




## SQL per le applicazioni

Esercitazione PHP e MySQL  
Svolgimento

DBG

1

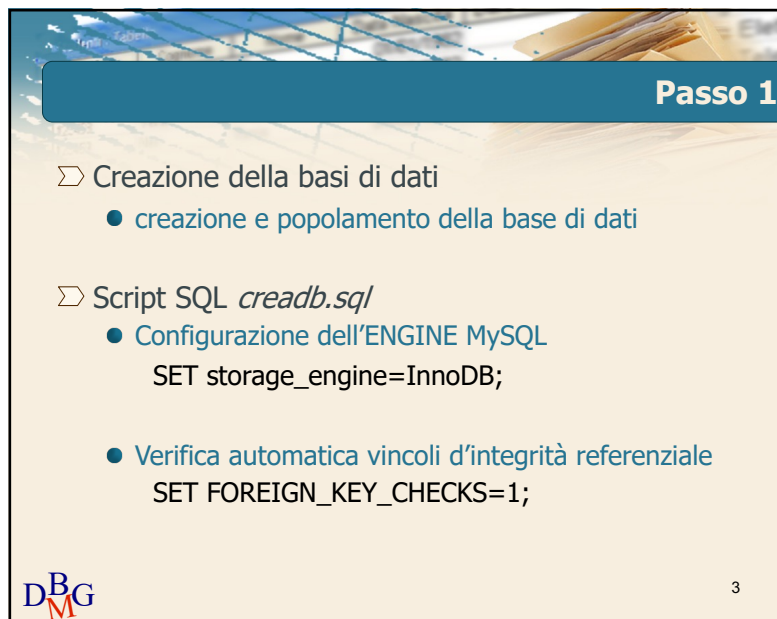


## Passi di risoluzione

- ▷ Passo 1
  - creazione e popolamento della base di dati
    - Creazione di un script SQL
- ▷ Passo 2
  - creazione di una query d'interrogazione
  - creazione di una query di aggiornamento
- ▷ Passo 3
  - creazione della pagina Web statica (client-side) con form d'interrogazione
- ▷ Passo 4
  - inserimento dello script PHP per l'interfacciamento dinamico al DB (server-side)

DBG

2



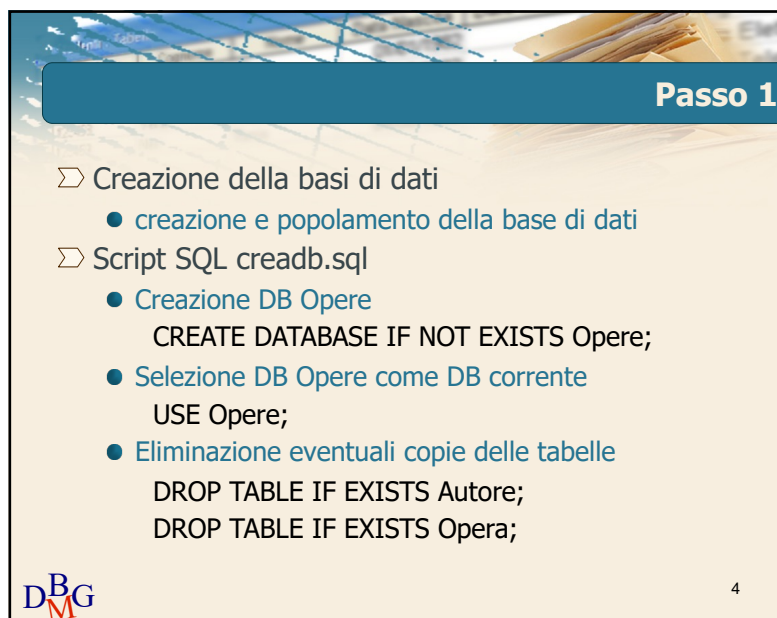
**Passo 1**

- ▷ Creazione della basi di dati
  - creazione e popolamento della base di dati
- ▷ Script SQL *creadb.sql*
  - Configurazione dell'ENGINE MySQL  
SET storage\_engine=InnoDB;
  - Verifica automatica vincoli d'integrità referenziale  
SET FOREIGN\_KEY\_CHECKS=1;

DBG  
M

3

3



**Passo 1**

- ▷ Creazione della basi di dati
  - creazione e popolamento della base di dati
- ▷ Script SQL *creadb.sql*
  - Creazione DB Opere  
CREATE DATABASE IF NOT EXISTS Opere;
  - Selezione DB Opere come DB corrente  
USE Opere;
  - Eliminazione eventuali copie delle tabelle  
DROP TABLE IF EXISTS Autore;  
DROP TABLE IF EXISTS Opera;

DBG  
M

4

4


**Passo 1**

> Creazione della basi di dati
 

- creazione e popolamento della base di dati

> Script SQL *creadb.sql*

- Disattivazione commit automatico ad ogni istruzione  
SET AUTOCOMMIT = 0;
- Creazione transazione di creazione tabelle  
START TRANSACTION;  
*istruzioni di creazione delle tabelle*  
COMMIT;
- Creazione transazione di popolamento tabelle  
START TRANSACTION;  
*istruzioni di popolamento delle tabelle*  
COMMIT;


5

5


**Passo 1**

> Creazione della basi di dati
 

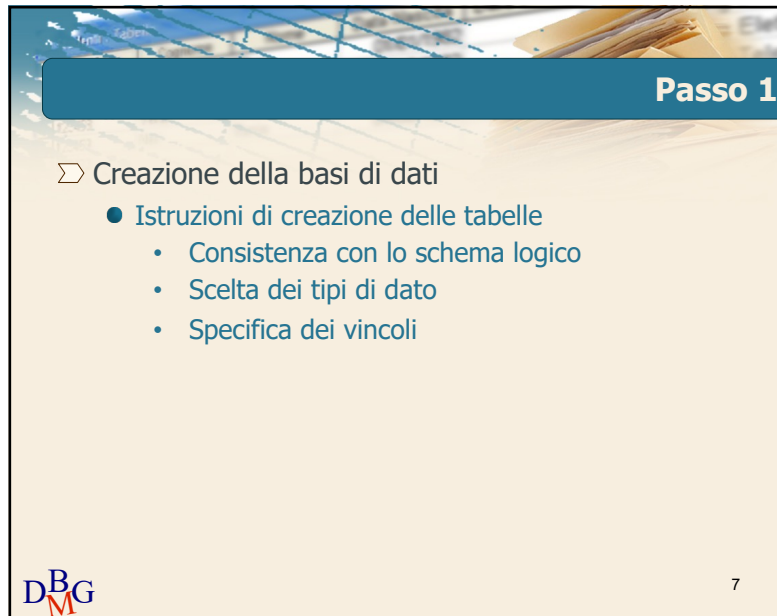
- creazione e popolamento della base di dati

```

SET storage_engine=InnoDB;
SET FOREIGN_KEY_CHECKS=1;
CREATE DATABASE IF NOT EXISTS Opere;
USE Opere;
DROP TABLE IF EXISTS Autore;
DROP TABLE IF EXISTS Opera;
SET AUTOCOMMIT=0;
START TRANSACTION;
... istruzioni di creazione delle tabelle...
COMMIT;
START TRANSACTION;
... istruzioni di popolamento delle tabelle...
COMMIT;
    
```


6

6



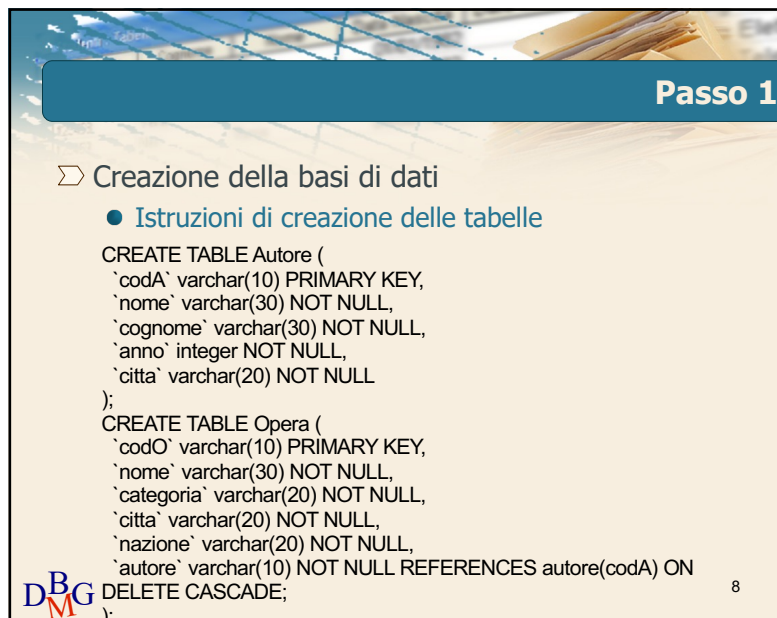
**Passo 1**

➤ Creazione della basi di dati

- Istruzioni di creazione delle tabelle
  - Consistenza con lo schema logico
  - Scelta dei tipi di dato
  - Specifica dei vincoli

DBG 7

7



**Passo 1**

➤ Creazione della basi di dati

- Istruzioni di creazione delle tabelle

```
CREATE TABLE Autore (  
  `codA` varchar(10) PRIMARY KEY,  
  `nome` varchar(30) NOT NULL,  
  `cognome` varchar(30) NOT NULL,  
  `anno` integer NOT NULL,  
  `citta` varchar(20) NOT NULL  
);  
CREATE TABLE Opera (  
  `codO` varchar(10) PRIMARY KEY,  
  `nome` varchar(30) NOT NULL,  
  `categoria` varchar(20) NOT NULL,  
  `citta` varchar(20) NOT NULL,  
  `nazione` varchar(20) NOT NULL,  
  `autore` varchar(10) NOT NULL REFERENCES autore(codA) ON  
DELETE CASCADE;  
);
```

DBG 8

8



## Passo 1

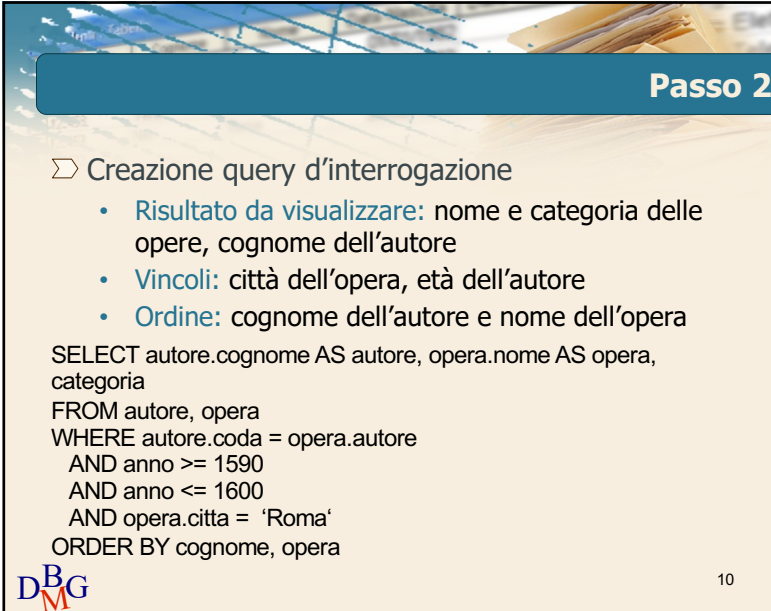
➤ Creazione della basi di dati

- Istruzioni di popolamento delle tabelle

```
INSERT INTO Autore VALUES ('1', 'Gian Lorenzo', 'Bernini',  
1598, 'Napoli');  
INSERT INTO Autore VALUES ('2', 'Francesco', 'Borromini',  
1599, 'Bissone');  
  
INSERT INTO Opera VALUES ('1', 'Apollo e Dafne', 'scultura',  
'Roma', 'Italia', '1');  
INSERT INTO Opera VALUES ('2', 'Baldacchino S.Pietro',  
'architettura', 'Roma', 'Italia', '1');  
INSERT INTO Opera VALUES ('3', 'Fontana dei fiumi',  
'architettura', 'Roma', 'Italia', '1');  
INSERT INTO Opera VALUES ('4', 'S.Ivo la Sapienza',  
'architettura', 'Roma', 'Italia', '2');
```

 9

9




## Passo 2

➤ Creazione query d'interrogazione

- Risultato da visualizzare: nome e categoria delle opere, cognome dell'autore
- Vincoli: città dell'opera, età dell'autore
- Ordine: cognome dell'autore e nome dell'opera

```
SELECT autore.cognome AS autore, opera.nome AS opera,  
categoria  
FROM autore, opera  
WHERE autore.coda = opera.autore  
AND anno >= 1590  
AND anno <= 1600  
AND opera.citta = 'Roma'  
ORDER BY cognome, opera
```

 10

10

## Passo 2

- ▷ Creazione query d'aggiornamento
  - Inserimento nuovo autore X
  - Inserimento nuova opera Y dell'autore X
- ▷ Una transazione composta da due istruzioni
  - Inserimento nuovo autore X
  - Inserimento nuova opera Y dell'autore X

```
START TRANSACTION;
INSERT INTO Autore (CodA, Nome, Cognome, Anno, Città) VALUES('X',
..., ..., ...);

INSERT INTO Opera (CodO, Nome, Categoria, Città, Nazione, CodA)
VALUES ('Y', ....., ....., 'X');
COMMIT;
```

11

11

## Passo 3

- ▷ creazione pagina Web statica
  - Scrittura pagina HTML con editor di testo (ad es. notepad, HAPedit)

```
01 <html>
02
03 <head>
04 <title>Esempio di prova</title>
05 ...
06 </head>
07
08 <body bgcolor="#FFFFFF">
09 ...
10 ...
11 </body>
12
13 </html>
```

Intestazione (header)

Contenuto (body)

12

12

Passo 3

➤ Creazione form di accesso al DB

- Scrittura pagina HTML con editor di testo (ad es. notepad, HAPedit)

**Impostare anno di nascita dell'autore e città interessata**

Data di nascita  
compresa tra


anno

e anno

Città:

**Form contenente:**

- Due campi di testo (anni di inizio e fine)
- Un menù a tendina (città dell'opera)
- Un button Submit
- Un button Reset


13

13

Passo 3

➤ Creazione pagina Web statica

- Metodo: GET
- Action: *test.php* (vedi Passo 4)

```

1 <html>
2 <head>
3 <title>Opere</title>
4 </head>
5 <body>
6 <h4> Impostare anno di nascita dell'autore e città interessata </h4>
7
8 <form method="get" action="test.php">
9
10
```


14

14

**Passo 3**

➤ Creazione pagina Web statica

```

11 <table>
12
13 <tr> Data di nascita compresa tra </tr>
14
15 <tr> <td> anno </td>
16 <td> <input type="text" size="4" maxlength="4" name="anno1"> </td> </tr>
17
18 <tr> <td> e anno </td>
19 <td> <input type="text" size="4" maxlength="4" name="anno2"> </td> </tr>
20
21 <tr> <td> Città: </td>
22
23 <td>
24 <select name="città">
25 <option value="Torino">Torino</option>
26 <option value="Milano">Milano</option>
27 <option value="Roma">Roma</option>
28 <option value="Napoli">Napoli</option>
29 </select>
30 </td> </tr>
31
32 </table>

```

15

**Passo 3**

➤ Creazione pagina Web statica


```

35 <br>
36
37 <input type="reset" value="Cancella">
38 <input type="submit" value="Invia">
39
40
41 </form>|
42
43
44 </body>
45 </html>

```

Il Submit innesca il passaggio dei parametri che saranno processati dall'application server per la creazione del contenuto dinamico

16



16




**Passo 4**

> Creazione pagina Web dinamica
 

- Creazione dinamica della pagina di risposta in funzione del contenuto del DB

> Pagina *test.PHP* richiamata dalla Action del form al Passo 3
 

- Contiene codice HTML statico per la strutturazione del contenuto della pagina
- Contiene codice PHP per la generazione del contenuto dinamico


17

17

**Passo 4**

> Creazione pagina Web dinamica
 

- Pagina HTML statica per la visualizzazione dei risultati (*test.php*)

```

<html>
<head>
<title>Risultati</title>
</head>
<body>
...
</body>
</html>

```


**Nel BODY inserisco il codice PHP necessario alla creazione del contenuto dinamico**

```

<?php
...
?>

```

**Il codice PHP sarà processato server-side**


18

18

**Passo 4**

> Creazione pagina Web dinamica
 

- Codice PHP

> Il codice PHP svolge i seguenti passi
 

1. Lettura dei parametri passati dal form e verifica di correttezza e consistenza
2. memorizzazione dei valori dei parametri in apposite variabili
3. Costruzione della query usando le variabili suddette
4. Connessione al DB
5. Esecuzione della query
6. Generazione del codice HTML con i risultati (visibile client-side mediante browser Web)
7. Chiusura della connessione al DB

**DBG**
19

19

**Passo 4**

> Creazione pagina Web dinamica
 

- Lettura e verifica dei parametri
  - Segnalazioni di errore per dati mancanti o inconsistenti (ad es. anno di fine < anno d'inizio)
  - Interrompe il flusso di esecuzione

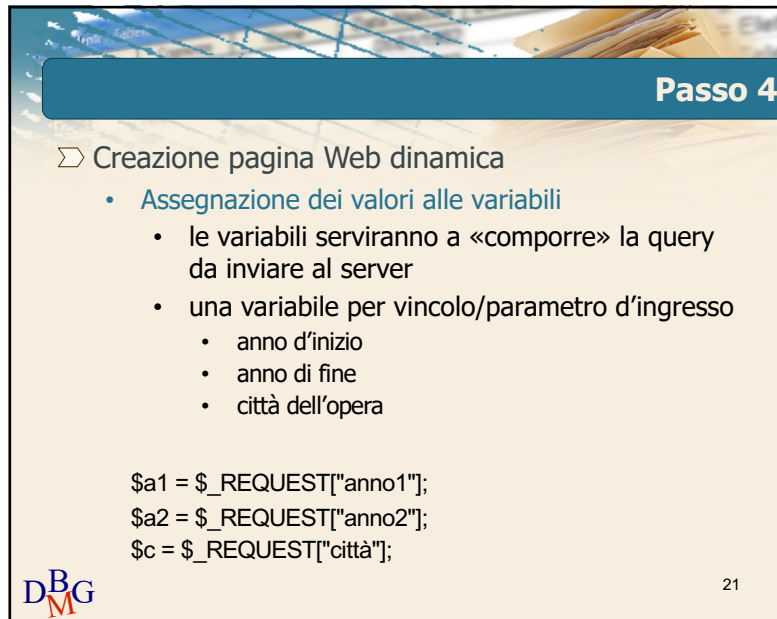
```

if( !isset($_REQUEST["anno1"]) or
!isset($_REQUEST["anno2"]) or !isset($_REQUEST["città"]){
    die("Errore: inserire tutti i dati richiesti");}

if( !is_numeric($_REQUEST["anno1"]) or
!is_numeric($_REQUEST["anno2"]) or ($_REQUEST["anno1"]
> $_REQUEST["anno2"]){
    die("Errore: anni non corretti");}
  
```

**DBG**
20

20




### Passo 4

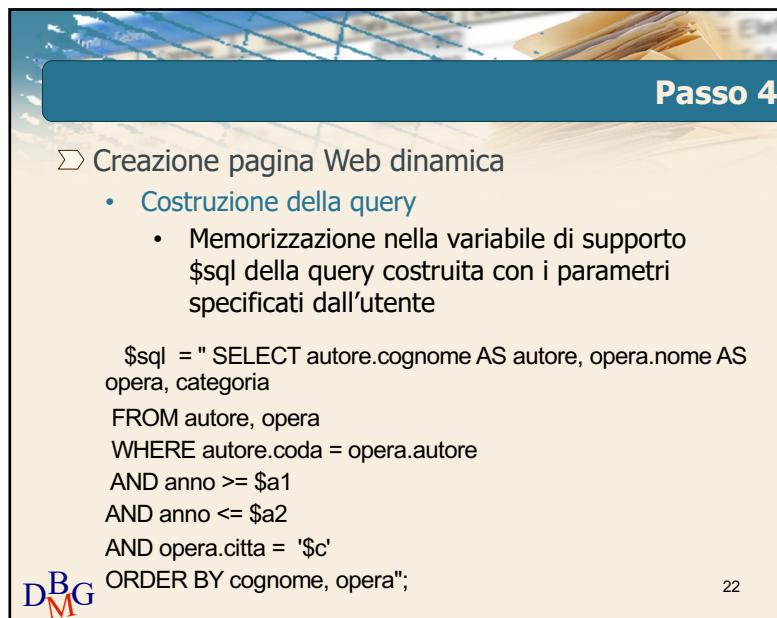
➤ Creazione pagina Web dinamica

- **Assegnazione dei valori alle variabili**
  - le variabili serviranno a «comporre» la query da inviare al server
  - una variabile per vincolo/parametro d'ingresso
    - anno d'inizio
    - anno di fine
    - città dell'opera

```
$a1 = $_REQUEST["anno1"];  
$a2 = $_REQUEST["anno2"];  
$c = $_REQUEST["città"];
```

 21

21




### Passo 4

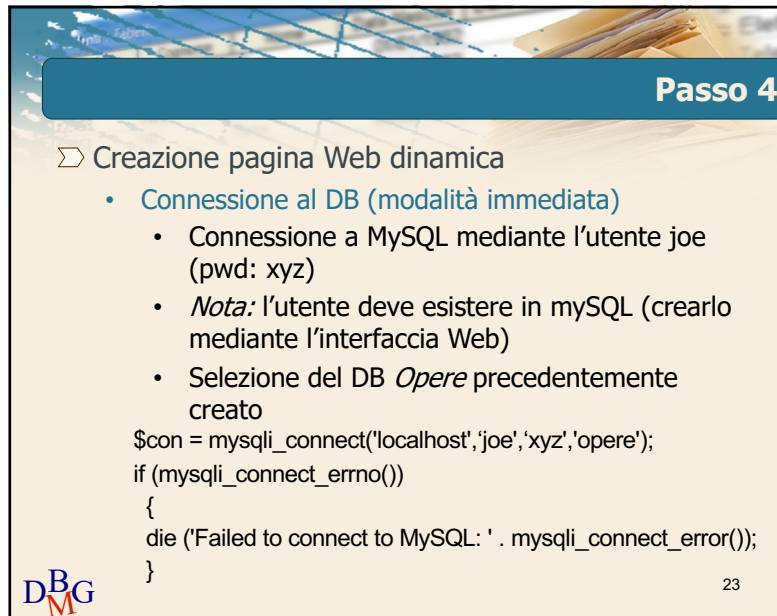
➤ Creazione pagina Web dinamica

- **Costruzione della query**
  - Memorizzazione nella variabile di supporto \$sql della query costruita con i parametri specificati dall'utente

```
$sql = " SELECT autore.cognome AS autore, opera.nome AS  
opera, categoria  
FROM autore, opera  
WHERE autore.coda = opera.autore  
AND anno >= $a1  
AND anno <= $a2  
AND opera.città = '$c'  
ORDER BY cognome, opera";
```

 22

22




**Passo 4**

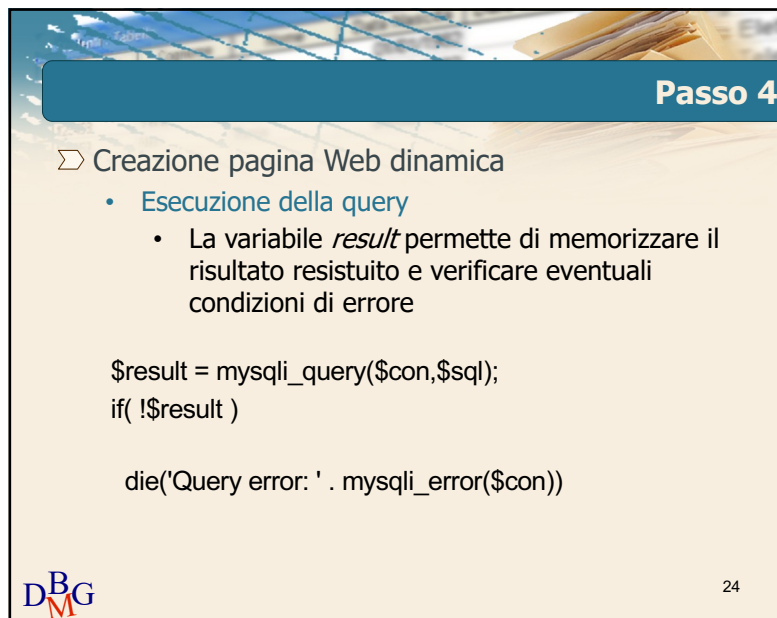
➤ Creazione pagina Web dinamica

- **Connessione al DB (modalità immediata)**
  - Connessione a MySQL mediante l'utente joe (pwd: xyz)
  - *Nota:* l'utente deve esistere in MySQL (crearlo mediante l'interfaccia Web)
  - Selezione del DB *Opere* precedentemente creato

```
$con = mysqli_connect('localhost','joe','xyz','opere');  
if (mysqli_connect_errno())  
{  
    die ('Failed to connect to MySQL: ' . mysqli_connect_error());  
}
```

 23

23




**Passo 4**

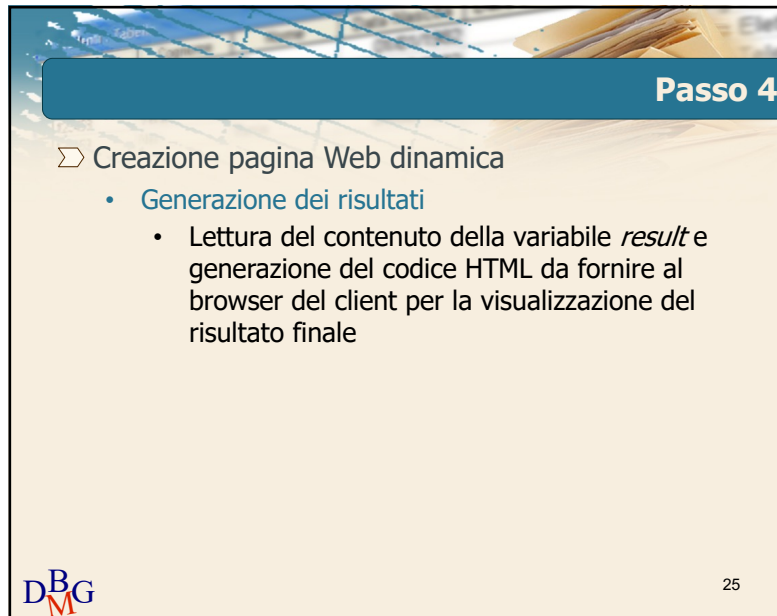
➤ Creazione pagina Web dinamica

- **Esecuzione della query**
  - La variabile *result* permette di memorizzare il risultato restituito e verificare eventuali condizioni di errore

```
$result = mysqli_query($con,$sql);  
if( !$result )  
  
    die('Query error: ' . mysqli_error($con))
```

 24

24



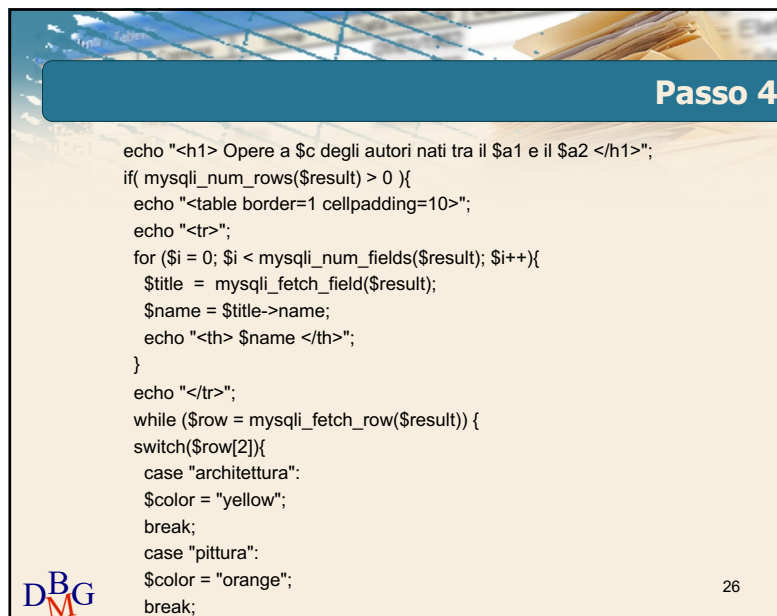
**Passo 4**

➤ Creazione pagina Web dinamica

- Generazione dei risultati
  - Lettura del contenuto della variabile *result* e generazione del codice HTML da fornire al browser del client per la visualizzazione del risultato finale

DBG  
25

25

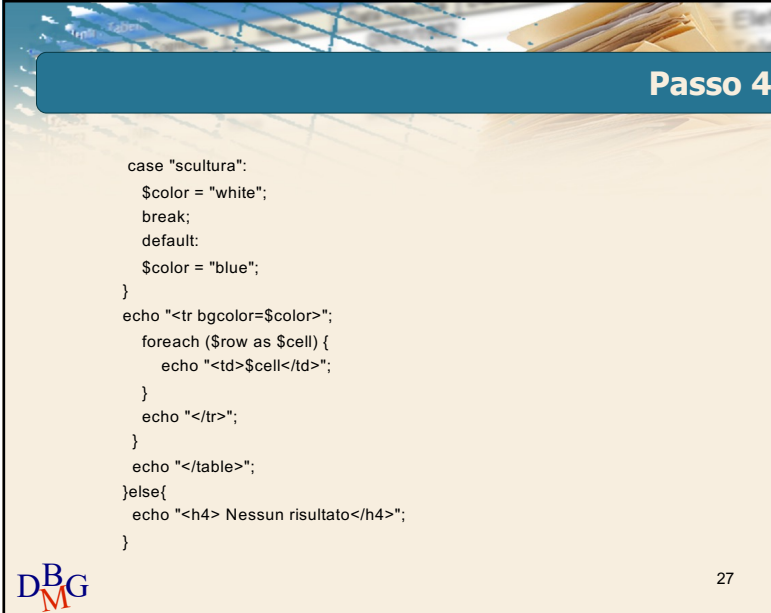


**Passo 4**

```
echo "<h1> Opere a $c degli autori nati tra il $a1 e il $a2 </h1>";
if( mysqli_num_rows($result) > 0 ){
  echo "<table border=1 cellpadding=10>";
  echo "<tr>";
  for ($i = 0; $i < mysqli_num_fields($result); $i++){
    $title = mysqli_fetch_field($result);
    $name = $title->name;
    echo "<th> $name </th>";
  }
  echo "</tr>";
  while ($row = mysqli_fetch_row($result)) {
    switch($row[2]){
      case "architettura":
        $color = "yellow";
        break;
      case "pittura":
        $color = "orange";
        break;
    }
  }
}
```

DBG  
26

26



**Passo 4**

```
case "scultura":
    $color = "white";
    break;
default:
    $color = "blue";
}
echo "<tr bgcolor=$color>";
foreach ($row as $cell) {
    echo "<td>$cell</td>";
}
echo "</tr>";
}
echo "</table>";
}else{
echo "<h4> Nessun risultato</h4>";
}
```

**DBG**

27

27