

-- Lab 03 Solutions --

-- Question 1 -

```
SELECT posterId, message, createdAt
FROM user_post
WHERE wallId IS NULL
ORDER BY createdAt DESC;
```

-- Question 2 --

```
SELECT u.id, u.firstName, u.lastName, u.middleName
FROM user u
WHERE u.middleName IS NULL
ORDER BY u.lastName;
```

-- Question 3 --

```
SELECT * FROM user u
WHERE timestampdiff(Week, registeredAt, now())<2;
```

-- Question 4 --

```
SELECT p.wallId, p.posterId, p.message, p.createdAt
FROM user_post p
WHERE p.wallId = 1
AND p.createdAt <= '2021-08-01';
```

-- Question 5 --

```
SELECT uf.sourceId, uf.targetId, uf.status, uf.updatedAt
FROM user_friend uf
WHERE uf.sourceId = 66 AND uf.status = 'rejected';
```

-- Question 6 --

```
SELECT uf.sourceId, COUNT(uf.targetId) AS NoOfFriends
FROM user_friend uf
WHERE uf.status = 'accepted'
GROUP BY uf.sourceId
ORDER BY NoOfFriends DESC;
```

DMM - Lab Tutorial Answer Key

-- Question 7 --

```
SELECT uf.sourceId, uf.targetId, uf.status, uf.updatedAt
FROM user_friend uf
WHERE uf.status='accepted'
AND uf.sourceId = 32
ORDER BY updatedAt DESC
LIMIT 1;
```

-- Question 8 --

```
SELECT g.*
FROM social.group g
WHERE g.status != ('blocked')
ORDER BY updatedAt DESC;
```

-- Question 9 --

```
SELECT gm.groupId, COUNT(gm.userId) AS NoOfMember
FROM group_member gm
WHERE gm.status = 'approved'
GROUP BY gm.groupId
ORDER BY NoOfMember;
```

-- Question 10 --

```
SELECT DISTINCT gm.groupId, gm.role, COUNT(gm.userId) AS
NoOfMember
FROM group_member gm
WHERE gm.status = 'approved' AND gm.role = 'administrator'
GROUP BY gm.groupId, gm.role
ORDER BY NoOfMember DESC;
```

-- Question 11 --

```
SELECT gm.userId, COUNT(gm.groupId) AS NoOfGroups
FROM group_member gm
WHERE gm.status = 'approved'
GROUP BY gm.userId
Order BY NoOfGroups DESC
Limit 10;
```

DMM - Lab Tutorial Answer Key

-- Question 12 --

```
SELECT groupId , Count(*) NumOfPosts
FROM group_post
WHERE createdAt between '2022-08-01' AND '2022-08-31'
GROUP BY groupId
ORDER BY NumOfPosts DESC
Limit 3;
```

-- Question 13 --

```
SELECT up.posterId, up.wallId, up.message, up.createdAt FROM
user_post up
WHERE up.message LIKE '%ban1%'
      OR up.message LIKE '%ban2%'
      OR up.message LIKE '%ban3%'
      OR up.message LIKE '%ban4%'
      OR up.message LIKE '%ban5%';
```

-- Lab 04 Solutions --

-- Task 1.

```
SELECT g.id, g.title, CONCAT(u.firstName, ' ', u.lastName) AS
Founder
FROM social.group g INNER JOIN user u
    ON g.createdBy = u.id;
```

-- Task 2

-- Using JOIN

```
SELECT ut.id AS targetId, ut.firstName AS targetName,
ut.lastName AS targetLastName,
ut.email AS targetEmail, ut.mobile AS targetMobile
FROM user ut INNER JOIN user_friend uf
    ON ut.id = uf.targetId
WHERE uf.sourceId = 11
AND uf.status = 'rejected';
```

```
SELECT u.id, u.firstName, u.lastName, u.email, u.mobile
FROM user u
WHERE u.id IN (
    SELECT uf.targetId FROM user_friend uf
    WHERE uf.sourceId = 11 AND uf.status = 'rejected'
);
```

-- Task 3

```
SELECT g.id, g.title, CONCAT(u.firstName, ' ', u.lastName) AS
PosterName, p.message, p.createdAt
FROM group_post p
    INNER JOIN social.group g ON p.groupId = g.id
    INNER JOIN user u ON p.userId = u.id
WHERE p.createdAt = (SELECT MIN(createdAt) FROM group_post pp
WHERE pp.groupId = p.groupId AND status = 'approved');
```

-- Task 4

This answer key provides sample queries as taught during the lab sessions, there can be different approaches of getting the required output.

DMM - Lab Tutorial Answer Key

```
SELECT g.id, g.title,
(SELECT COUNT(gp.id) FROM group_post gp INNER JOIN user u ON
u.id = gp.userId WHERE gp.groupId = g.id AND u.gender='m') AS
MalePosts,
(SELECT COUNT(gp.id) FROM group_post gp INNER JOIN user u ON
u.id = gp.userId WHERE gp.groupId = g.id AND u.gender='f') AS
FemalePosts,
(SELECT COUNT(gp.id) FROM group_post gp INNER JOIN user u ON
u.id = gp.userId WHERE gp.groupId = g.id AND u.gender IS NULL)
AS Unidentified
FROM social.group g;
```

-- Task 5

```
SELECT u.firstName, u.lastName, u.email FROM user u
WHERE u.id NOT IN (
    SELECT sourceId FROM user_friend
) AND u.id NOT IN (
    SELECT targetId FROM user_friend
);
```

-- Task 6

```
SELECT g.title, u.id AS userId, u.firstName, u.lastName
FROM group_member gm INNER JOIN user u
    ON gm.userId = u.id
    INNER JOIN social.group g
    ON g.id = gm.groupId
WHERE gm.groupId = 15
    AND gm.status = "approved"
    AND gm.userId NOT IN (
        SELECT userId FROM group_post WHERE groupId =
gm.groupId
    );
```

-- Task 7

```
SELECT u.*
FROM user u LEFT JOIN user_post up
    ON u.id = up.wallId
WHERE up.wallId IS NULL;
```

-- Task 8

This answer key provides sample queries as taught during the lab sessions, there can be different approaches of getting the required output.

DMM - Lab Tutorial Answer Key

```
SELECT u.id, u.firstName, u.email,
( SELECT COUNT(posterId) FROM user_post WHERE posterId = u.id
AND wallId IS NULL ) AS PublicPosts,
( SELECT COUNT(posterId) FROM user_post WHERE posterId = u.id
AND wallId IS NOT NULL ) AS PostsToFriend,
( SELECT COUNT(userId) FROM group_post WHERE userId = u.id AND
status = 'approved') AS GroupPosts,
( SELECT COUNT(posterId) FROM user_post WHERE posterId = u.id
AND wallId IS NULL ) +
( SELECT COUNT(posterId) FROM user_post WHERE posterId = u.id
AND wallId IS NOT NULL ) +
( SELECT COUNT(userId) FROM group_post WHERE userId = u.id AND
status = 'approved') AS TotalPosts
FROM user u;
```

-- Task 9

```
SELECT
(SELECT COUNT(*) FROM (SELECT COUNT(gm.id)
FROM sns.group g INNER JOIN group_member gm ON g.id = gm.groupId
WHERE gm.status = "approved"
GROUP BY g.id, g.title HAVING COUNT(gm.id) < 5) AS GP) AS
LightGroup,
(SELECT COUNT(*) FROM (SELECT COUNT(gm.id)
FROM sns.group g INNER JOIN group_member gm ON g.id = gm.groupId
WHERE gm.status = "approved"
GROUP BY g.id, g.title HAVING COUNT(gm.id) BETWEEN 5 AND 10) AS
GP) AS MediumGroup,
(SELECT COUNT(*) FROM (SELECT COUNT(gm.id)
FROM sns.group g INNER JOIN group_member gm ON g.id = gm.groupId
WHERE gm.status = "approved"
GROUP BY g.id, g.title HAVING COUNT(gm.id) > 10) AS GP) AS
DensedGroup;
```

-- Task 10

This answer key provides sample queries as taught during the lab sessions, there can be different approaches of getting the required output.

DMM - Lab Tutorial Answer Key

```
SELECT ( SELECT COUNT(u.id) FROM user u WHERE timestampdiff(DAY,  
u.registeredAt, now()) < 14) AS Newbie,  
      ( SELECT COUNT(u.id) FROM user u WHERE timestampdiff(DAY,  
u.registeredAt, now()) > 14  
        AND timestampdiff(DAY, u.registeredAt, now()) <  
21) AS Pros,  
      ( SELECT COUNT(u.id) FROM user u WHERE timestampdiff(DAY,  
u.registeredAt, now()) >= 21) AS Veterans;
```

-- Lab 05 Solutions --

Task 2:

```
HSET product 102 "{id:102, name:rockingChair, category:home,
status:available}"
```

Task 3:

```
HSET productCmnt C02 "{id:C02, comment:Office chair sorted,
username:jon.snow, productName:officeChair}"
```

Task 4:

```
MSET wishlist_count:P01 1 wishlist_count:P02 5
```

Task 5:

```
INCR wishlist_count:P01
```


-- Lab 06 Solutions --

Note: For questions 1 to 4 please refer to the lab slides

.

Task 5:

```
db.order.aggregate([{\n  $lookup: {\n    from: 'product',\n    localField: 'productId',\n    foreignField: 'id',\n    as: 'productInfo'\n  }\n}, {\n  $match: {\n    'productInfo.description.countrOfOrigin': 'Maldives'\n  }\n}, {\n  $group: {\n    _id: '$status',\n    productCount: {\n      $sum: 1\n    }\n  }\n}\n])
```

Task 6:

```
db.order.aggregate([{\n  $lookup: {\n    from: 'product',\n    localField: 'productId',\n    foreignField: 'id',\n    as: 'productInfo'\n  }\n}, {\n  $lookup: {\n    from: 'user',\n    localField: 'userId',\n    foreignField: 'userId',\n    as: 'userInfo'\n  }\n}, {\n  $match: {\n    status: 'Confirmed',\n    'payment.paymentStatus': 'Pending',\n    'payment.total': {\n      $gt: 5000\n    }\n  }\n}, {\n  $project: {\n    'userInfo.email': 1,\n    'payment.total': 1,\n    'payment.paymentStatus': 1,\n    status: 1\n  }\n}]
```

Task 7:

```
[{
  $lookup: {
    from: 'product',
    localField: 'productId',
    foreignField: 'id',
    as: 'productInfo'
  }
}, {
  $lookup: {
    from: 'user',
    localField: 'userId',
    foreignField: 'userId',
    as: 'userInfo'
  }
}, {
  $match: {
    status: 'Confirmed',
    'payment.paymentStatus': 'Pending',
    'payment.total': {
      $gt: 5000
    }
  }
}, {
  $group: {
    _id: '$userInfo.country',
    totalUsers: {
      $sum: 1
    }
  }
}, {
  $sort: {
    totalUsers: -1
  }
}]
```