Dellimeter //

Create procedure getaccountbyId(In id integer)

BEGIN

SELECT \* FROM tblaccount WHERE tblaccount.id = id;

END //

Dellimeter //

Create procedure acceptBidder(In JobId integer, In applicantId integer, In status varchar(255), In isAvailable varchar(255))

BEGIN

UPDATE job\_applicants JOIN jobs ON job\_applicants.job\_id = jobs.id

SET job\_applicants.status = status, jobs.is\_available = isAvailable

WHERE job\_applicants.applicant\_id = applicantId AND job\_applicants.job\_id = jobId;

END //

Dellimeter //

Create procedure getAcceptJobs(In id integer)

BEGIN

SELECT jobs.id, jobs.title as title, jobs.fixed\_price as fixedPrice, jobs.from\_price as fromPrice, jobs.to\_price as toPrice, jobs.description as description, jobs.subject as subject, jobs.from\_date as fromDate, jobs.to\_date as toDate, jobs.date\_posted as datePosted, jobs.language\_from as languageFrom, jobs.language\_to as languageTo, tblaccount.username as username, COUNT(job\_applicants.job\_id) as bids FROM jobs JOIN tblaccount

ON jobs.posted\_by\_id = tblaccount.id JOIN job\_applicants ON job\_applicants.job\_id = jobs.id

WHERE job\_applicants.applicant\_id = id AND jobs.is\_available = false AND job\_applicants.status = "accepted";

END //

Dellimeter //

Create procedure getAllJobs(In id integer)

BEGIN

SELECT jobs.id as id, jobs.category as category, jobs.description as description, jobs.fixed\_price as fixedPrice, jobs.from\_date as fromDate, jobs.from\_price as fromPrice, jobs.language\_from as languageFrom, jobs.language\_to as languageTo, jobs.level\_of\_confidentiality as leveOfConfidentiality, jobs.title as title, jobs.to\_date as toDate, jobs.to\_price as toPrice, jobs.subject as subject, jobs.posted\_by\_id as postedById, COUNT(job\_applicants.id) as bids FROM jobs LEFT JOIN job\_applicants ON job\_applicants.job\_id = jobs.id WHERE jobs.posted\_by\_id != id AND jobs.is\_available = 'true' GROUP BY jobs.id;

END //

Dellimeter //

Create procedure getBids(In id integer)

BEGIN

SELECT tblaccount.\*, tblrole.role\_type as roleType, T2.id as applicationId FROM jobs JOIN job\_applicants AS T2 on T2.job\_id = jobs.id JOIN tblaccount ON T2.applicant\_id = tblaccount.id JOIN tblrole on tblrole.id = tblaccount.roleid

WHERE jobs.id = id;

END //

Dellimeter //

Create procedure getFile(In id integer)

SELECT

\*

FROM

jobs JOIN tblaccount ON

tblaccount.id = jobs.posted\_by\_id

WHERE

jobs.posted\_by\_id = id;

END //

Dellimeter //

Create procedure getJobById(In id integer)

SELECT jobs.id as id, jobs.file as file, jobs.category as category, jobs.description as description, jobs.fixed\_price as fixedPrice, jobs.from\_date as fromDate, jobs.from\_price as fromPrice, jobs.language\_from as languageFrom, jobs.language\_to as languageTo, jobs.level\_of\_confidentiality as leveOfConfidentiality, jobs.title as title, jobs.to\_date as toDate, jobs.to\_price as toPrice, jobs.subject as subject, jobs.posted\_by\_id as postedById, jobs.date\_posted as datePosted, COUNT(job\_applicants.job\_id) as bids, tblaccount.username as username, tblaccount.email as email from tblaccount JOIN jobs on jobs.posted\_by\_id = tblaccount.id LEFT JOIN job\_applicants on job\_applicants.job\_id = jobs.id

WHERE jobs.id = id GROUP BY jobs.id;

END //

Dellimeter //

Create procedure getMembers(In id integer)

SELECT \* from tblaccount JOIN org\_members\_account on tblaccount.id = org\_members\_account.account\_id

WHERE org\_members\_account.org\_id = id AND tblaccount.is\_disabled = 'false';

END //

Dellimeter //

Create procedure getMyJobs(In id integer)

SELECT jobs.id, jobs.category, jobs.description, jobs.file, jobs.fixed\_price as fixedPrice, jobs.from\_date as fromDate, jobs.from\_price as fromPrice, jobs.language\_from as languageFrom, jobs.language\_to as languageTo, jobs.level\_of\_confidentiality as levelOfConfidentiality, jobs.price\_type as priceType, jobs.subject, jobs.title, jobs.to\_date as toDate, jobs.date\_posted as datePosted, tblaccount.username, COUNT(job\_applicants.job\_id) as bids from jobs LEFT JOIN job\_applicants on job\_applicants.job\_id = jobs.id JOIN tblaccount on jobs.posted\_by\_id = tblaccount.id WHERE jobs.posted\_by\_id = id GROUP BY job\_applicants.job\_id;

END //

Dellimeter //

Create procedure getPost()

SELECT tbl\_forum\_transactions.id, tbl\_forum\_post.topic as topic,tbl\_forum\_post.description as description, tbl\_forum\_post.date\_posted as datePosted, tblaccount.username as username, tbl\_forum\_transactions.account\_id as accountId, tbl\_forum\_transactions.post\_id as postId FROM tbl\_forum\_post JOIN tbl\_forum\_transactions ON tbl\_forum\_transactions.post\_id = tbl\_forum\_post.id JOIN tblaccount ON tblaccount.id = tbl\_forum\_transactions.account\_id;

END //

Dellimeter //

Create procedure getSaveJobs(In id integer)

SELECT jobs.id as id, jobs.category as category, jobs.description as description, jobs.fixed\_price as fixedPrice, jobs.from\_date as fromDate, jobs.from\_price as fromPrice, jobs.language\_from as languageFrom, jobs.language\_to as languageTo, jobs.level\_of\_confidentiality as leveOfConfidentiality, jobs.title as title, jobs.to\_date as toDate, jobs.to\_price as toPrice, jobs.subject as subject, jobs.posted\_by\_id as postedById, jobs.date\_posted as datePosted, tblaccount.username as username, tblaccount.email as email, COUNT(job\_applicants.job\_id) as bids FROM jobs JOIN save\_job on save\_job.job\_id = jobs.id JOIN tblaccount on jobs.posted\_by\_id = tblaccount.id LEFT JOIN job\_applicants on job\_applicants.job\_id = jobs.id

WHERE save\_job.save\_by\_id = id GROUP BY jobs.id;

END //

Dellimeter //

Create procedure viewForumDetails(In id integer)

SELECT tbl\_forum\_transactions.post\_id, tbl\_forum\_post.topic, tbl\_forum\_post.description, tbl\_forum\_post.date\_posted, tbl\_comments.comment, tbl\_comments.date\_commented, COUNT(tbl\_comment\_likes.id) from tbl\_forum\_transactions JOIN tbl\_forum\_post ON tbl\_forum\_transactions.post\_id = tbl\_forum\_post.id JOIN tblaccount on tblaccount.id = tbl\_forum\_transactions.account\_id LEFT JOIN tbl\_comments ON tbl\_comments.id = tbl\_forum\_transactions.comment\_id JOIN tbl\_comment\_likes ON tbl\_comment\_likes.comment\_id = tbl\_comments.id

WHERE tbl\_forum\_transactions.post\_id = id GROUP BY tbl\_comments.id;

END //