

Ministero dell'istruzione e del merito

A038 - ESAME DI STATO CONCLUSIVO DEL SECONDO CICLO DI ISTRUZIONE
Indirizzo ITIA - INFORMATICA E TELECOMUNICAZIONI ARTICOLAZIONE "INFORMATICA"
(Testo valevole anche per gli indirizzi quadriennali IT32 e ITIT) Disciplina: INFORMATICA
Sessione suppletiva 2023
Seconda prova scritta **Disciplina:** INFORMATICA

Il candidato svolga la prima parte della prova e due tra i quesiti proposti nella seconda parte.

PRIMA PARTE

La società SportOvunque, proprietaria di diverse palestre di fitness in alcune città d'Italia, vuole lanciare un servizio innovativo che dia la possibilità alle persone che viaggiano di poter accedere, con un unico abbonamento, ad attività offerte da palestre gestite anche da altre società, in tutta Italia. A tale scopo SportOvunque ha preso accordi con palestre interessate ad aderire a questa iniziativa su tutto il territorio nazionale. Le palestre che hanno aderito praticeranno tariffe di ingresso agevolate riservate agli abbonati di SportOvunque.

L'abbonamento a questo servizio può essere acquistato sul portale web di SportOvunque, dove i potenziali clienti possono registrarsi pagando on-line la relativa quota annuale.

Ogni palestra che aderisce all'iniziativa dovrà caricare sul portale i propri dati, comprensivi di indirizzo, città e coordinate geografiche. Dovrà poi caricare anche la propria offerta di fitness: per ogni tipologia standardizzata di attività (es: Sala macchine, GAG, Zumba, Pilates ecc...) resa disponibile agli abbonati, indicherà i giorni della settimana (lunedì, martedì, ...) in cui le attività vengono svolte, ciascuna con gli orari in cui possono essere prenotate, il relativo prezzo convenzionato e il numero massimo di posti disponibili.

Un abbonato potrà prenotare attività presso una qualsiasi palestra tra quelle presenti sul portale: individuerà la palestra, l'attività, il giorno della settimana e l'orario, e sceglierà la data specifica di suo interesse. Se in quella data sono già state effettuate prenotazioni in numero pari al massimo dei posti disponibili per quell'attività, l'attività non sarà prenotabile. Altrimenti la prenotazione verrà regolarmente registrata, e l'abbonato pagherà on-line il relativo prezzo a SportOvunque; quest'ultima provvederà poi a versare mensilmente, alle varie palestre che hanno aderito al servizio, gli incassi relativi alle attività prenotate.

Il candidato, fatte le opportune ipotesi aggiuntive, sviluppi:

- a) un'analisi della realtà di riferimento, giungendo alla definizione di uno schema concettuale della base di dati che, a suo motivato giudizio, sia idoneo a gestire la realtà presentata;
- b) il relativo schema logico;
- c) la definizione in linguaggio SQL di un sottoinsieme delle relazioni della base di dati in cui siano presenti alcune di quelle che contengono vincoli di integrità referenziale e/o vincoli di dominio, se esistenti;
- d) le interrogazioni espresse in linguaggio SQL che restituiscono:
 - a. l'elenco delle attività disponibili in una data città il mercoledì, indicando, per ciascuna attività, la palestra, la tipologia di attività, l'orario di svolgimento, il numero massimo di posti disponibili, ed il prezzo, ordinate per palestra;
 - b. l'importo totale delle prenotazioni pagate dagli abbonati nel mese di aprile, che SportOvunque dovrà versare ad una determinata palestra;
 - c. la classifica annuale delle palestre di una certa città, ordinata in base al numero di attività prenotate;
- e) il progetto di massima della struttura funzionale dell'applicazione web per la gestione della realtà sopra presentata;
- f) una parte significativa dell'applicazione web che consente l'interazione con la base di dati, utilizzando appropriati linguaggi a scelta sia lato client che lato server.

SECONDA PARTE

- I. Con riferimento al tema presentato nella prima parte, si supponga di offrire ad un abbonato la possibilità di disdire la prenotazione di un'attività fino a 48 ore prima del suo svolgimento, ricevendo sul portale un credito pari al costo dell'attività che viene disdetta. Tale credito potrà essere utilizzato per successive prenotazioni. Si descrivano le modifiche da apportare alla base di dati e la struttura delle pagine web dedicate a tale funzionalità, scrivendo in un linguaggio a scelta una porzione di codice significativa di tali pagine.
- II. In relazione al tema proposto nella prima parte, si realizzi, in un linguaggio a scelta, una pagina dinamica che consenta all'utente di ottenere l'elenco in forma testuale delle palestre di una certa città. In alternativa, si rappresenti tale insieme di palestre attraverso una mappa geografica di quella città con la loro geolocalizzazione.
- III. Si descrivano le differenze, anche attraverso esempi, tra i predicati ANY, ALL ed EXISTS del linguaggio SQL
- IV. Si consideri la seguente tabella:

Nome	Cognome	Data di nascita	Città di nascita	Società Sportiva	Tel-Squadra	Allenatore	Tel-Allenatore
Mario	Rossi	23-04-2004	Taranto	Atleti Taranto	099123123	Vittorio	3331122331
Luca	Verdi	07-06-2001	Milano	Corridori Milanesi	02987987	Bruno	3501236786
Bianca	Rosa	01-01-2001	Perugia	Atleti Perugini	057893541	Raffaele	3475641830
Lorenzo	Verdi	04-05-2002	Roma	Roma sportivi	06549183	Giulio	3465921344
Manuela	Verdi	09-06-2002	Roma	Roma sportivi	06549183	Maria	3468712533
Umberto	Neri	01-03-2001	Taranto	Atleti Taranto	099123123	Vittorio	3331122331

Il candidato verifichi le proprietà di normalizzazione e proponga uno schema equivalente che rispetti la terza Forma Normale, motivando le scelte effettuate.

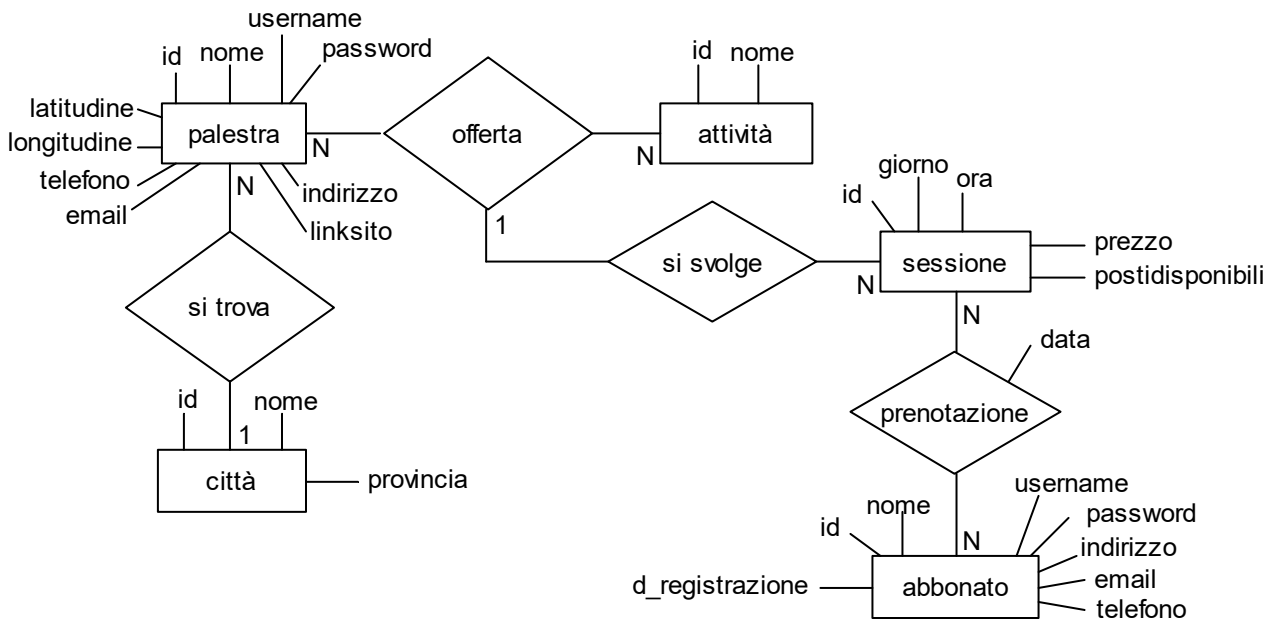
Durata massima della prova: 6 ore. È consentito soltanto l'uso dei manuali di riferimento dei linguaggi di programmazione (language reference) e di calcolatrici scientifiche e/o grafiche purché non siano dotate di capacità di calcolo simbolico. È consentito l'uso del dizionario bilingue (italiano-lingua del paese di provenienza) per i candidati di madrelingua non italiana. Non è consentito lasciare l'Istituto prima che siano trascorse 3 ore dalla consegna della traccia.

Soluzione prima parte

La traccia richiede la realizzazione di un portale web che consente alle persone che viaggiano di usufruire, con un unico abbonamento, delle attività di fitness offerte in tutta Italia dalle palestre gestite dalla società "SportOvunque" e da altre società che aderiscono all'iniziativa.

Schema concettuale (Diagramma E/R)

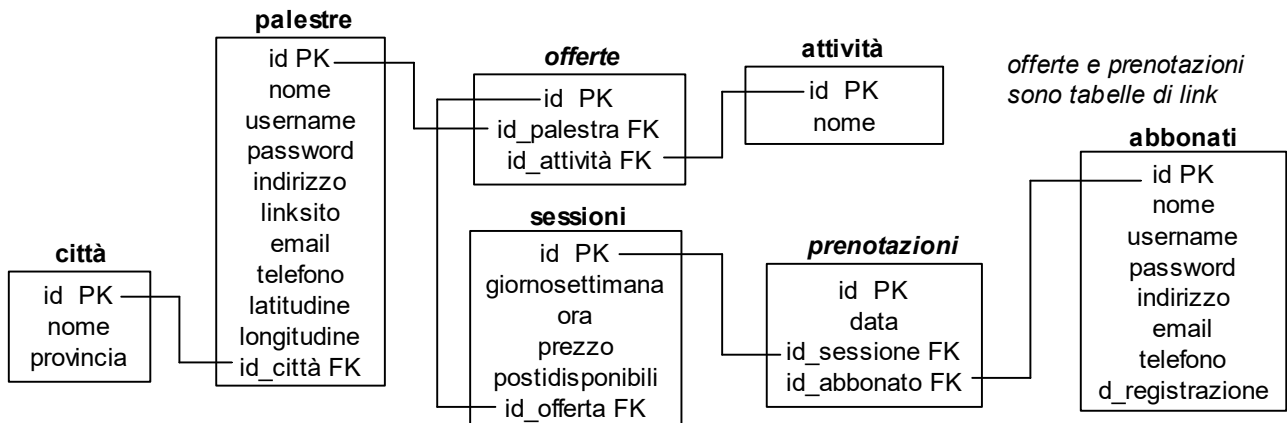
Per semplicità gli attributi delle entità sono quelli più importanti e strettamente necessari per implementare le interrogazioni richieste nella traccia



Letture del diagramma E/R

- Ogni palestra si trova in una città, in ogni città ci sono N palestre
- Ogni palestra offre N attività, ogni attività è offerta da N palestre
- Ogni attività offerta si svolge in N sessioni, ogni sessione è relativa ad una attività offerta
- Ogni abbonato prenota N sessioni, ogni sessione è prenotata da N abbonati

Schema logico



Definizione delle relazioni in linguaggio SQL

Struttura delle tabelle

Si fa riferimento al database MySql Nome del database: sportovunque

tabella	campi	key	tipo	lung	Descrizione/note
città	id	PK	Int		Contatore auto_increment
	nome		VarChar	40	
	provincia		Char	2	Sigla della provincia
palestre	id	PK	Int		Contatore auto_increment
	nome		VarChar	40	Nome della palestra
	username		VarChar	20	
	password		VarChar	32	Si utilizza la codifica MD5()
	indirizzo		VarChar	40	
	linksito		VarChar	60	Link al sito web
	email		VarChar	30	
	telefono		VarChar	30	
	latitudine		Decimal	10,6	
	longitudine		Decimal	10,6	
attività	id_città	FK	Int		Fa riferimento a città.id
	id	PK	Int		Contatore auto_increment
attività	nome		VarChar	40	
	id	PK	Int		Contatore auto_increment
offerte	id_palestra	FK	Int		Fa riferimento a palestre.id
	id_attività	FK	Int		Fa riferimento ad attività.id
	id	PK	Int		Contatore auto_increment
sessioni	giornosettimana		Int		1 Lunedì, 2 Martedì, 3 Mercoledì, 4 Giovedì 5 Venerdì, 6 Sabato, 7 Domenica
	ora		VarChar	16	Ad es. 13:30 – 14:30
	prezzo		Int		Ad es. 12 euro
	postidisponibili		Int		
	id_offerta	FK	Int		Fa riferimento ad offerte.id
	id	PK	Int		Contatore auto_increment
abbonati	nome		VarChar	40	
	username		VarChar	20	
	password		VarChar	32	Si utilizza la codifica MD5()
	indirizzo		VarChar	60	
	email		VarChar	30	
	telefono		VarChar	30	
	d_registrazione		Date		Data di registrazione al portale
	id	PK	Int		Contatore auto_increment
prenotazioni	data		Date		
	id_sessione	FK	Int		Fa riferimento a sessioni.id
	id_abbonato	FK	Int		Fa riferimento ad abbonati.id

Query MySql per la creazione del database e delle tabelle

```
CREATE DATABASE sportovunque;
```

```
CREATE TABLE IF NOT EXISTS città (  
id INT AUTO_INCREMENT PRIMARY KEY,  
nome VARCHAR(40),  
provincia CHAR(2));
```

```
CREATE TABLE IF NOT EXISTS palestre(  
id INT AUTO_INCREMENT PRIMARY KEY,  
nome VARCHAR(40),  
username VARCHAR(20),  
password VARCHAR(32),  
indirizzo VARCHAR(40),  
linksito VARCHAR(60),  
email VARCHAR(30),  
telefono VARCHAR(30),  
latitudine DECIMAL(10,6),  
longitudine DECIMAL(10,6),  
id_città INT,  
FOREIGN KEY (id_città) REFERENCES città(id));
```

```
CREATE TABLE IF NOT EXISTS attività (  
id INT AUTO_INCREMENT PRIMARY KEY,  
nome VARCHAR(40));
```

```
CREATE TABLE IF NOT EXISTS offerte (  
id INT AUTO_INCREMENT PRIMARY KEY,  
id_palestra INT,  
id_attività INT,  
FOREIGN KEY (id_palestra) REFERENCES palestre(id),  
FOREIGN KEY (id_attività) REFERENCES attività(id));
```

```
CREATE TABLE IF NOT EXISTS sessioni(  
id INT AUTO_INCREMENT PRIMARY KEY,  
giornosettimana INT,  
ora VARCHAR(16),  
prezzo INT,  
postidisponibili INT,  
id_offerta INT,  
FOREIGN KEY (id_offerta) REFERENCES offerte(id));
```

```
CREATE TABLE IF NOT EXISTS abbonati(  
id INT AUTO_INCREMENT PRIMARY KEY,  
nome VARCHAR(40),  
username VARCHAR(20),  
password VARCHAR(32),  
indirizzo VARCHAR(60),  
email VARCHAR(30),  
telefono VARCHAR(30),  
d_registrazione DATE );
```

```
CREATE TABLE IF NOT EXISTS prenotazioni (  
id INT AUTO_INCREMENT PRIMARY KEY,  
data DATE,  
id_sessione INT,  
id_abbonato INT,  
FOREIGN KEY (id_sessione) REFERENCES sessioni(id),  
FOREIGN KEY (id_abbonato) REFERENCES abbonati(id));
```

Interrogazioni espresse in linguaggio SQL

Le query sono state testate utilizzando MySql e PhpMyAdmin di Wamp Server 3.3.0 64 bit

1. l'elenco delle attività disponibili in una data città il mercoledì, indicando, per ciascuna attività, la palestra, la tipologia di attività, l'orario di svolgimento, il numero massimo di posti disponibili, ed il prezzo, ordinate per palestra;

```
Select palestre.nome as 'Palestra' , città.nome as 'Città', città.provincia as 'Pr',
attività.nome as 'Attività', sessioni.ora as 'Mercoledì',
sessioni.postidisponibili as 'Max posti', sessioni.prezzo as 'Prezzo'
From palestre,città, attività,offerte,sessioni
Where palestre.id_città=città.id
And offerte.id_attività=attività.id
And offerte.id_palestra=palestre.id
And sessioni.id_offerta=offerte.id
And città.nome='Teramo'
And sessioni.giornosettimana=3
Order by palestre.nome, attività.nome;
```

Palestra	Città	Pr	Attività	Mercoledì	Max posti	Prezzo
Non solo Pilates	Teramo	TE	Ginnastica ritmica	8:30-9:30	15	15
Non solo Pilates	Teramo	TE	Ginnastica ritmica	9:30-10:30	15	15
SportOvunque	Teramo	TE	Basket	15:30-16.30	10	20
SportOvunque	Teramo	TE	Basket	16:30-17:30	10	20
SportOvunque	Teramo	TE	Sala macchine	8:30-9:30	40	12
SportOvunque	Teramo	TE	Sala macchine	9:30-10.30	40	12
SportOvunque	Teramo	TE	Sala macchine	10:30-11:30	40	12
SportOvunque	Teramo	TE	Sala macchine	11.30-12:30	40	12
SportOvunque	Teramo	TE	Sala macchine	12:30-13:30	40	12
SportOvunque	Teramo	TE	Sala macchine	13:30-14:30	40	12
SportOvunque	Teramo	TE	Sala macchine	14:30-15:30	40	12

2. l'importo totale delle prenotazioni pagate dagli abbonati nel mese di aprile, che SportOvunque dovrà versare ad una determinata palestra;

Supponiamo che la palestra sia "Non solo Pilates" di Teramo che ha id=7

```
Select palestre.nome as 'Palestra' , città.nome as 'Città', città.provincia as 'Pr',
sum(sessioni.prezzo) as 'Totale prenotazioni pagate nel mese di aprile 2025'
From palestre,città, sessioni,prenotazioni
Where palestre.id_città=città.id
And sessioni.id=prenotazioni.id_sessione
And palestre.id=7
and prenotazioni.data BETWEEN '2025-04-01' AND '2025-04-30';
```

Palestra	Città	Pr	Totale prenotazioni pagate nel mese di aprile 2025
Non solo Pilates	Teramo	TE	105

3. la classifica annuale delle palestre di una certa città, ordinata in base al numero di attività prenotate;

```
Select palestre.nome,città.nome,città.provincia,  
count(*) as 'Numero attività svolte a Teramo nel 2025'  
From palestre,città, offerte, sessioni,prenotazioni  
Where palestre.id_città=città.id  
And offerte.id_palestra=palestre.id  
And sessioni.id_offerta=offerte.id  
And sessioni.id=prenotazioni.id_sessione  
And città.nome='Teramo'  
and prenotazioni.data BETWEEN '2025-01-01' AND '2025-12-31'  
Group by palestre.nome
```

nome	nome	provincia	Numero attività svolte a Teramo nel 2025
Non solo Pilates	Teramo	TE	161
SportOvunque	Teramo	TE	69

Progetto di massima della struttura dell'applicazione web per la gestione della realtà sopra presentata

Un'applicazione web:

- c) è un software basato su un'architettura client-server
- d) risiede su un server remoto e non deve essere installata sul dispositivo dell'utente
- e) viene eseguita online su un qualsiasi dispositivo (pc, tablet, smartphone) dotato di un browser e di una connessione di rete (Internet o Intranet aziendale)
- f) è accessibile tramite il protocollo HTTP o HTTPS
- g) offre evidenti vantaggi ma la scrittura del codice presenta difficoltà e criticità dovute alla natura stessa del web che è stato inizialmente concepito per la presentazione di contenuti statici

Struttura e funzionalità dell'applicazione web

- **Lato client** (Frontend), ovvero l'interfaccia che consente all'utente di interagire con l'applicazione. Per la sua realizzazione utilizziamo principalmente HTML5, CSS e Javascript. Per rendere le pagine interattive e professionali utilizziamo AJAX
- **Lato server** (Backend), ovvero il cuore dell'applicazione che si connette al database per leggere e scrivere i dati, elabora le richieste del client e restituisce le risposte. Tra i molteplici linguaggi di programmazione disponibili, scegliamo di utilizzare PHP
- **Database** che memorizza e gestisce i dati dell'applicazione web. Utilizziamo il Database MySql

Il server remoto che ospita l'applicazione, considerando l'elevato numero di utenti potenziali, deve essere potente, performante e in grado di garantire la disponibilità, l'affidabilità e la sicurezza dei dati necessarie.

Riguardo alla funzionalità, i profili indicati nella traccia sono due: le palestre che aderiscono al progetto SportOvunque e offrono i propri servizi di fitness e gli abbonati di SportOvunque che in tutta Italia possono prenotare online le attività presso le palestre presenti sul portale

Un terzo profilo di cui tener conto è quello dei tecnici che gestiscono i dati delle tabelle comuni (città, attività standardizzate), amministrano l'applicazione web e forniscono assistenza a palestre e abbonati.

L'accesso all'applicazione è consentito esclusivamente con una procedura di login.

L'applicazione prevede funzionalità avanzate quali il pagamento online, le ricerche tramite mappe cliccabili, filtri, ordinamenti e report; sono inoltre previste le opzioni di utilità del recupero password e della chat in tempo reale con il servizio di assistenza tecnica del portale.

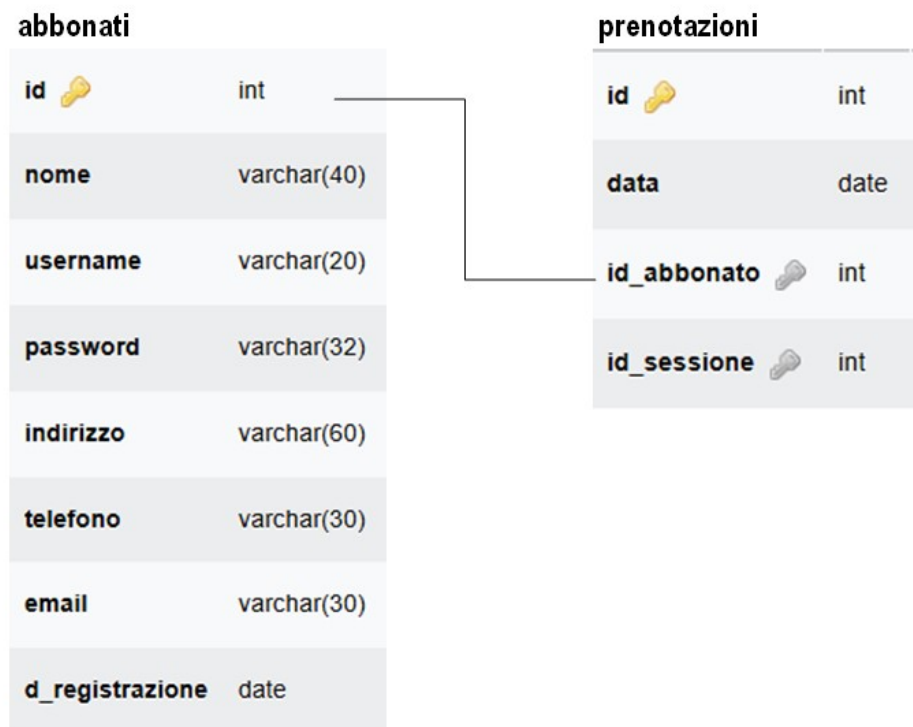
Una parte significativa dell'applicazione web che consente l'interazione con la base di dati, utilizzando appropriati linguaggi a scelta sia lato client che lato server

Scegliamo di codificare la pagina che visualizza l'elenco degli abbonati che a partire da una certa data fornita in input non hanno effettuato alcuna prenotazione. A questa funzionalità, utile per fini di controllo e di marketing, possono accedere i tecnici e le palestre.

Il codice, composto da due file **displayabbonati.html** e **trovaabbonati.php**, è molto essenziale e non utilizza alcun css. Manca inoltre la parte che verifica se l'utente è autorizzato o meno ad accedere alla pagina stessa.

Database MySql: **sportovunque**

Tablette coinvolte: **abbonati e prenotazioni**



Linguaggi utilizzati: **HTML5, JavaScript, AJAX, PHP**

displayabbonati.html

```
<!DOCTYPE html>
<html lang="it">
<head>
  <title>Display abbonati SportOvunque</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <script src="http://code.jquery.com/jquery-latest.js"></script>
  <script
    src="//ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>
</head>
<script type="text/javascript">
$(document).ready(function() {
  $('#reset').click(function(){
    $('#risposta').empty(); //il reset deve pulire anche div risposta
  });
  $('#home').click(function(){
    var url = "index.php";
    //$(location).attr('href',url);// reindirizza ad esempio a index.php
  });
  $('#submit').click(function(){
    $('#risposta').empty();//pulizia di div risposta
    var d1=$('#d1').val();
    if (d1==""){ d1="2025-01-01"; // data iniziale limite
    }
    $.ajax({ //chiamata AJAX di jquery
      type: "POST",
      url: "trovaabbonati.php",data: "d1="+d1,dataType: "html",
      success: function (msg){
        $('#risposta').html(msg);
      }
    });
    return false;
  });
});
</script>
<body style="background-color:#fafafa">
  <h3>Abbonati che non hanno effettuato prenotazioni</h3>
  <form name="form1" id=" form" method="post" action="">
    <label>La pagina fornisce l'elenco degli abbonati al portale<br/>
    SportOvunque che a partire da una data iniziale fornita in <br/>
    input non hanno effettuato prenotazioni</label><br/><br/>
    <label>Dalla data iniziale</label><br/>
    <input type="date" id="d1" style="width:120px" ><br/><br/><br/>
    <button type="submit" id="submit" style="width:62px">Submit</button>
    <button type="reset" id="reset" style="width:62px">Reset</button>
    <button type="button" id="home" style="width:62px">Home</button>
    <br/><br/>
    <div id="risposta"> </div><br/>
  </form>
</body>
</html>
```

trovaabbonati.php

```
<style>
  td {border: 1 px solid #444444}
  table{border-collapse: collapse}
</style>
<?php
  $conn=mysqli_connect("localhost","root","","sportovunque");
  $d1=$_POST["d1"];
  $d=date_create($d1);
  $ds=date_format($d,"d/m/Y"); // trasformo la data nel formato d/m/Y
  echo "<h3>Abbonati che dal ". $ds." non hanno effettuato
prenotazioni</h3>";
  ?>
  <style>
    table, th, td {
      border: 1px solid;
    }
  </style>
  <table width="50%">
    <tr bgcolor="lightyellow">
      <td width="3%">Nr</td><td width="24%">Nome</td>
      <td width="37%">Indirizzo</td><td width="12%">Telefono</td>
      <td width="22%">Email</td>
    </tr>
  </table>
  <?php
  $s1="Select * from abbonati order by nome";// Seleziono tutti gli abbonati
  $q1=mysqli_query($conn,$s1);
  $k=0;
  while($r = mysqli_fetch_array($q1))
  { //per ogni abbonato verifico se ha effettuato prenotazioni dopo la
    data iniziale
    $id_abbonato=$r[0];
    $s2="Select data from prenotazioni
      Where id_abbonato=$id_abbonato
      And data>='$d1'";
    $q2=mysqli_query($conn,$s2);
    $numorighe=mysqli_num_rows($q2);
    if($numorighe==0) // se non c'è alcuna prenotazione dalla data
    iniziale
    {
      $k++;
      echo"<tr><td>$k</td><td>$r[nome]</td><td>$r[indirizzo]</td>
      <td>$r[telefono]</td><td>$r[email]</td></tr>";
    }
  }
  echo"</table>";
  ?>
```

Test e spiegazione del codice

Per il test del codice ho utilizzato Wamp Server 3.3.0 64bit, PHP 8.0.26, MySql 8.0.31 e PHpMyAdmin 5.2.0.

- Lato client (displayabbonati.html): l'utente (tecnico del portale o palestra) all'interno del form di inserimento digita la data iniziale e preme il bottone Submit. JavaScript, tramite una richiesta asincrona AJAX, invia la data al server PHP.

Abbonati che non hanno effettuato prenotazioni

La pagina fornisce l'elenco degli abbonati al portale SportOvunque che a partire da una data iniziale fornita in input non hanno effettuato prenotazioni

Dalla data iniziale

gg/mm/aaaa

Submit

Reset

Home

- Lato server: lo script trovaabbonati.php riceve il valore della data iniziale tramite la variabile `$_POST["..."]`, esegue le query di selezione necessarie e restituisce una tabella come risposta
- Lato client: la risposta viene mostrata nell'elemento `<div id="risposta"> </div>` del form senza che la pagina HTML5 venga ricaricata

Abbonati che non hanno effettuato prenotazioni

La pagina fornisce l'elenco degli abbonati al portale SportOvunque che a partire da una data iniziale fornita in input non hanno effettuato prenotazioni

Dalla data iniziale

15/05/2025

Submit

Reset

Home

Abbonati che dal 15/05/2025 non hanno effettuato prenotazioni

Nr	Nome	Indirizzo	Telefono	Email
1	Guido lavespa	Piazza buia snc Roma	444989898	guidolavespa@yyy.com
2	Remo la barca	Via del canto libero, 28 Milano	888979797	remolabarca@xxx.it

Soluzione seconda parte

- II. In relazione al tema proposto nella prima parte, si realizzi, in un linguaggio a scelta, una pagina dinamica che consenta all'utente di ottenere l'elenco in forma testuale delle palestre di una certa città. In alternativa, si rappresenti tale insieme di palestre attraverso una mappa geografica di quella città con la loro geolocalizzazione

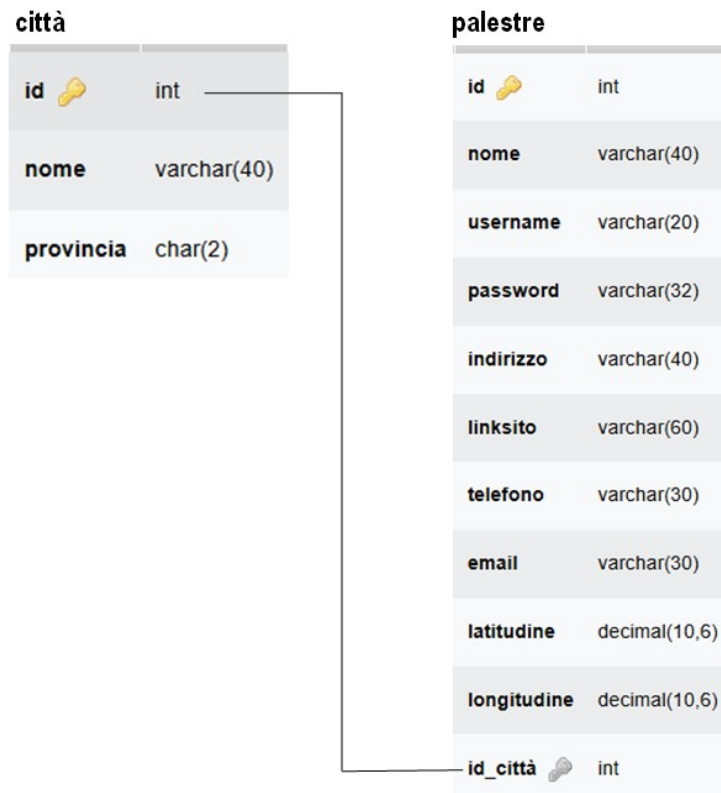
La pagina dinamica che vogliamo realizzare è composta dal file **elencocittà.php**, tramite il quale selezioniamo la città (l'estensione è php perché occorre utilizzare il linguaggio PHP per popolare la select città), e dal file **elencopalestre.php** che esegue le query necessarie per restituire l'elenco desiderato.

Coerentemente con il codice proposto nella soluzione della prima parte, anche in questo caso utilizziamo AJAX che consente lo scambio in background dei dati tra i due file ed evita il ricaricamento della pagina HTML. Con AJAX, JavaScript inoltra una richiesta GET o POST al server, riceve ed elabora la risposta e aggiorna la pagina corrente. L'utente non si accorge che qualcosa è stato trasmesso al server e l'applicazione risulta più fluida, veloce e interattiva.

Riguardo alla richiesta alternativa di rappresentare l'insieme di palestre attraverso una mappa geografica della città, invito a consultare su questo stesso sito www.maurodeberardis.it la proposta di soluzione della seconda prova scritta di Informatica dell'Esame di Stato 2023 - Sessione straordinaria ("Bancadati mieli italiani")

Database MySql: **sportovunque**

Tabelle coinvolte: **città, palestre**



Linguaggi utilizzati: **HTML5, JavaScript, AJAX, PHP**

elencocittà.php

```
<!DOCTYPE html>
<html lang="it">
<head>
  <title>Elenco città SportOvunque</title>
  <meta charset="utf-8">
  <meta name="viewport" content="width=device-width, initial-scale=1">
  <script src="http://code.jquery.com/jquery-latest.js"></script>
  <script
src="//ajax.googleapis.com/ajax/libs/jquery/1.11.1/jquery.min.js"></script>
</head>
<script type="text/javascript">
$(document).ready(function() {
  $('#reset').click(function(){
    $('#risposta').empty(); //il reset deve pulire anche div risposta
  });
  $('#home').click(function(){
    var url = "index.php";
    //$(location).attr('href',url);// reindirizza ad esempio a index.php
  });
  $('#submit').click(function(){
    var cit=$('#cit').val();
    if(cit==0){
      alert("Selezionare la città");
      return false;
    }
    //chiamata ajax di jquery.
    //Chiamata asincrona che evita il ricaricamento della pagina
    $.ajax({
      type: "POST",
      url: "elencopalestre.php",data: "cit="+cit,dataType: "html",
      success: function (msg){
        $('#risposta').html(msg);
      }
    });
    return false;
  });
});
</script>
<body>
  <h1>Elenco palestre per città</h1>
  <form name="form1" id=" form" method="post" action="">
    <label>Città</label><br/>
    <select name="cit" id="cit" style="width:200px">
    <?php
      //si connette al database
      $conn=mysqli_connect("localhost","root","", "sportovunque");
      // Legge le righe della tabella città
      $s = "Select id,nome,provincia From città Order By nome";
      $q=mysqli_query($conn,$s);
      // popola la select città con i dati della tabella
      echo '<option value="0">Seleziona la città</option>';
```

```
while($r = mysqli_fetch_array($q)) {
    echo '<option value="'. $r[0].'">'. $r[1].' ('. $r[2].')</option>';
}
mysqli_close($conn);
?>

</select><br/><br/>
<button type="submit" id="submit" style="width:62px">Submit</button>
<button type="reset" id="reset" style="width:62px">Reset</button>
<button type="button" id="home" style="width:62px">Home</button>
<br/><br/><div id="risposta"> </div><br/>
</form>
</body>
</html>
```

Test e spiegazione del codice

Per il test del codice ho utilizzato Wamp Server 3.3.0 64bit, PHP 8.0.26, MySQL 8.0.31 e PHPMyAdmin 5.2.0.

- Lato client (elencocittà.php): l'utente tramite la select "città" seleziona la città. La select viene popolata tramite codice PHP prendendo i dati dalla tabella "città" del database MySQL "sportovunque". JavaScript, tramite una richiesta asincrona AJAX, invia l'id della città al server PHP.

Elenco palestre per città

Città

- Lato server: elencopalestre.php riceve i valori dell'id città attraverso la variabile \$_POST["..."], esegue le query che consentono di ottenere l'elenco desiderato e stampa una tabella come risposta
- Lato client: la risposta viene mostrata nell'elemento `<div id="risposta"> </div>` del form senza che la pagina HTML5 venga ricaricata

Elenco palestre per città

Città

Teramo (TE) Palestre SportOvunque

Nome	Indirizzo	Latitudine-Longitudine	Telefono	Email
Non solo Pilates	Salita ripida 260	3333.333333-7777.777777	888888888	pilateste@zzz.com
SportOvunque	Viale stretto 25	1111.111111-222.222222	222222222	sportovunque@xxx.it

Cliccando sul nome di ciascuna riga, tramite il riferimento al campo linksito, viene visualizzato il sito web della palestra selezionata

III. Si descrivano le differenze, anche attraverso esempi, tra i predicati ANY, ALL ed EXISTS del linguaggio SQL

- ALL** viene utilizzato per confrontare un valore con tutti i risultati di una subquery
ANY viene utilizzato per confrontare un valore con almeno un risultato di una subquery
EXISTS viene utilizzato per verificare se una subquery restituisce almeno una riga

Per gli esempi relativi alle differenze tra ANY e ALL facciamo riferimento alle tabelle impiegati e operai

impiegati

id	Nome	stipendioannuale
1	Rossi Paolo	37000
2	Bianchi Paolo	32870
3	Verdi Primo	33450
4	Marroni Carlo	28400

operai

id	Nome	stipendioannuale
1	Pinco Pallino	27100
2	Labarca Remo	24570
3	Taglia Primo	31050
4	De Michelis Michele	24990
5	Berardi Bruno	28700

**SELECT * FROM impiegati
WHERE stipendioannuale > ALL(SELECT stipendioannuale from operai);**

Seleziona gli impiegati il cui stipendio è maggiore di tutti gli stipendi nella tabella operai

id	Nome	stipendioannuale
1	Rossi Paolo	37000
2	Bianchi Paolo	32870
3	Verdi Primo	33450

**SELECT * FROM operai
WHERE stipendioannuale > ANY(SELECT stipendioannuale from impiegati);**

Seleziona gli operai il cui stipendio è maggiore di almeno uno stipendio nella tabella impiegati

id	Nome	stipendioannuale
3	Taglia Primo	31050
5	Berardi Bruno	28700