SQL Queries:

Sign Up Page
insert into portfolio_users (name, password, email) values ($name, $password, $email)
-inserts new user into the portfolio_users table

Portfolio Manager base page
select name from portfolios where owner='$user'
-gets the list of portfolios created by the owner

select id from portfolios where name='@rows[$i]' and owner='$user'
-only allows the portfolios that have been created by the user to be clicked on and accessed

New Portfolio Page
insert into portfolios (id, name, cash_amount, owner) values ($uuidstring, $portname, 0, $owner)
-creates a new portfolio

Portfolio Page
select name, cash_amount from portfolios where id='$id' and owner='$user'
-pulls up portfolios created by the user and the respective cash amounts

select stock_ticker from holding where portfolio_id='$id'
-gets the list of stock tickers in various holdings

select stock_ticker from holding where stock_ticker='@stocksList[$i]'
-again gets stock tickers created by the query above to create them dynamically

select quantity from holding where portfolio_id='$id' and stock_ticker='@stocksList[$i]'
-gets the quantity of each holding within a portfolio

Correlation Matrix
select stock_ticker from holding where portfolio_id='$tempid'
-pulls the stock tickers contained within the portfolio

Deposit/Withdraw Page
select cash_amount from portfolios where id='$tempid'
-pulls the current amount of cash in the portfolio

update portfolios set cash_amount = '$updateVal' where id='$tempid'
-updates the amount of money either deposited or withdrawn from the portfolio assuming its valid

update portfolios set cash_amount = '$updateVal' where id='$tempid'
-update portfolio amount in the case where all the money is being withdrawn

Buy/Sell Page
select cash_amount from portfolios where id='$tempid'
-gets the cash in the current portfolio

select stock_ticker from holding where portfolio_id='$tempid'
-grabs the tickers of all the stocks in the portfolio in holdings

select stock_ticker from holding where stock_ticker='@rows[$i]'
-same as above but done inside a loop to handle various amounts of different holdings

select quantity from holding where stock_ticker='@rows[$i]' and portfolio_id='$tempid'
-gets the quantity of stocks in the holdings

select quantity from holding where portfolio_id='$tempid' and stock_ticker='$Tickr'
-gets the quantity of stocks in the holding for a given portfolio and ticker

select quantity from holding where stock_ticker='$Tickr' and portfolio_id='$tempid'
-gets us the quantity of the stock holding

update holding set quantity='$newQuan' where stock_ticker='$Tickr' and portfolio_id='$tempid'
-updates the number of stocks we have in a holding in a portfolio

insert into holding (quantity, portfolio_id, stock_ticker) values ('$quan', '$tempid', '$Tickr')
-inserts into holding a new stock, quantity, and for a given portfolio

update portfolios set cash_amount='$cashamnt' where id='$tempid'
-updates the cash amount after purchasing or selling a stock

delete from holding where stock_ticker = '$Tickr' and portfolio_id='$tempid'
-if all of a stock holding is sold, we delete the record of it from holding

Stock
select quantity from holding where stock_ticker='$ticker' and portfolio_id='$tempid'
-pulls the quantity of the stock in the current holding

select timestamp, open, high, low, close, volume from cs339.StocksDaily where
timestamp<='$endTime' and timestamp>='$startTime' and symbol='$ticker'
-gets all the details of a stock across the given time window

Daily Stock Info

```
insert into daily_stocks (symbol, open, close, high, low, volume, timestamp) values
('$sym','$stockInfoArr[9]','$stockInfoArr[7]',null,null,'$stockInfoArr[13]','$timestamp')
```

-insert the desired information into the created daily stock information table

```
insert into daily_stocks (symbol, open, close, high, low, volume, timestamp) values
('$sym','$open','$close','$high','$low','$volume','$timestamp')
```

-inserts the manually entered user data into the daily stock info table

Future_Plot_Stock.pl

```
select timestamp, close from ".GetStockPrefix()."StocksDaily where symbol=$symbol union
select timestamp, close from daily_stocks where symbol='$symbol' order by timestamp
```

-Unioning the historic and new data for the future plot stock script

Get_covar.pl

```
  $sql = "select count(*),avg(l.$field1),stddev(l.$field1),avg(r.$field2),stddev(r.$field2) from
".GetStockPrefix()."StocksDaily l join ".GetStockPrefix()."StocksDaily r on l.timestamp=
r.timestamp where l.symbol='$s1' and r.symbol='$s2'";
    $sql.= " and l.timestamp>=$from" if $from;
    $sql.= " and l.timestamp<=$to" if $to;
    $sql.= " union all select count(*),avg(l.$field1),stddev(l.$field1),avg(r.$field2),stddev(r.$field2)
from daily_stocks l join daily_stocks r on l.timestamp= r.timestamp where l.symbol='$s1' and
r.symbol='$s2'";
    $sql.= " and l.timestamp>=$from" if $from;
    $sql.= " and l.timestamp<=$to" if $to;
```

-unioning the historic data and daily stock data in order to get covariance across both

```
 $sql = "select avg((l.$field1 - $mean_f1)*(r.$field2 - $mean_f2)) from
".GetStockPrefix()."StocksDaily l join ".GetStockPrefix()."StocksDaily r on
l.timestamp=r.timestamp where l.symbol='$s1' and r.symbol='$s2'";
      $sql.= " and l.timestamp>= $from" if $from;
      $sql.= " and l.timestamp<= $to" if $to;
      $sql.= " union all select avg((l.$field1 - $mean_f1)*(r.$field2 - $mean_f2)) from daily_stocks l
join daily_stocks r on l.timestamp=r.timestamp where l.symbol='$s1' and r.symbol='$s2'";
      $sql.= " and l.timestamp>= $from" if $from;
      $sql.= " and l.timestamp<= $to" if $to;
```

-unioning the historic data and daily stock data in order to get covariance across both

Get_data.pl

```
$sql = "select " . join(",",@fields) . " from ".GetStockPrefix()."StocksDaily";
$sql.= " where symbol = '$symbol'";
$sql.= " and timestamp >= $from" if $from;
```

$sql.= " and timestamp <= $to" if $to;
$sql.= " union select " . join(",",@fields) . " from daily_stocks";
$sql.= " where symbol = '$symbol'";
$sql.= " and timestamp >= $from" if $from;
$sql.= " and timestamp <= $to" if $to;
$sql.= " order by timestamp" if $from;
-unioning the historic data and daily stock data in order to get stock data in this perl script

Get_info.pl

```
  $sql = "select count($field), avg($field), stddev($field), min($field), max($field) from
".GetStockPrefix()."StocksDaily where symbol='$symbol'";
  $sql.= " and timestamp>=$from" if $from;
  $sql.= " and timestamp<=$to" if $to;
  $sql.= " union all select count($field), avg($field), stddev($field), min($field), max($field) from
daily_stocks where symbol='$symbol'";
  $sql.= " and timestamp>=$from" if $from;
  $sql.= " and timestamp<=$to" if $to;
```

-unioning the historic data and daily stock data in order to get stock info in this perl script

Plot_stock.pl

select timestamp, close from ".GetStockPrefix()."StocksDaily where symbol=$symbol and timestamp <= '$end' and timestamp >= '$start' union select timestamp, close from daily_stocks where symbol='$symbol' and timestamp <= '$end' and timestamp >= '$start' order by timestamp

-unions the historic data and the daily stocks table to plot the stock data queried