Solving equations for the Atwood lab

```
# Set up the constants
g, w_1, w_2, a = var("g w_1 w_2 a")

# Set up the variables
b, F_f = var("b F_f")

# Set up substitutions
m_1 = w_1 + b
m_2 = w_2 + b
a_1 = 2*a
a_2 = a
```

1 | For Bucket

```
# solution_b = solve([2*(w_1+b)*(-2*a+g) == (w_2+b)*(a+g)+6*F_f], b)

solution_b = solve((m_2*a_2+m_2*g)/2 == m_1*-a_1+m_1*g, b)


b_a(a, w_1, w_2, g) = solution_b[0].rhs()
b_1=b_a(0.751109, 0.3, 0.4, 9.81) # group 1, trial 1
b_2=b_a(0.454363, 0.2, 0.2, 9.81) # group 2, trial 1
(b_1, b_2)
b_a

solve([b==b_1, b==b_2], (b, F_f))
```

We will also take the partial derivatives with respect to each variable, and square them, in order to be able to propergate error

Error by acceleration:

```
d = var("Δa")
bucket_accel_error = b_a.diff(a)*d
bucket_accel_error
```

Error by mass w_1:

```
d = var("Δw_1")
bucket_w1_error = b_a.diff(w_1)*d
bucket_w1_error
```

Error by mass w_2:

```
d = var("Δw_2")
bucket_w2_error = b_a.diff(w_2)*d
bucket_w2_error
```

Total error:

```
bucket_error = sqrt(bucket_accel_error^2 + bucket_w1_error^2 + bucket_w2_error^2)
bucket_error
```

2 | For Friction

```
solution_f = solve([2*(w_1+b)*(-2*a+g) == (w_2+b)*(a+g)+6*F_f], F_f)

ff_a(a, w_1, w_2, g) = solution_f[0].rhs()
ff_1=ff_a(0.751109, 0.3, 0.4, 9.81)
ff_2=ff_a(0.454363, 0.2, 0.2, 9.81)
(ff_1, ff_2)

solve([F_f==ff_1, F_f==ff_2], (b, F_f))
```

We will also take the partial derivatives with respect to each variable, and square them, in order to be able to propergate error

Error by acceleration:

```
d = var("Δa")
friction_accel_error = ff_a.diff(a)*d
friction_accel_error
```

Error by mass w_1:

```
d = var("Δw_1")
friction_w1_error = ff_a.diff(w_1)*d
friction_w1_error
```

Error by mass w_2:

```
d = var("Δw_2")
friction_w2_error = ff_a.diff(w_2)*d
friction_w2_error
```

Total error:

```
friction_error = sqrt(friction_accel_error^2 + friction_w1_error^2 + friction_w2_error^2)
friction_error
```