**Queries Documentation for MyHealthTracker Application**

**Query 1: Fetch User Data**

**SQL Query:**

SELECT u.username, u.age, u.gender, u.weight, u.height, u.age\_group,

l.smoking, l.drinking, l.physical\_activity, l.education\_levels

FROM Users u

LEFT JOIN Life\_style l ON u.username = l.user\_username

WHERE u.username = %s;

**Purpose:** This query retrieves detailed information about a specific user, including their personal details (age, gender, weight, height, age group) and lifestyle data (smoking, drinking, physical activity, education levels).

**Usage:** Used in the /api/user\_data endpoint to display user information and calculate their BMI.

**Query 2: Add User Test Data**

**SQL Query:**

INSERT INTO User\_Tests (username, test\_name, test\_date, value)

VALUES (%s, %s, %s, %s);

**Purpose:** This query inserts a new medical test result for a user into the User\_Tests table. **Usage:** Used in the /api/add\_test endpoint to allow users to add their medical test results.

**Query 3: Fetch User Tests**

**SQL Query:**

SELECT \* FROM User\_Tests

WHERE username = %s

ORDER BY test\_date DESC;

**Purpose:** This query retrieves all test results for a specific user, ordered by the most recent test date.

**Usage:** Used in the /api/get\_user\_tests endpoint to display the history of user tests.

**Query 4: Predict Diabetes Risk**

**SQL Query (Fetching Data):**

SELECT Users.age, Users.height, Users.weight, Users.gender,

Life\_style.smoking, Life\_style.drinking, Life\_style.physical\_activity,

Life\_style.education\_levels

FROM Users

LEFT JOIN Life\_style ON Users.username = Life\_style.user\_username

WHERE Users.username = %s;

**SQL Query (Fetching Test Results):**

SELECT test\_name, value

FROM User\_Tests

WHERE username = %s AND test\_name IN ('BP\_HIGH', 'BP\_LWST', 'TOT\_CHOLE');

**Purpose:** These queries fetch user data and specific test results required to predict the risk of diabetes. **Usage:** Used in the /api/predict\_diabetes endpoint as part of the data preparation for the prediction model.

**Query 5: Predict Stroke Risk**

**SQL Query (Fetching User Data):**

SELECT Users.gender, Users.age, Users.height, Users.weight, Life\_style.smoking

FROM Users

LEFT JOIN Life\_style ON Users.username = Life\_style.user\_username

WHERE Users.username = %s;

**SQL Query (Fetching Test Data):**

SELECT test\_name, value

FROM User\_Tests

WHERE username = %s AND test\_name IN ('BP\_HIGH', 'BP\_LWST', 'BLDS');

**Purpose:** These queries retrieve user data and relevant test data to calculate the risk of stroke. **Usage:** Used in the /api/predict\_stroke endpoint for stroke risk prediction.

**Query 6: Predict Depression Risk**

**SQL Query:**

SELECT age, marital\_status, education\_levels, children, smoking, physical\_activity,

work, drinking, dietary\_habit, sleep\_pattern

FROM Users

LEFT JOIN Life\_style ON Users.username = Life\_style.user\_username

WHERE Users.username = %s;

**Purpose:** This query retrieves the lifestyle data needed to calculate the risk of depression. **Usage:** Used in the /api/predict\_depression endpoint for depression risk prediction.

**Query 7: Predict Heart Disease Risk**

**SQL Query:**

SELECT Users.age, Users.gender, Users.height, Users.weight, Life\_style.smoking,

User\_Tests.test\_name, User\_Tests.value

FROM Users

LEFT JOIN Life\_style ON Users.username = Life\_style.user\_username

LEFT JOIN User\_Tests ON Users.username = User\_Tests.username

WHERE Users.username = %s AND User\_Tests.test\_name IN ('BP\_HIGH', 'BP\_LWST', 'TOT\_CHOLE', 'BLDS');

**Purpose:** This query fetches user and test data required to calculate heart disease risk.

**Usage:** Used in the /api/predict\_heart\_disease endpoint for heart disease risk prediction.

**Query 8: Compare Tests**

**SQL Query (Create Temporary Table):**

CREATE TEMPORARY TABLE temp\_similar\_users AS

SELECT username

FROM Users

JOIN Life\_style ON Users.username = Life\_style.user\_username

WHERE age\_group = %s

AND education\_levels = %s

AND smoking = %s

AND drinking = %s

AND physical\_activity = %s

LIMIT 1000;

**SQL Query (Fetch Test Results):**

SELECT test\_name, value AS user\_value

FROM User\_Tests

WHERE username = %s;

**SQL Query (Generate Histogram Data):**

SELECT test\_name, FLOOR(value / 10) \* 10 AS bin, COUNT(\*) AS frequency

FROM User\_Tests

WHERE username IN (SELECT username FROM temp\_similar\_users)

GROUP BY test\_name, bin

ORDER BY test\_name, bin;

**Purpose:** These queries compare a user’s test results with those of similar users and generate histogram data for visualization.

**Usage:** Used in the /api/compare\_tests endpoint for statistical comparison.

**Query 9: Get Test Limits**

**SQL Query:**

SELECT test\_name, lower\_limit, upper\_limit

FROM Tests\_Values

WHERE age\_group = %s;

**Purpose:** This query retrieves the acceptable test value limits for a user's age group.

**Usage:** Used in the /api/get\_test\_limits endpoint to provide users with reference ranges for their test results.

**Query 10: Login Authentication**

**SQL Query:**

SELECT \* FROM Users

WHERE username = %s AND password = %s;

**Purpose:** This query verifies the user’s credentials for login.

**Usage:** Used in the /api/login endpoint to authenticate users.

**Query 11: Signup User**

**SQL Query:**

INSERT INTO Users (username, password, height, weight, age, age\_group, gender)

VALUES (%s, %s, %s, %s, %s, %s, %s);

**Purpose:** This query registers a new user by adding their information to the Users table.

**Usage:** Used in the /api/signup endpoint for user registration.

**Query 12: Update User Information**

**SQL Query:**

UPDATE Users

SET age = %s, height = %s, weight = %s

WHERE username = %s;

**Purpose:** This query updates an existing user's personal information.

**Usage:** Used in the /api/update\_user\_info endpoint to allow users to edit their details.

**Query 13: Get Health Alerts**

**SQL Query:**

SELECT t.test\_name, ts.full\_name, t.test\_date, t.value, v.lower\_limit, v.upper\_limit

FROM User\_Tests t

JOIN Tests\_Values v ON t.test\_name = v.test\_name

JOIN Tests ts ON t.test\_name = ts.test\_name

WHERE t.username = %s AND v.age\_group = %s

AND t.test\_date = (

SELECT MAX(t2.test\_date)

FROM User\_Tests t2

WHERE t2.username = t.username AND t2.test\_name = t.test\_name

)

ORDER BY t.test\_date DESC;

**Purpose:** This query retrieves the latest test results for a user along with the normal ranges for their age group.

**Usage:** Used in the /api/user\_health\_alerts endpoint to generate health alerts.

**Query 14: Update Lifestyle Information**

**SQL Query:**

INSERT INTO Life\_style (user\_username, marital\_status, education\_levels, children, physical\_activity, work, dietary\_habit, sleep\_pattern, drinking, smoking)

VALUES (%s, %s, %s, %s, %s, %s, %s, %s, %s, %s)

ON DUPLICATE KEY UPDATE

marital\_status=VALUES(marital\_status),

education\_levels=VALUES(education\_levels),

children=VALUES(children),

physical\_activity=VALUES(physical\_activity),

work=VALUES(work),

dietary\_habit=VALUES(dietary\_habit),

sleep\_pattern=VALUES(sleep\_pattern),

drinking=VALUES(drinking),

smoking=VALUES(smoking);

**Purpose:** This query updates or inserts a user's lifestyle information.

**Usage:** Used in the /api/update\_lifestyle\_info endpoint for managing lifestyle data.

**Notes**

* All queries are optimized for performance and include appropriate indexing.
* SQL placeholders (%s) are used to prevent SQL injection attacks.
* Additional error handling is implemented in the API code to manage unexpected scenarios.