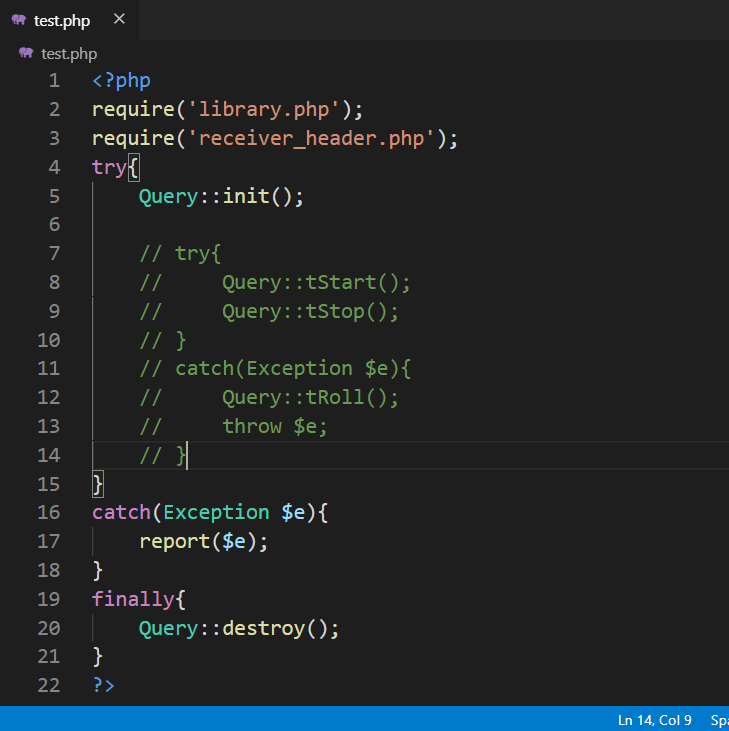
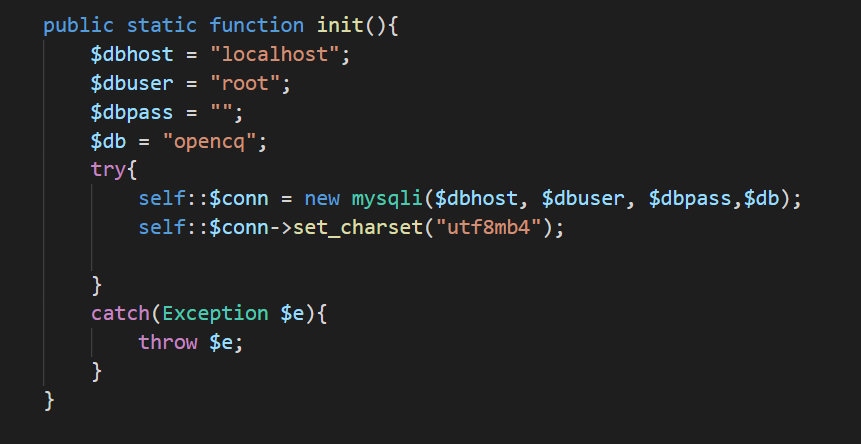
**How to use library.php?**

Template:

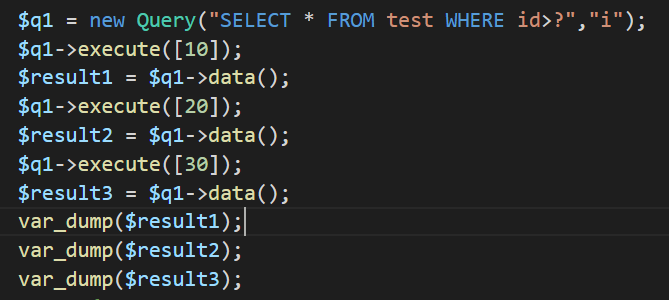


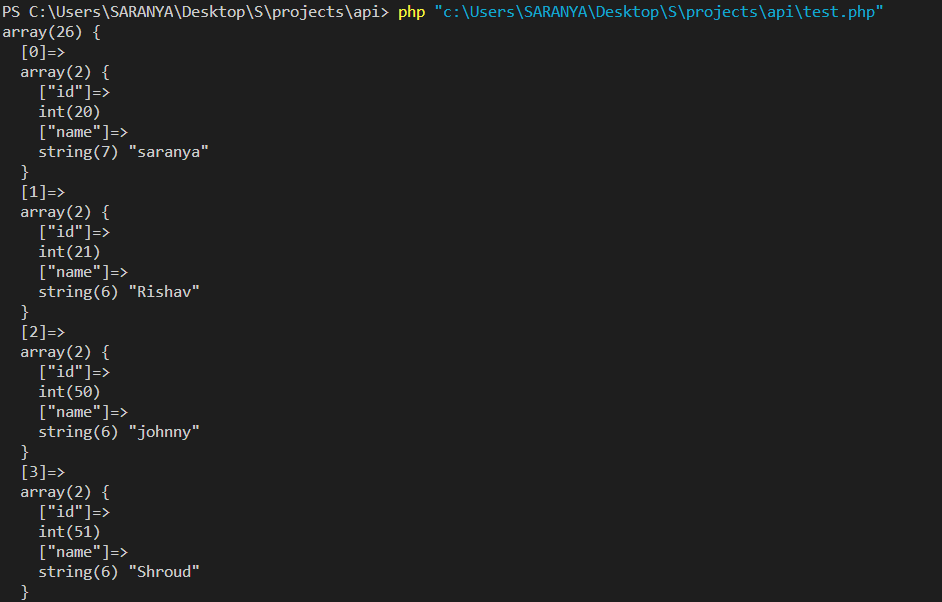
**Query::init()**



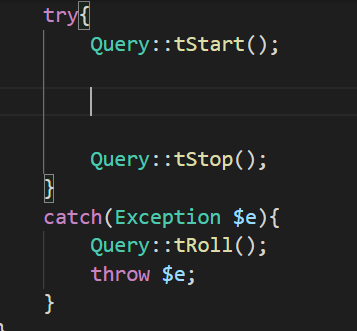
This actually establishes the connection with database. Place it at the top to be able to execute queries.

**How can we execute queries?**





**How to use Transactions?**Transactions are important when there are multiple inserts or updates. If at any time one query fails, we don’t want other queries to change anything in the database i.e. we want to rollback the changes in database.



Here we can group our queries in one transaction. If any query fails **Query::tRoll()** rollback the changes in database



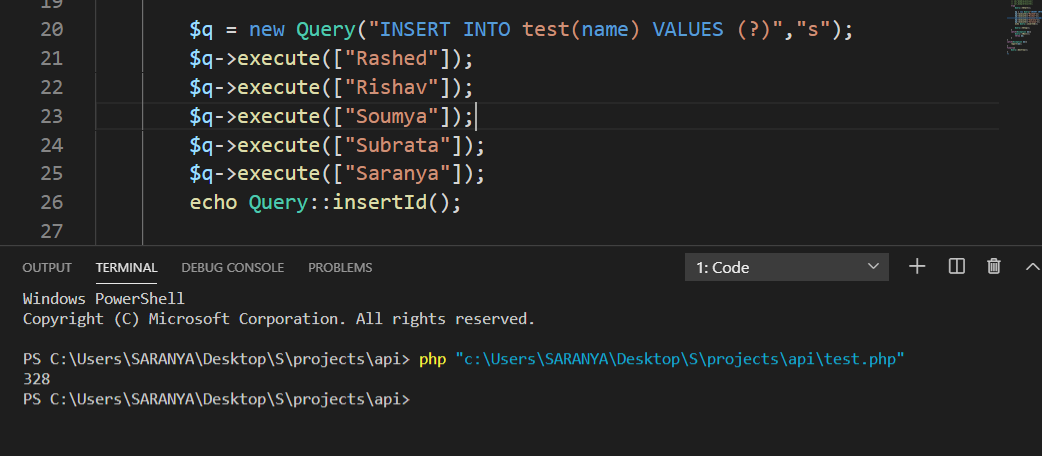
*Note: Above work can be done with one execute statement   
$q = new Query(“INSERT INTO test(name) VALUES (?),(?),(?),(?),(?)”,”sssss”);*

*$q->execute([“Rashed”,”Rishav”,”Soumya”,”Subrata”,”Saranya”]);*

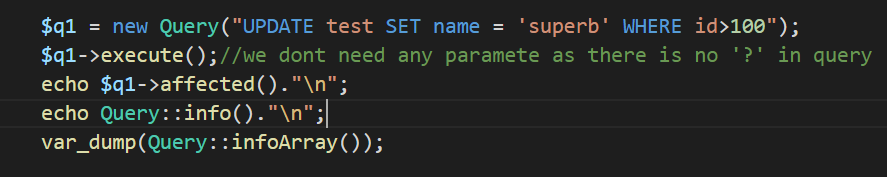
*Try to minimize the access to database.*

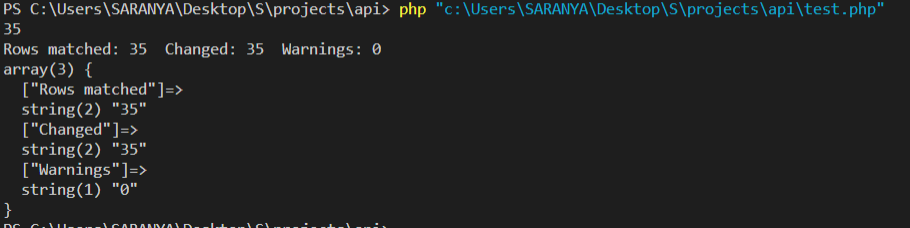
**How to get the last insert id ?**

Suppose there is an Auto Increment Column in a table. In that table we insert some rows. We need the value of the Auto Increment Column of last inserted row. We can do it using **Query::insertId()**



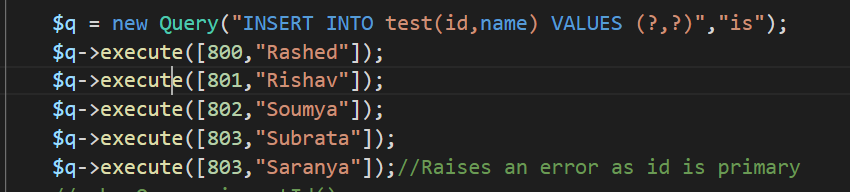
**Some other functions**

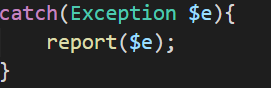


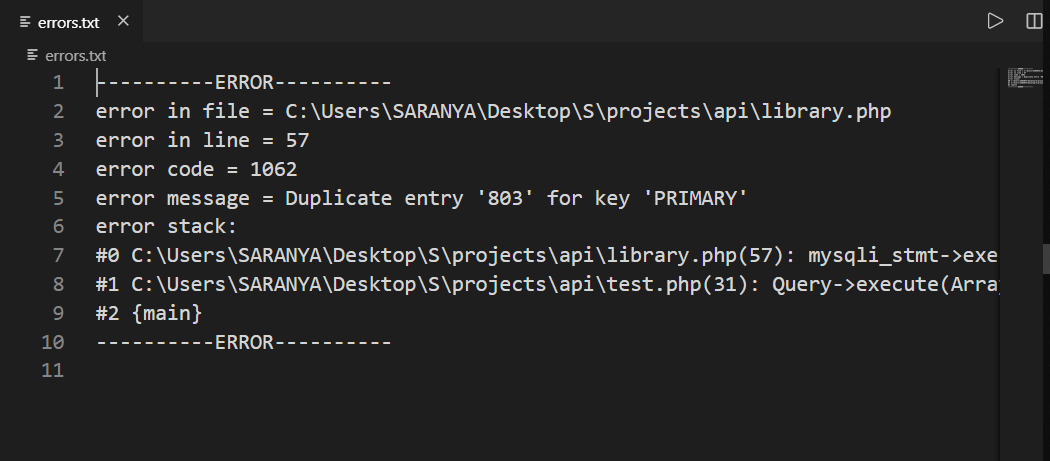


**Report**

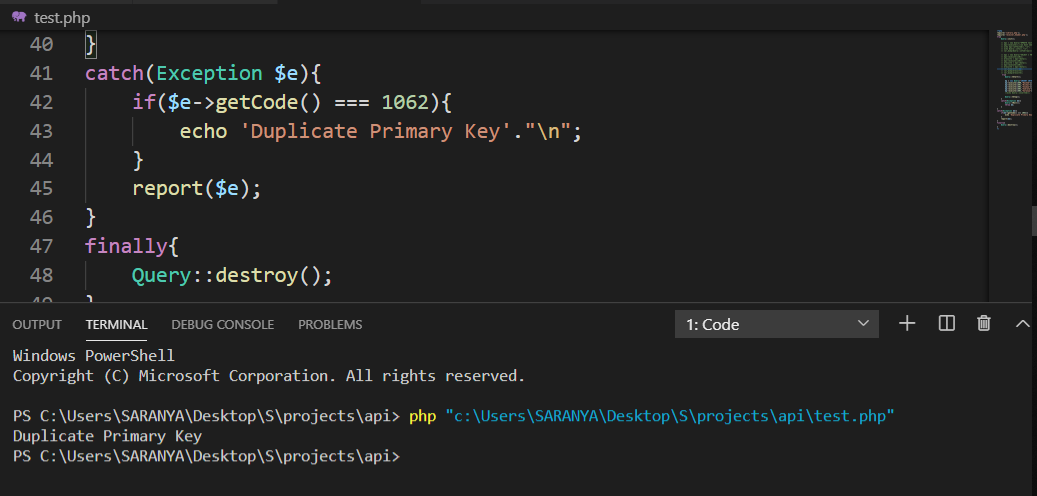
Reports in error.txt if there is some error







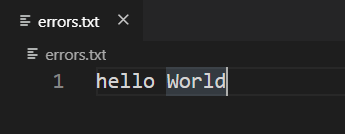
*Tip: Check Error Code to tackle different situations*



**Err()**

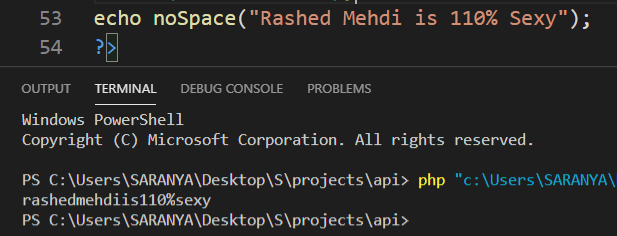
It is used to append anything in error.txt .





**noSpace()**

This function is for fun. This function removes all the whitespace characters from a string and convert it into lowercase.   
*Note: This function does not modify the string passed as its parameter. It returns a new string.*



**Query::destroy()** closes the connection with database.

**Template:**

