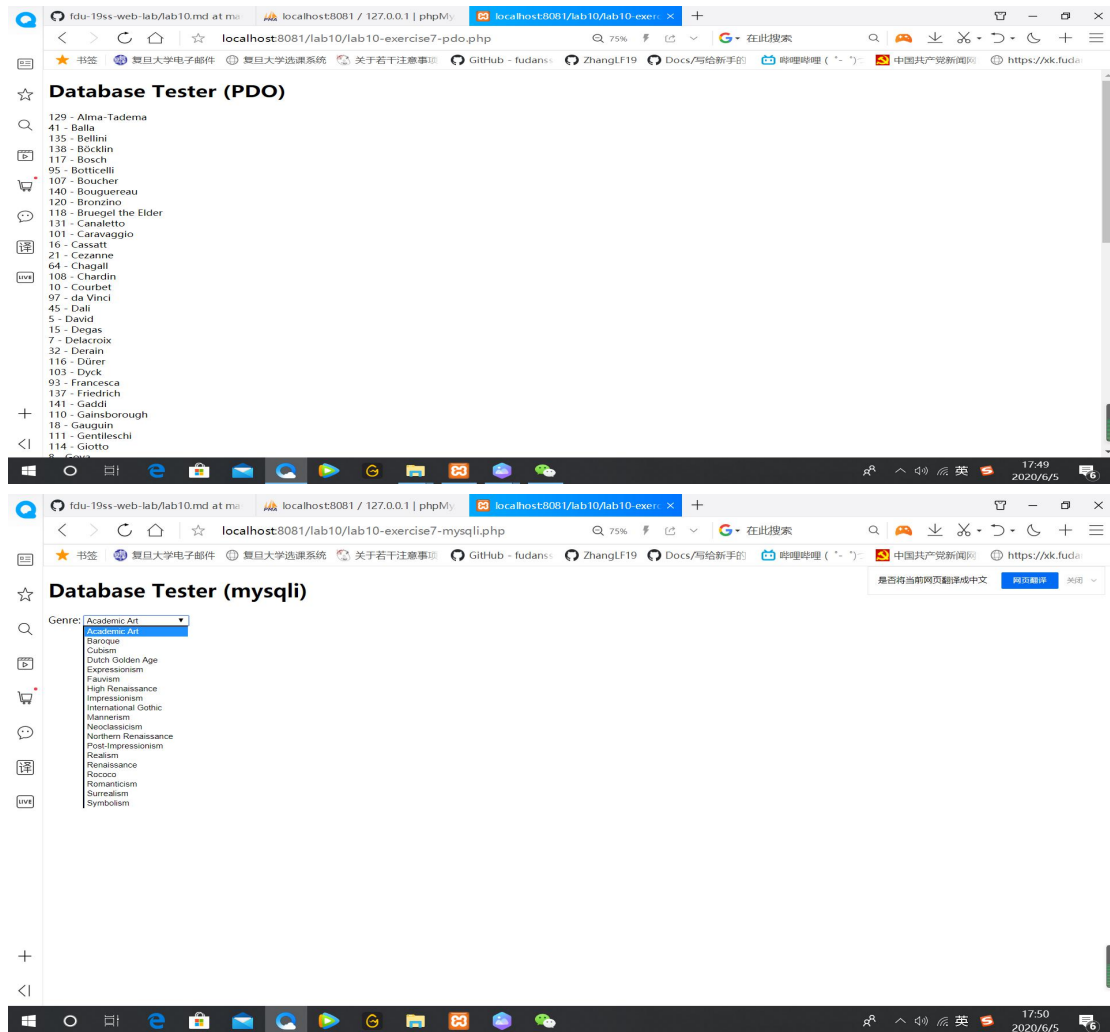


exercise7:



(1) Pdo

// 连接数据库。

```
$pdo = new PDO(DBCONNSTRING,DBUSER,DBPASS);
```

// setAttribute : 设置属性。ATTR_ERRMODE: 错误报告。ERRMODE_EXCEPTION: 抛出 exceptions 异常。

```
$pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
```

//设置查询。

```
$sql = "select * from Artists order by LastName";
```

//返回结果集。

```
$result = $pdo->query($sql);
```

// 使用循环获得数据行，输出内容。

最后释放资源，关闭数据库连接。

```
while ($row = $result->fetch()) {
```

```
    echo $row['ArtistID'] . " - " . $row['LastName'] . "<br/>";
```

```
}
```

```
$pdo = null;
```

(2) mysqli

//mysqli 函数，连接到 MySQL 服务器，获得数据库资源。

```
$connection = mysqli_connect(DBHOST, DBUSER, DBPASS, DBNAME);
```

// 检查尝试连接的结果。

```
if ( mysqli_connect_errno() ) {
```

```
    die( mysqli_connect_error() );
```

```
}
```

// 设置所执行查询

```
$sql = "select * from Genres order by GenreName";
```

// 如果查询成功，进入循环，mysqli_fetch_assoc() 返回代表读取行的关联数组。如果结果集中没有更多的行则返回 NULL。输出 option 标签及其内容。循环结束后，释放结果集。

```
if ($result = mysqli_query($connection, $sql)) {
```

```
    // loop through the data
```

```
    while($row = mysqli_fetch_assoc($result)) {
```

```
        echo '<option value="' . $row['GenreID'] . "'>';
```

```
        echo $row['GenreName'];
```

```
        echo "</option>";
```

```
    }
```

```
    // release the memory used by the result set
```

```
    mysqli_free_result($result);
```

```
}
```

```
mysqli_close($connection); //断开连接。
```

exercise8:

Screenshot of a web application running on localhost:8081/lab10/exercise8.php?id=95. The application displays a list of artists under the heading "User Input". The list includes: Alma-Tadema, Belli, Bellini, Böcklin, Bosch, Botticelli, Boucher, Bouguereau, Brancino, Brunel the Elder, Canaletto, Caravaggio, Cassatt, Cezanne, Chagall, Chardin, Courbet, da Vinci, Dali, David, Delacroix, Deraun, Dürer, Dyck, Francesca, Friedrich, Gaddi, Gainsborough, and Gauguin. The application also displays four detailed descriptions of Botticelli's paintings, each accompanied by a small image:

- Primavera**: Primavera, also known as Allegory of Spring, is a tempera panel painting by Italian Renaissance artist Sandro Botticelli. Painted ca. 1482, the painting is described in Culture & Values (2009) as "[o]ne of the most popular paintings in Western art". It is also "one of the most written about, and most controversial paintings in the world."
- The Birth of Venus**: The Birth of Venus is a 1486 painting by Sandro Botticelli. Botticelli was commissioned to paint the work by the Medici family of Florence, specifically Lorenzo di Pierfrancesco de' Medici under the influence of his cousin Lorenzo de' Medici, close friend to Botticelli. It depicts the goddess Venus, having emerged from the sea as a fully grown woman, arriving at the sea-shore (which is related to the Venus Anadyomene motif). The painting is on display at the Uffizi Gallery in Florence, Italy.
- Venus and Mars**: Venus and Mars is a c. 1483 painting by the Italian Renaissance master Sandro Botticelli. It shows the Roman gods Venus and Mars in an allegory of Beauty and Valour. Venus watches Mars sleep while two infant satyrs play carrying his armour as another rests under his arm. A fourth blows a small conch shell in his ear in an effort to wake him.
- The Mystical Nativity**: The Mystical Nativity is a painting of circa 1500–1501 by the Italian Renaissance master Sandro Botticelli, in the National Gallery in London. Botticelli built up the image using oil paint on canvas. It is his only signed work, and has a very unusual iconography for a Nativity.

+ outputArtists()

通过 PDO 连接 Artists 数据库，获得 30 行数据，在循环中将每一行的 LastName 在 a 标签中输出，并设置 a 标签的各项属性，通过 if 判断改变其 class 属性。最后释放资源，断开数据库连接。

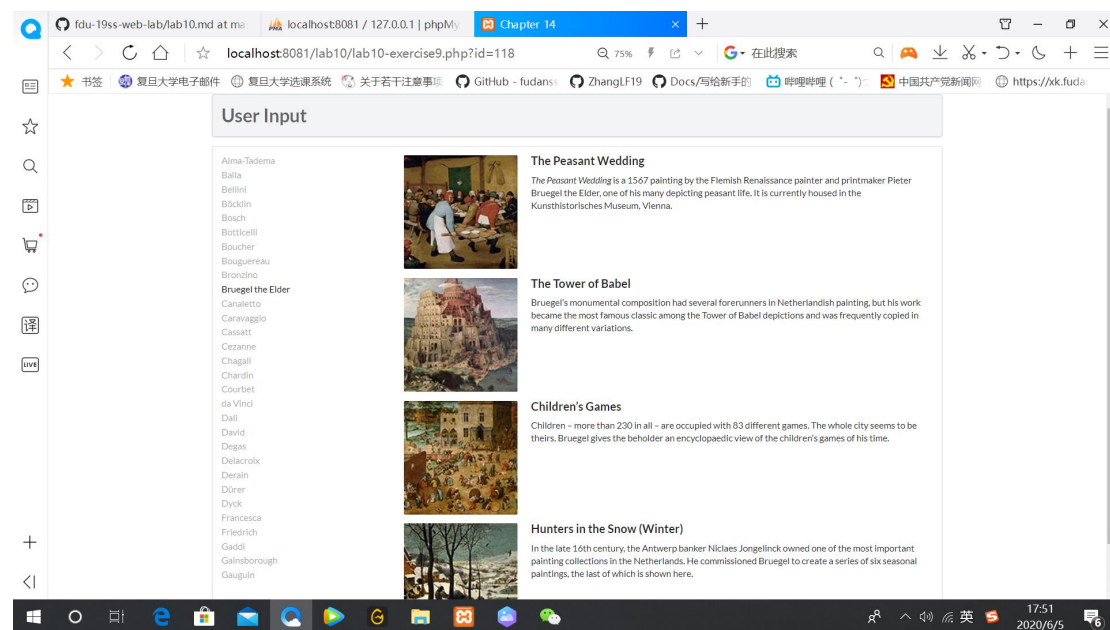
+ outputPaintings()

通过 PDO 连接 Paintings 数据库，通过点击获得的 id 定位得到数据，进入循环，执行 outputSinglePainting 方法，最后释放资源，断开数据库连接。

+ outputSinglePainting(\$row)

此方法通过获得的 \$row，定位 url 输出照片，标题以及介绍，

exercise9:



+ ResultSet executeQuery (String sql) throws SQLException: 用于查询。

+ int executeUpdate (String sql) throws SQLException: 执行 DDL、DML 语句，前者返回 0，后者返回受影响行数。

+ boolean execute (String sql) throws SQLException: 可执行任何 SQL 语句。如果执行后第一个结果为 ResultSet (即执行了查询语句)，则返回 true; 如果执行了 DDL、DML 语句，则返回 false。返回结果为 true，则随后可通过该 Statement 对象的 getResultSet() 方法获取结果集对象 (ResultSet 类型)，返回结果为 false，则可通过 Statement 对象的 getUpdateCount () 方法获得受影响行数。

PreparedStatement

+ PreparedStatement 预编译 SQL 语句，性能更好，执行更快。

+ PreparedStatement 无须“拼接”SQL 语句，编程更简单。

+ PreparedStatement 可以防止 SQL 注入（如将输入的 `true` 当成直接量，导致判断直接通过，从而降低了安全性），安全性更好。