In Magento 2, we can use models to access and modify the database data. But in some cases, we may require to communicate with the database using custom SQL queries. This blog will help you to run a custom SQL query without using model files.

Here the sample table structure,

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7 | CREATE TABLE IF NOT EXISTS `mytable` (      `id` int(11) NOT NULL AUTO\_INCREMENT,      `name` varchar(100) NOT NULL,      `email` varchar(255) NOT NULL,      `website\_url` varchar(255) NOT NULL,      PRIMARY KEY (`id`)  ) ENGINE=InnoDB DEFAULT CHARSET=utf8 AUTO\_INCREMENT=1 ; |

In order to communicate with the database in Magento 2, we need to instantiate  
**Magento\Framework\App\ResourceConnection** class. We can inject this resource class in our class constructor and run our custom queries by using that object within the class.

In the below example, I will be using an object manager to instantiate the resource class.

**Insert Query:**

Here is the example code to insert data on the database table,

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | $objectManager = \Magento\Framework\App\ObjectManager::getInstance();  $resource = $objectManager->get('Magento\Framework\App\ResourceConnection');  $connection = $resource->getConnection();  $tableName = $resource->getTableName('mytable');    $name = 'Thomas Raj';  $email = 'thomas.raj@example.com';  $website\_url = 'http://www.example.com';  $sql = "INSERT INTO " . $tableName . " (id, name, email, website\_url) VALUES ('', '$name', '$email', '$website\_url')";  $connection->query($sql); |

**Select Query:**

Here is the example code to fetch data from the database table,

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | $objectManager = \Magento\Framework\App\ObjectManager::getInstance();  $resource = $objectManager->get('Magento\Framework\App\ResourceConnection');  $connection = $resource->getConnection();  $tableName = $resource->getTableName('mytable');    $sql = "SELECT \* FROM " . $tableName;  $result = $connection->fetchAll($sql);  echo '<pre>'; print\_r($result); |

The above code will return the output as follows,

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10 | Array  (      [0] => Array          (              [id] => 1              [name] => Thomas Raj              [email] => thomas.raj@example.com              [website\_url] => http://www.example.com          )  ) |

Another method to fetch the data from the database, by using this, we can fetch specific field data instead of all the fields,

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13 | $objectManager = \Magento\Framework\App\ObjectManager::getInstance();  $resource = $objectManager->get('Magento\Framework\App\ResourceConnection');  $connection = $resource->getConnection();  $tableName = $resource->getTableName('mytable');    $id = 2;  $fields = array('id', 'name');  $sql = $connection->select()                    ->from($tableName, $fields)                    ->where('id = ?', $id);    $result = $connection->fetchAll($sql);  echo '<pre>'; print\_r($result); |

**Update Query:**

Here is the example code to update data on the database table,

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | $objectManager = \Magento\Framework\App\ObjectManager::getInstance();  $resource = $objectManager->get('Magento\Framework\App\ResourceConnection');  $connection = $resource->getConnection();  $tableName = $resource->getTableName('mytable');    $id = 1;  $sql = "UPDATE " . $tableName . " SET name = 'John', email = 'john@example.com', website\_url = 'http://www.mydomain.com' WHERE id = " . $id;  $connection->query($sql); |

**Delete Query:**

Here is the example code to delete data on the database table,

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8 | $objectManager = \Magento\Framework\App\ObjectManager::getInstance();  $resource = $objectManager->get('Magento\Framework\App\ResourceConnection');  $connection = $resource->getConnection();  $tableName = $resource->getTableName('mytable');    $id = 1;  $sql = "DELETE FROM " . $tableName . " WHERE id = " . $id;  $connection->query($sql); |