

Elaborato per il corso  
«Basi di Dati »  
A.A. 2019/2020



# MAPESI RESTAURANT

Penazzi Andrea

Marchitelli Daniele

Singh Simaranjeet

Il nostro database è stato realizzato per la catena di ristoranti Mapesi con il fine di gestire la parte economica e gestionale e per rendicontare entrate ed uscite contabili. Come prima cosa ci siamo dedicati alla realizzazione dello schema Entity-Relationship, successivamente abbiamo effettuato una ristrutturazione di tale schema (togliendo la gerarchia) del nostro progetto. Abbiamo poi implementato la base di dati appoggiandoci al programma Ampps per realizzare e gestire database in SQL.

# 1. Definire l'obiettivo del database 1

Il nostro obiettivo è stato quello di creare una base di dati che potesse essere utile agli amministratori della catena Mapesi per contabilizzare e gestire il personale, fornitori e banchetti. Tale database sarà accessibile esclusivamente dai proprietari, che decideranno poi cosa fare delle informazioni ricavate. Fissato il dominio del nostro progetto, ci siamo concentrati su quali informazioni implementare nella nostra base di dati.

La nostra è una catena di ristoranti in provincia di Brescia, che si occupa di banchetti comprendenti dalle 10 alle 20 persone ciascuno.

## 1. Definire l'obiettivo del database 2

Dato che il banchetto è prenotato da un solo cliente, ogni ordine verrà visto come realizzato dalla stessa persona che potrà però effettuare più ordini.

La scelta viene effettuata su 4 tipologie di menù già preimpostati che comprenderanno un antipasto, un primo, un secondo ed un dolce.

I nostri ristoranti sono composti da un buon numero di dipendenti che si divideranno in dipendenti di sala e di cucina.

Vista la vicinanza tra i nostri ristoranti, non sarà necessario avere più di un magazzino, quindi ogni ristorante si rifornirà dallo stesso, ed ogni fornitore approvvigionerà tutti i ristoranti.

## 1. Definire l'obiettivo del database 3

La contabilità del nostro database fa riferimento al mese di giugno 2020.

I ristoranti prevedono come giorno di chiusura il lunedì.



## 2. Analisi dei requisiti

Non avendo contatti con un effettivo gestore di una catena di ristoranti, abbiamo fissato noi i requisiti del progetto (ovvero le funzionalità che avremmo voluto che la nostra base di dati fornisse), partendo dalle nostre conoscenze del mondo della ristorazione.

## 2. Analisi dei requisiti

In base a queste nostre conoscenze abbiamo ritenuto che la nostra base di dati dovesse fornire informazioni sui seguenti elementi dei ristoranti, a nostro parere fondamentali: personale, fornitori, clienti(banchetti), magazzino, ingredienti, portate, menu e ordini.



### 3. Organizzare le informazioni

Successivamente abbiamo deciso di quali informazioni tenere traccia per ogni elemento ritenuto fondamentale:

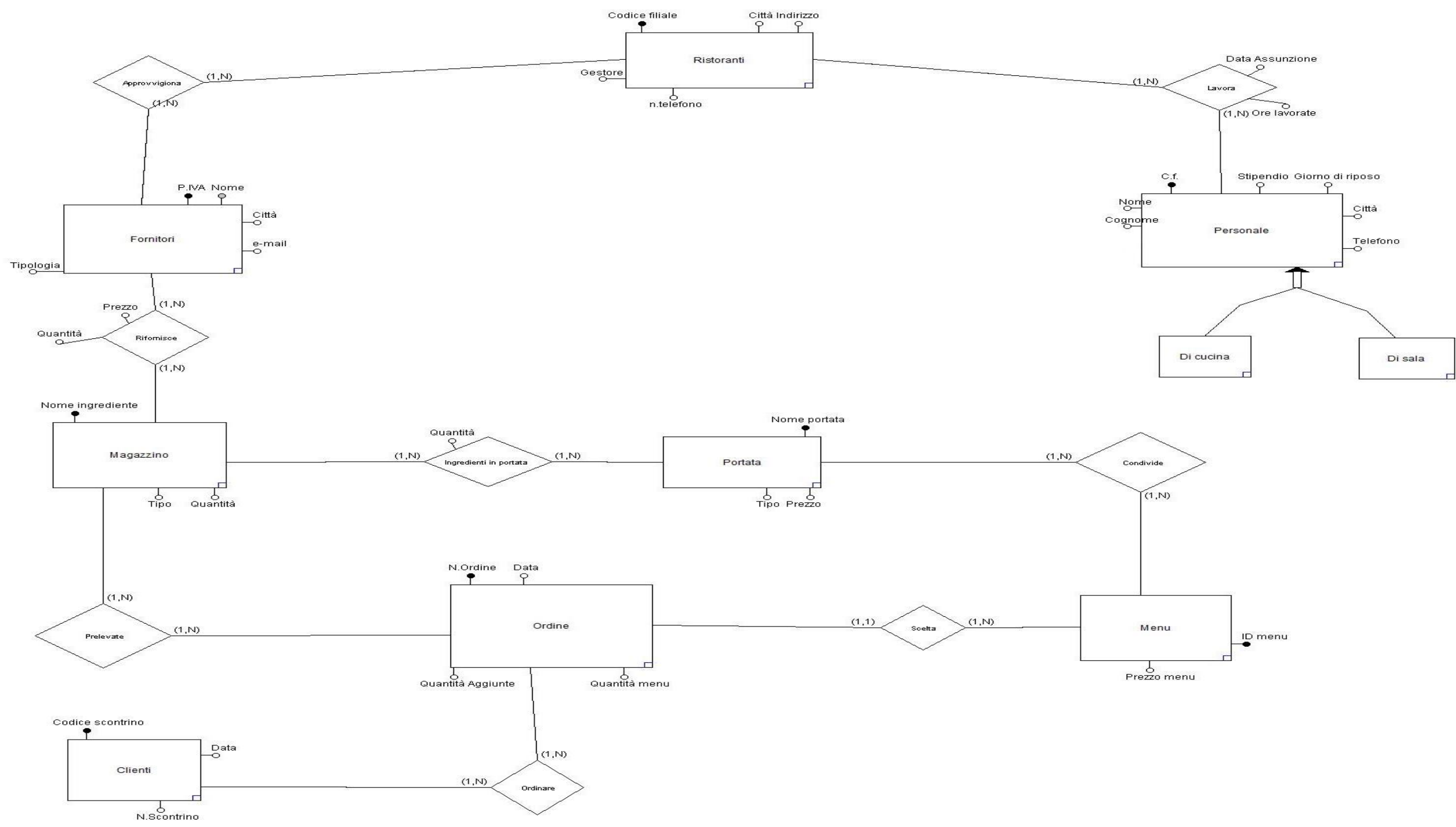
- Ristoranti: codice filiale, gestore(identificato dal codice fiscale), città, indirizzo e e-mail
- Personale: codice fiscale, codice dipendente, nome, cognome, ruolo, città , giorno di riposo, paga oraria e telefono
- Fornitori: P. Iva, nome, e-mail, tipologia(fornitura) e città
- Magazzino: nome ingrediente, tipo(di ingrediente) e quantità
- Menù: Id menù, nome menù e prezzo menù
- Portata: Id menù, nome portata, tipo portata e prezzo
- Ordine: N. Ordine, data, menù, quantità menù e quantità aggiunte
- Clienti: N. Scontrino, Id scontrino e data

## 4. Modello E-R

Conclusa l'organizzazione delle informazioni (che ci è stata utile anche come *Analisi del dominio*) siamo stati in grado di tradurre tutto ciò in uno schema E-R (Entity-Relationship): fondamentale per la progettazione di un database, in quanto è la base per la creazione del Modello Logico che verrà effettivamente implementato in un DBMS.

Abbiamo quindi identificato tutte le entità e le associazioni in gioco (entrambe con i loro eventuali attributi), rappresentandole secondo la notazione del Modello E-R, assegnando i nomi più significativi possibili e definendo le chiavi primarie e le cardinalità delle associazioni.

Di seguito lo schema risultante:



Ci siamo avvalsi di alcuni costrutti per migliorare la comprensione del nostro modello:

Gerarchia: per il personale. Totale(t) in quanto l'unione delle entità figlie mi dà l'intero insieme del personale ed Esclusiva(e) siccome non può esistere del personale appartenente a più di un entità figlia.

Entità deboli: le entità che dipendono da altre entità alle quali sono associate (Menù, Ristoranti ecc..)

## 5. Ristrutturazione 1

Prima di ricavare lo Schema Logico, è necessario ristrutturare lo schema ER, in quanto alcuni costrutti che abbiamo utilizzato, in particolare la gerarchia, non può essere tradotta da un modello all'altro.

Per eliminare la gerarchia abbiamo usato la tecnica del collasso verso l'alto: non avendo, i figli, degli attributi, l'entità padre non li “assorbe”, ma ottiene l'attributo *Ruolo*, un selettore per indicare di che tipo di personale si tratti.

## 5. Ristrutturazione 2

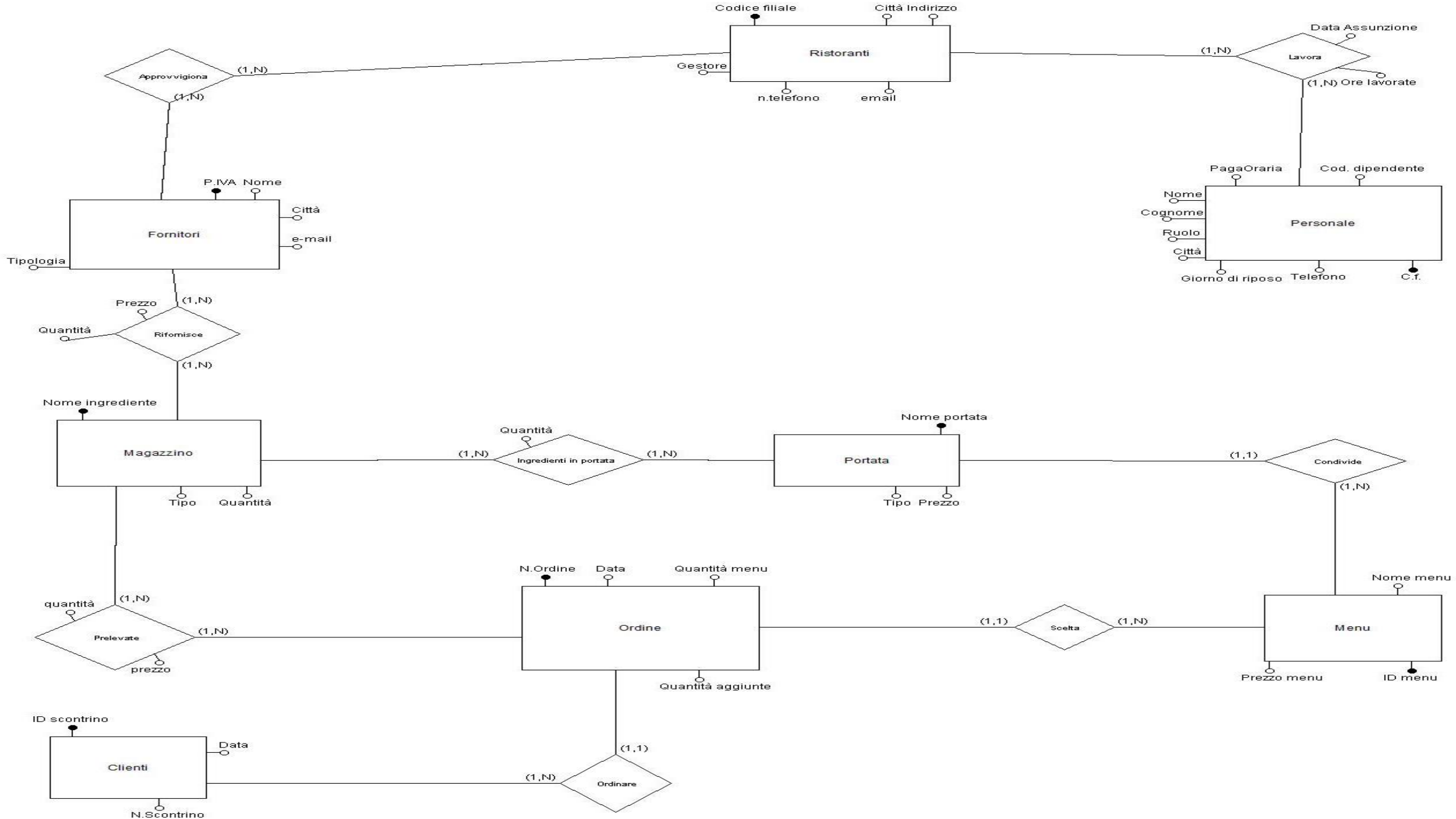
Durante l'ideazione del modello logico, ci siamo accorti che alcune considerazioni non erano congruenti alla nostra idea iniziale di progetto, di conseguenza abbiamo modificato come di seguito:

Il menu condivide una ed una sola portata.

Aggiunta di attributi (quantità e prezzo) in prelevate.

Visto che il nostro ristorante si occupa di banchetti, ogni ordine verrà visto come effettuato da un solo cliente.

Le modifiche al nostro ER sono riportate nel seguente schema.





## 6 Progettazione logica 1

Dopo aver sviluppato il modello concettuale (E-R) nella forma adatta, abbiamo la possibilità di procedere con la progettazione logica, in modo da ottenere un modello logico pronto per l'implementazione in un DBMS (in particolare, un *modello relazionale*). Entità e associazioni vengono tradotte in Relazioni che saranno le tabelle del nostro database.

## 6 Progettazione logica 2

Nelle associazioni con cardinalità (1,1) si possono accorpare le relazioni:

Scelta, condivide ed ordinare. Esse sono state accorpate dall'entità con cardinalità (1,1) che, in pratica, acquisisce anche la chiave dell'altra entità (in quanto un'associazione viene tradotta in una relazione contenente le chiavi delle entità che collega).

## PROGETTO LOGICO

- Ristoranti(CFiliale, gestore, città, indirizzo, telefono, e-mail)
- Lavora(CodiceDip, cf, CFiliale, data assunzione, ore lavorate)
- Personale(cf, CodiceDip, nome, cognome, ruolo, città, giorno riposo, paga oraria, telefono)
- Approvvigiona(CFiliale, P.Iva)
- Fornitori(P.Iva, nomi, e-mail, tipologia, città)
- Rifornisce(P.Iva, nome ingrediente, quantità rifornita, prezzo totale)
- Magazzino(nome ingrediente, tipo, quantità)
- IngredientiInPortata(Nome ingrediente, Nome portata, quantità)
- Portata(IDMenù, nome portata, tipo portata, prezzo)
- Menu(IDMenù, Nome menù, prezzo menù)
- Ordine(N.Ordine, Menù, data, quantità menù, quantità aggiunte)
- Prelevate(quantità, nome ingrediente, N.Ordine, prezzo unitario)
- Clienti(N.Scontrino, ID scontrino, data, N.Ordine)

IN VERDE: vincoli di integrità referenziale (foreign keys -> le chiavi di altre relazioni). Da associazioni o accorpamenti.

## 7. Vincoli di integrità referenziale

Nello schema logico riportato nella slide precedente i Vincoli di integrità referenziale sono indicati con l'evidenziazione (in verde) dell'attributo che fa riferimento ad un'altra relazione.

Per evidenziare meglio questi collegamenti riportiamo una schermata ottenuta successivamente all'implementazione della basi di dati con un DBMS, ma che evidenzia i vincoli in modo chiaro:

v	mapesi fornitori
	PIVA : int(11)
	Nome : varchar(30)
	Email : varchar(40)
	Tipologia : varchar(40)
	Città : varchar(30)

v	mapesi approvvigiona
	CodiceFiliale : int(11)
	PIVA : int(11)

v	mapesi ristoranti
	CodiceFiliale : int(11)
	GestoreCF : varchar(16)
	Città : varchar(40)
	Indirizzo : varchar(40)
	Telefono : int(10)
	e-mail : varchar(20)

v	mapesi lavora
	CodiceDip : varchar(2)
	CF : varchar(16)
	CodiceFiliale : int(11)
	DataAssunzione : date
	OreLavorate : int(3)

v	mapesi rifornisce
	PIVA : int(11)
	NomeIngrediente : varchar(40)
	QuantitàRifornita : int(11)
	PrezzoTotale : int(11)

v	mapesi personale
	CF : varchar(16)
	CodiceDipendente : varchar(2)
	Nome : varchar(30)
	Cognome : varchar(30)
	Ruolo : enum('sala','cucina')
	Città : varchar(20)
	GiornoRiposo : enum('Martedì','Mercoledì','Giovedì','Venerdì','Sabato','Domenica')
	PagaOraria : double
	Telefono : bigint(11)

v	mapesi magazzino
	NomeIngrediente : varchar(40)
	Tipo : enum('Carne','Pesce','Spezia','Bevanda','Verdura','Frutta','Generico')
	Quantità : decimal(10,0)

v	mapesi prelevate
	Quantità : int(11)
	NomeIngrediente : varchar(40)
	N.Ordine : int(11)
	prezzoUnitario : int(2)

v	mapesi ordine
	N.Ordine : int(11)
	IDScontrino : int(11)
	data : date
	Menu : int(3)
	QuantitàMenu : int(2)
	QuantitàAggiunte : int(2)

v	mapesi clienti
	N.Scontrino : int(11) unsigned
	IDScontrino : int(11)
	Data : date

v	mapesi ingredientiinportata
	NomeIngrediente : varchar(40)
	NomePortata : varchar(50)
	Quantità : int(6)

v	mapesi portata
	IDMenu : int(3)
	NomePortata : varchar(50)
	TipoPortata : enum('Antipasto','Primo','Secondo','Dolce')
	Prezzo : decimal(10,0)

v	mapesi menu
	IDMenu : int(3)
	NomeMenù : varchar(50)
	PrezzoMenu : decimal(10,0)

## 8 DBMS

Come *DataBase Management System* abbiamo scelto di utilizzare MySQL, in quanto il nostro prodotto non richiederà prestazioni altissime e la quantità di personale che lo utilizzerà sarà limitata: potrebbe bastare infatti un solo responsabile.

Per quanto riguarda l'inserimento dei dati, abbiamo inserito la contabilizzazione di un mese a titolo esemplificativo.

## 9. Query

Sulla base delle richieste del nostro cliente, abbiamo creato alcune query che forniscono ,aggiornano o eliminano dati sul sito.

Di seguito riportiamo alcune delle query prodotte direttamente nel sito internet.



## 1)Query ricerca personale

Questa query è utilizzata per la ricerca del personale e mostra solo quello che gli è stato domandato di ricercare altrimenti mostra l'intera tabella personale

```
if(isset($_POST['search'])){
    $searchKey = $_POST['search'];
    $sql = "SELECT * FROM personale AS p
    INNER JOIN lavora AS l
    ON p.CF = l.CF
    WHERE Nome LIKE '%$searchKey%' OR CodiceFiliale
    LIKE '%$searchKey%' OR CodiceDipendente LIKE '%$searchKey%'
    OR Ruolo LIKE '%$searchKey%' OR GiornataRiposo LIKE '%$searchKey%'";
}else{
    $sql = "select * from personale as p
    INNER JOIN lavora as l
    ON p.CF = l.CF
    $searchKey = """;
}
```

## 2)Query ricerca fornitori

Funziona come la query del personale qui sopra

```
if(isset($_POST['search'])){
    $searchKey = $_POST['search'];
    $sql = "SELECT * FROM fornitori AS f
    WHERE Nome LIKE '%$searchKey%' OR Tipologia LIKE '%$searchKey%'";
}else{
    $sql = "select * from fornitori as f";
    $searchKey = "";
}
$result = mysqli_query($conn, $sql);
?>
```

cucina	Search						
Nome	Cognome	Ruolo	CodiceFiliale	CodiceDipendente	GiornoRiposo	Operazioni	
Flavio	Bergel	cucina	74011	E1	Sabato	<button>Aggiorna</button>	<button>Elimina</button>
Fulgenzia	Brangini	cucina	74011	F1	Domenica	<button>Aggiorna</button>	<button>Elimina</button>
Benigna	Chieffi	cucina	74014	B4	Mercoledì	<button>Aggiorna</button>	<button>Elimina</button>
Delfino	Chiappellone	cucina	74012	B2	Mercoledì	<button>Aggiorna</button>	<button>Elimina</button>
Ambrogio	Cistriani	cucina	74013	E3	Sabato	<button>Aggiorna</button>	<button>Elimina</button>
Manfredo	Frydel	cucina	74012	C2	Giovedì	<button>Aggiorna</button>	<button>Elimina</button>
Olivia	Cinola	cucina	74011	D1	Venerdì	<button>Aggiorna</button>	<button>Elimina</button>

Bevande	Search			
PIVA	Nome	Email	Tipologia	Città
1267270748	Marangi Giovanni	marangigiovanni@gmail.it	Bevande	LUMEZZANE (BS)
2147270748	Franciacorta	franciacorta@gmail.it	Bevande	PASSIRANO (BS)

### 3)Elimina dipendente

Query per eliminare dipendenti dal sito

```
$cf = $_GET['id'];  
$query = "delete from personale where CF = '$cf'";  
$data = mysqli_query($conn, $query);
```

### 4)Login

login nella parte "privata" del database

```
$query = mysqli_query($conn, "select * from personaldata where user='$user' and pass='$pass'");
```

### 5)Update personale

query per modificare i dati del personale

```
$sql = "update personale set CF='$cf', CodiceDipendente='$CodiceDipendente', Nome='$Nome',  
Cognome='$Cognome', Ruolo='$Ruolo', Citta='$Citta', GiornoRiposo='$GiornoRiposo',  
PagaOraria='$PagaOraria', Telefono='$Telefono' WHERE CF='$cf'";
```

```
$query = mysqli_query($conn, $sql);
```

```
if($query){
```

```
    $sql2 = "update lavora set CF='$cf', CodiceDip='$CodiceDipendente', CodiceFiliale='$CodiceFiliale',
```

```
DataAssunzione='$DataAssunzione', OreLavorate='$OreLavorate' WHERE CF='$cf'";
```

```
    mysqli_query($conn, $sql2);
```

```
}
```

### 6)Update Magazzino

query per modificare i prodotti in magazzino

```
$sql = "update magazzino set NomeIngrediente='$nomeIng', Tipo='$tipo',  
Quantità='$quantità' WHERE NomeIngrediente='$nomeIng'";
```

```
$query = mysqli_query($conn, $sql);
```

## Inserire informazione da aggiornare

## Inserire informazione da aggiornare

## 7) Aggiunta personale

Query per inserire nuovo personale

```
$sql = "insert into personale values ('$cf', '$CodiceDipendente', '$Nome', '$Cognome', '$Ruolo', '$Città', '$GiornoRiposo', '$PagaOraria', '$Telefono')";  
$query = mysqli_query($conn, $sql);
```

```
if($query){  
    $sql2 = "insert into lavora values ('$CodiceDipendente', '$cf', '$CodiceFiliale', '$DataAssunzione', '$OreLavorate')";  
    mysqli_query($conn, $sql2);  
}
```

## 8)aggiunta fornitori

Query per aggiungere nuovi fornitori con l'inserimento di nuovi dati

```
$sql = "insert into fornitori (PIVA, Nome, Email, Tipologia, Città) values ('$Piva', '$Nome', '$Email', '$Tipologia', '$Citta')";
```

### Inserire informazione richiesta

Partita iva	Nome
E-mail	Carne
Desenzano	Aggiungere

### Inserire informazione richiesta

CF
CodiceDipendente
Alessandro
Rossi
sala
Città
Martedì
PagaOraria
Telefono
CodiceFiliale
DataAssunzione
OreLavorate
Aggiungere

### 9)Visualizza nome ed indirizzo della catena

```
$sql = "SELECT *from ristoranti;";  
$result = mysqli_query($conn, $sql);  
$resultCheck = mysqli_num_rows($result);
```

```
if($resultCheck > 0){  
while($row = mysqli_fetch_assoc($result)){  
$branch[] = $row['Città'];  
$addr[] = $row['Indirizzo'];
```

### 10)Visualizza ordini

```
$ds = "SELECT * FROM ordine ";  
$result = mysqli_query($conn, $ds);
```

### 11)Visualizza magazzino

```
$sql = "SELECT * FROM magazzino ";  
$result1 = mysqli_query($conn, $sql);
```

### 12)Visualizza menù diviso per portata

questa query è utilizzata per far visualizzare ogni portata per ogni menu.

In questo caso specifico mostra la portata antipasto

ma la query è stata scritta per ogni tipo di portata

```
$m = "SELECT * FROM `portata` as p join menu as m  
on p.idmenu = m.IDMenu where TipoPortata= 'Antipasto';"  
$queryM = mysqli_query($conn, $m);
```

## La Nostra Catena Di Ristoranti

Brescia

via Roma, 25

Manerbio

via dell'assunta, 15

Salò





Via Vecchia filanda,21

Sarezzo

via Aldo Moro, 5

N.ORDINE	DATA	MENU	QuantitàMenu
1	2020-06-02	4	17
2	2020-06-02	4	10
3	2020-06-02	2	11

Nome Ingrediente	Tipo	Quantità	Operazioni
Acqua	Bevanda	75	<button>Aggiorna</button>
Baccalà	Pesce	0	<button>Aggiorna</button>
Pizza	Bevanda	50	<button>Aggiorna</button>

 Antipasto	 Primo	 Secondo	 Dolce
<b>Mari e monti</b> Carbonara <i>Uova, Guanciale, Pasta, Pecorino,</i>			
€25		<b>Bresciano</b> Casoncelli al brasato <i>Pasta, Vino Rosso, Carne Chianina, cipolla, Uova, olio extravergine,</i>	€30
<b>Cacciatore</b> Gnocchi di patate alla romana <i>Patate, Carne Chianina, Uova, pomodoro,</i>		<b>Il carnivoro</b> Risotto con le rane <i>Rane, Riso, cipolla,</i>	€30
€25			

### 13)menù del mese

questa query è utilizzata per mostrare quale menù è stato venduto di più nel mese corrente

```
$piùVenduto = "select s.NomeMenù, s.PrezzoMenu, MAX(s.total) as maximum
                from(SELECT Menu, NomeMenù, PrezzoMenu, sum(QuantitàMenu) total
                FROM `ordine` as o join menu as m on o.Menu = m.IDMenu GROUP BY Menu) as s
                GROUP by s.NomeMenù, s.PrezzoMenu
                ORDER by maximum DESC LIMIT 1";

$queryMenu =$conn ->query($piùVenduto);
```

### 14)Costo mensile per tutto il personale

Query utilizzata per controllare il costo totale del personale

```
$sql = "select * from personale as p
        INNER JOIN lavora as l
        ON p.CF = l.CF
        order by Nome";

$result = mysqli_query($conn, $sql);
$resultCheck = mysqli_num_rows($result);
if ($resultCheck > 0) {
    while ($row = mysqli_fetch_assoc($result)) {
        $res += $row['OreLavorate'] * $row['PagaOraria'] . "<br>";    }}


```

### 15)Valore fornitura

query utilizzata per vedere il costo totale del rifornimento delle materie che saranno poi utilizzate per preparare i menù

```
$sql1 = "select * from rifornisce";
$result1 = mysqli_query($conn, $sql1);
$resultCheck1 = mysqli_num_rows($result1);
if ($resultCheck1 > 0) {
    while ($row1 = mysqli_fetch_assoc($result1)) {
        // $res1 += $row1['QuantitàRifornita'] * $row1['PrezzoTotale'] . "<br>";
        $res1 += $row1['PrezzoTotale'] . "<br>";    }}


```

## This Month's Menu

Mari e monti

€ 95

Costo Mensile Personale

77998 €

Valore Fornitura

1401 €

## 16)Guadagno totale

query utilizzata per visualizzare il guadagno che proviene dalla vendita di tutti i menù

```
$guadagno = "SELECT Menu, NomeMenù, PrezzoMenu, sum(QuantitàMenu) FROM `ordine` as o join menu  
as m on o.Menu = m.IDMenu GROUP BY Menu";
```

```
$guadagnoQuery = mysqli_query($conn, $guadagno);
```

```
$guad = mysqli_num_rows($guadagnoQuery);
```

```
if ($guad > 0) {  
    while ($row2 = mysqli_fetch_assoc($guadagnoQuery)) {  
        $guadagoTotale += $row2['PrezzoMenu'] * $row2['sum(QuantitàMenu)'] . "<br>";  
    }  
}
```

## 17)Ingredienti in portata

Query utilizzata per visualizzare gli ingredienti delle portate

In questo caso specifico mostra la portata antipasto

ma la query è stata scritta per ogni tipo di portata

```
$ingredient = "SELECT * FROM `portata` as p join ingredientiinportata as i on  
p.NomePortata = i.NomePortata";
```

```
$queryIng = mysqli_query($conn, $ingredient);
```

Guadagno Totale

158220 €

### Mari e monti

Contorno misto in agrodolce

*Zucchina, Melanzana, peperone  
giallo, cipolla,*

€30