Useful path Module Functions

Function	Description
`path.basename(pathStr);`	Returns the filename of the **pathStr**. Ex. “picture.png” for “img/picture.png”
`path.extname(pathStr);`	Returns the file type/file extension of the **pathStr**. Ex. “.png” for “img/picture.png”
`path.dirname(pathStr);`	Returns the directory name of the **pathStr**. Ex: “img/” for “img/picture.png”

The glob module

Function	Description

`glob(pattern, callback);`
 	
`glob(“img/*”, (err, paths) => {`
`  ...`
`});`
 	
`// promisified`
`paths = await glob(“img/*”);`
	Globs all path strings matching a provided **`pattern`**. The pattern will generally follow the structure of a file path, along with any wildcards. If no paths match, the result array will be empty. If successful, passes the array of directory content paths (as strings) to the callback as **`contents`** parameter. Otherwise, passes error info to callback as **`error`** parameter.
	
	Common selectors:
	***** - A wildcard selector that will contextually match a single filename, suffix, or directory.
	****** - A recursive selector that will search into any subdirectories for the pattern that follows it.

The promise-mysql module

Function	Description
`mysql.createConnection({` **`  host : hostname, // default localhost`** **`  port : port,`** **`  user : username,`** **`  password : pw,`** **`  database : dbname`** **`});`**	Returns a connected database object using config variables. If an error occurs during connection (e.g. the SQL server is not running), does not return a defined database object.
`db.query(qryString);` **`db.query(qryString, [placeholders]);`**	Executes the SQL query. If the query is a SELECT statement, returns a Promise that resolves to an array of RowDataPackets with the records matching the **qryString** passed. If the query is an INSERT statement, the Promise resolves to an OkPacket. Throws an error if something goes wrong during the query. When using variables in your query string, you should use ? placeholders in the string and populate [placeholders] with the variable names to sanitize the input against SQL injection