFAULT 0;

    SET _ok = (
        SELECT
            IF(S.NumeroPiante < S.CapienzaMassima, 1, 0)
        FROM
            Ripiano R
            NATURAL JOIN
            Sezione S
        WHERE IDRipiano = new.ripiano
    );

    IF _ok = 0 THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'La sezione ha raggiunto la capienza massima.';
    ELSE
        UPDATE
            Sezione S
            NATURAL JOIN
            (SELECT IDSezione
             FROM Ripiano

```

```

        WHERE IDRipiano = new.IDRipiano) as D
    SET S.NumeroPiante = S.NumeroPiante + 1;
END IF;
END $$

DELIMITER ;

/*Trigger che decrementa la variabile NumeroPiante nel caso in cui una pianta della
sezione sia cancellata dal database*/

```

```

DROP TRIGGER IF EXISTS Aggiorna_NumeroPiante_del;

```

```

DELIMITER $$

```

```

CREATE TRIGGER Aggiorna_NumeroPiante_del
AFTER DELETE ON Contenitore
FOR EACH ROW
BEGIN
    UPDATE
        Sezione S
        NATURAL JOIN
        (SELECT IDSezione
        FROM Ripiano
        WHERE IDRipiano = new.IDRipiano) as D
    SET S.NumeroPiante = S.NumeroPiante - 1;
END IF;
END $$

```

```

DELIMITER ;

```

```

/*Procedura che ricalcola il valore della variabile Preventivo.
La procedura viene chiamata ogni volta che si salva una configurazione di uno spazio*/

```

```

DROP PROCEDURE IF EXISTS Aggiorna_Preventivo;

```

```

DELIMITER $$

```

```

CREATE PROCEDURE Aggiorna_Preventivo(IN _spazio INT UNSIGNED)
BEGIN
    DECLARE _pre DOUBLE DEFAULT 0;

    SET _pre = (
        SELECT SUM(P.PrezzoIndicativo)
        FROM
            Settore S
            NATURAL JOIN
            Componente_giardino CG
    )

```

```

        NATURAL JOIN
        Pianta P
    WHERE
        S.IDSpazio = _spazio);

    UPDATE Spazio
    SET Preventivo = _pre
    WHERE IDSpazio = _spazio;
END $$

DELIMITER ;

/*Trigger per aggiornare l'attributo ridondante Credibilità della tabella Utente. */

DROP TRIGGER IF EXISTS Calcola_Credibilita;

DELIMITER $$

CREATE TRIGGER Calcola_Credibilita
AFTER UPDATE ON Post
FOR EACH ROW
BEGIN
    DECLARE NumeroPost INT DEFAULT 0;
    DECLARE SommaMedie DOUBLE DEFAULT 0;
    DECLARE VotoPost DOUBLE DEFAULT 0;
    DECLARE Finito INT DEFAULT 0;

    DECLARE CursoreMedia CURSOR FOR
        SELECT SommaVoti/NumeroVoti AS Media
        FROM Post
        WHERE Autore = NEW.Autore
            AND NumeroVoti IS NOT NULL
            AND NumeroVoti <> 0;

    DECLARE CONTINUE HANDLER FOR NOT FOUND
        SET Finito = 1;

    OPEN CursoreMedia;

scan: LOOP
    FETCH CursoreMedia INTO VotoPost;

    IF Finito = 1 THEN
        LEAVE scan;
    END IF;

    SET SommaMedie = SommaMedie + VotoPost;

```



```

        SET NumeroPost = NumeroPost + 1;

END LOOP scan;
CLOSE CursoreMedia;

UPDATE Utente
SET Credibilita = SommaMedie/NumeroPost
WHERE Nickname = NEW.Autore;
END $$

/*Trigger per impedire l'inserimento di un trattamento per una pianta che è già sotto trattamento*/

DROP TRIGGER IF EXISTS Impedisci_Trattamento;

DELIMITER $$

CREATE TRIGGER Impedisci_Trattamento
BEFORE INSERT ON Quarantena
FOR EACH ROW
BEGIN

    DECLARE _alt TINYINT DEFAULT 0;
    SET _alt = (
        SELECT COUNT(*)
        FROM
            Quarantena Q
            NATURAL JOIN
            Trattamento T
        WHERE
            NEW.IDContenitore = Q.IDContenitore
            AND (T.DataFineTrattamento IS NULL
                OR CURRENT_DATE < T.DataFineTrattamento)
    );

    IF _alt <> 0 THEN
        SIGNAL SQLSTATE '45000'
        SET MESSAGE_TEXT = 'Impossibile inserire dati. La pianta è già in quarantena.'
    END IF

END $$

DELIMITER ;

```

Area Analytics

Smart Design

La funzione "Smart design" viene utilizzata durante la realizzazione del proprio modello di giardino, suggerendo delle piante secondo tre criteri scelti dall'utente

- 1 - L'utente sceglie l'indice di manutenzione desiderato. Se si sceglie un indice di manutenzione alto saranno selezionate anche le piante con un indice di manutenzione più basso
- 2 - L'utente sceglie se le piante proposte devono fiorire in periodi diversi dell'anno rispetto alle altre sezioni del giardino oppure se non è rilevante.
- 3 - L'utente sceglie il prezzo massimo delle piante che saranno suggerite

La funzione viene implementata tramite una procedura che riceve in ingresso i seguenti parametri:

- _Manutenzione: contiene l'indice di manutenzione scelto dall'utente.
- _Diversificazione_fioritura: booleano che vale 1 se il periodo di fioritura è rilevante, 0 altrimenti.
- _PrezzoMassimo: contiene il prezzo massimo delle piante suggerite.
- _Spazio: Contiene l'identificativo dello spazio che si sta progettando.

E restituisce l'elenco delle piante suggerite.

```
--  
-- SMART DESIGN  
--
```

```
DROP PROCEDURE IF EXISTS SmartDesign;  
DELIMITER $$  
CREATE PROCEDURE SmartDesign(IN _Manutenzione varchar(50),  
                             IN _Diversificazione_fioritura TINYINT,  
                             IN _PrezzoMassimo INT,  
                             IN _Spazio INT UNSIGNED)  
BEGIN  
  
    IF _Diversificazione_fioritura = 0 THEN  
        IF _Manutenzione = 'Semplice' THEN  
            SELECT *  
            FROM Pianta  
            WHERE IndiceManutenzione = 'Semplice'  
                   AND _PrezzoMassimo >= PrezzoIndicativo;  
        ELSEIF _Manutenzione = 'Medio' THEN  
            SELECT *  
            FROM Pianta  
            WHERE (IndiceManutenzione = 'Semplice'  
                   OR IndiceManutenzione = 'Medio')  
                   AND _PrezzoMassimo >= PrezzoIndicativo;  
        ELSE  
            SELECT *  
            FROM Pianta  
            WHERE  
                _PrezzoMassimo >= PrezzoIndicativo;
```

```

END IF;
ELSE
  IF _Manutenzione = 'Semplice' THEN
    SELECT *
    FROM Pianta P1
    WHERE P1.IndiceManutenzione = 'Semplice'
      AND _PrezzoMassimo >= P1.PrezzoIndicativo
      AND (
        P1.PeriodoFioritura NOT IN(
          SELECT P2.PeriodoFioritura
          FROM
            Settore S
            NATURAL JOIN
            Componente_giardino CG
            NATURAL JOIN
            Pianta P2
          WHERE S.idSpazio = _Spazio)
        OR
        P1.PeriodoFioritura IS NULL
      );
  ELSEIF _Manutenzione = 'Medio' THEN
    SELECT *
    FROM Pianta P1
    WHERE (P1.IndiceManutenzione = 'Semplice'
      OR P1.IndiceManutenzione = 'Medio')
      AND _PrezzoMassimo >= P1.PrezzoIndicativo
      AND (
        P1.PeriodoFioritura NOT IN(
          SELECT P2.PeriodoFioritura
          FROM
            Settore S
            NATURAL JOIN
            Componente_giardino CG
            NATURAL JOIN
            Pianta P2
          WHERE S.idSpazio = _Spazio)
        OR
        P1.PeriodoFioritura IS NULL
      );
  ELSE
    SELECT *
    FROM Pianta P1
    WHERE
      _PrezzoMassimo >= P1.PrezzoIndicativo
      AND (
        P1.PeriodoFioritura NOT IN(
          SELECT P2.PeriodoFioritura

```

```

FROM
    Settore S
    NATURAL JOIN
    Componente_giardino CG
    NATURAL JOIN
    Pianta P2
WHERE S.idSpazio = _Spazio)
OR
P1.PeriodoFioritura IS NULL
);

END IF;
END IF;
END $$
DELIMITER ;

```

Reporting

La funzione “Reporting” consente l’individuazione delle piante soggette con maggior frequenza a patologie e, allo stesso tempo, delle piante con numero di vendite ridotto, in modo da poter organizzare una gestione ottimizzata degli ordini. La funzione ogni mese inserisce, per ogni pianta, informazioni sul numero di acquisti effettuati e sulle patologie che le hanno colpite in una tabella che potrà essere analizzata a fine anno.

```

-- -----
-- REPORTING
-- -----

```

```

CREATE OR REPLACE VIEW Acquisti_Piante AS
SELECT P.IDPianta, IF(D.IDOrdine IS NULL, 0, COUNT(D.IDOrdine)) AS NumeroAcquisti
FROM
    Pianta P
    LEFT OUTER JOIN
    (
        SELECT IDPianta, IDORDINE
        FROM Ordine O
        WHERE MONTH(O.Timestamp) = MONTH(CURRENT_DATE)
    ) AS D USING(IDPianta)
GROUP BY IDPianta;

```

```

CREATE OR REPLACE VIEW Patologia_Piante AS
SELECT C.IDPianta,
    R.IDPatologia,
    Pat.NomePatologia,
    COUNT(*) AS FrequenzaPatologia
FROM
    Contenitore C
    NATURAL JOIN
    Rilevazione R
    NATURAL JOIN

```

```

        Patologia Pat
WHERE R.IDPatologia IS NOT NULL
      AND MONTH(R.Data) = MONTH(CURRENT_DATE)
      AND YEAR(R.Data) = YEAR(CURRENT_DATE)
GROUP BY C.IDPianta, R.IDPatologia;

CREATE TABLE IOG_Reporting(
  IDRep INT UNSIGNED NOT NULL AUTO_INCREMENT,
  Mese INT NOT NULL,
  Anno INT NOT NULL,
  IDPianta INT UNSIGNED NOT NULL,
  Nome VARCHAR(50) NOT NULL,
  Genere VARCHAR(50) NOT NULL,
  Cultivar VARCHAR(50),
  NumeroAcquisti INT NOT NULL,
  IDPatologia INT UNSIGNED,
  NomePatologia VARCHAR(50),
  FrequenzaPatologia VARCHAR(50)
  PRIMARY KEY (IDRep)
)ENGINE=InnoDB DEFAULT CHARSET=Latin1;

DROP EVENT IF EXISTS Reporting;

CREATE EVENT Reporting
ON SCHEDULE EVERY 1 MONTH
STARTS '2016-01-31 23:55:00'
DO
  INSERT INTO LOG_Reporting(Mese, Anno, IDPianta, Nome, Genere, Cultivar, NumeroAcquisti, IDPatologia,
                           NomePatologia, FrequenzaPatologia)
  SELECT
    MONTH(CURRENT_DATE) AS Mese,
    YEAR(CURRENT_DATE) AS Anno,
    P.IDPianta,
    P.Nome,
    P.Genere,
    P.Cultivar,
    AP.NumeroAcquisti,
    PP.IDPatologia,
    PP.NomePatologia,
    PP.FrequenzaPatologia
  FROM
    Acquisti_Piante AP
  LEFT OUTER JOIN
    Patologia_Piante PP USING(IDPianta)
  NATURAL JOIN
    Pianta P;

```

Indagini statistiche

La funzione “Indagini Statistiche” prende in ingresso una patologia e ritorna l’elenco di tutte le piante da essa colpite. La lista viene arricchita con altre informazioni relative alla sezione, alla serra, e al ripiano in cui è posizionata la pianta, permettendo al personale addetto alla manutenzione di stabilire se le condizioni presenti favoriscono l’insorgere della patologia richiesta.

```
-- -----  
-- INDAGINI STATISTICHE  
-- -----  
  
DROP PROCEDURE IF EXISTS IndaginiStatistiche;  
  
DELIMITER $$  
  
CREATE PROCEDURE IndaginiStatistiche(IN _p VARCHAR (100))  
BEGIN  
SELECT  
    se.NomeSerra,  
    sz.NomeSezione,  
    sz.Umidita as UmiditaSezione,  
    sz.Temperatura as TemperaturaSezione,  
    pa.NomePatologia,  
    ri.Data as DataRilevazione,  
    ri.ConcentrazioneElementi  
FROM  
    Serra se  
    NATURAL JOIN  
    Sezione sz  
    NATURAL JOIN  
    Ripiano rp  
    NATURAL JOIN  
    Contenitore c  
    INNER JOIN  
    Rilevazione ri USING(IDContenitore)  
    NATURAL JOIN  
    Patologia pa  
WHERE pa.NomePatologia = _p  
ORDER BY pa.NomePatologia, ri.Data, se.NomeSerra;  
  
END $$  
DELIMITER ;
```