Sources:
https://engineering.stackexchange.com/questions/3348/calculating-pitch-yaw-and-roll-from-mag-acc-and-gyro-data

Main Idea:

Problems with IMU sensors:

Accelerometer: Accelerometer data can not be quite accurate due to how it works. Accelerometer gets its data based on the forces that are applied to it. Now the problem with this is vibrations of where an accelerometer is stored i.e a vehicle. With vibrations, there can be many forces applied to the accelerometer which can cause noise and result in inaccurate readings. Vibrations can range from certain frequency ranges and may always differ depending on the motion of the vehicle such as acceleration, breaking, rotations, etc.

Magnetometer: Magnetometer data can not be accurate due to metals nearby that can likely distort the readings. Magnetometer gets its readings from measuring Earth's magnetic field. The problem with this is if there is a metal nearby, this can cause an influx because of an induced magnetic field. Now, placing the IMU near metal is common, but generally would be programmed to have a standard baseline calibration to get data after being placed at X location surrounded by X metals. Now distortion can be from say a car driving nearby or some metal object that appears at some instance.

Gyroscope: Gyroscope data measures angular velocity of how fast the IMU is rotating about each axis (x,y,z) and does it without the worry of magnetic flux or external forces. The problem with gyroscopes is drifting over time. Gyroscope readings are based on an integration (summation) of all tiny rotations (degrees or radians) over a period of time. Now readings are not guaranteed to be fully accurate and since it's over a period of time, imagine missing a data value by 0.01 seconds. This error will accumulate/continue with the next reading and before you know it there will be X.XX seconds of errors being read which is not to correct reading for real time/currently. This is exactly the same idea as gaming controllers, hence stick drift.

Solutions: SENSOR FUSION - the act of combining data from multiple sensors (acc, mag, gyr) to create a more accurate reading while being more reliable as a result (having an accelerometer and magnetometer to cancel out gyro drift).
> Kalman Filter

What I want:

I want to achieve receiving the data of Roll, Pitch and Yaw.

- Roll (ϕ, phi)
- Pitch (θ, theta)

- Yaw (ψ, psi)

EQUATIONS USED + SOURCE:

Source:
https://theccontinuum.com/2012/09/24/arduino-imu-pitch-roll-from-accelerometer/#exmp

With the formulas of:

Roll = atan2(Ay, Az) and Pitch = atan2(-Ax / (sqrt(Ay*Ay + Az*Az))

This gives roll and pitch based on acceleration data. The problem is as I stated from above, acceleration data is determined based on forces acting on it. This equation is based only off of gravitational force. This acts as a “baseline/calibration” of our roll and pitch which we can use a sensor fusion of gyro and acc:roll pitch into kalman filter to get more accurate data of roll and pitch that can then be used to get yaw.

Source:
https://www.artekit.eu/resources/ak-mag3110/doc/AN4248.pdf?utm_source

Page 7 Eqn.22 gives the formula to determine Yaw.

$$\Rightarrow tan(\psi) = \left(\frac{-B_{fy}}{B_{fx}}\right) = \left(\frac{(B_{pz} - V_z) sin\,\phi - (B_{py} - V_y) cos\,\phi}{(B_{px} - V_x) cos\,\theta + (B_{py} - V_y) sin\,\theta sin\,\phi + (B_{pz} - V_z) sin\,\theta cos\,\phi}\right)$$

phi = Roll
theta = Pitch
V = Bias/Weight
Bp = Raw magnetometer data

this equation gives Psi, which is Yaw.